

wöchentlich

Compute mit

42/84 COMMODORE

2,80 DM
24 öS
2,80 sF

Das unabhängige Magazin für Anwender und Interessenten von Commodore Computern

Programme Programme Programme Programme

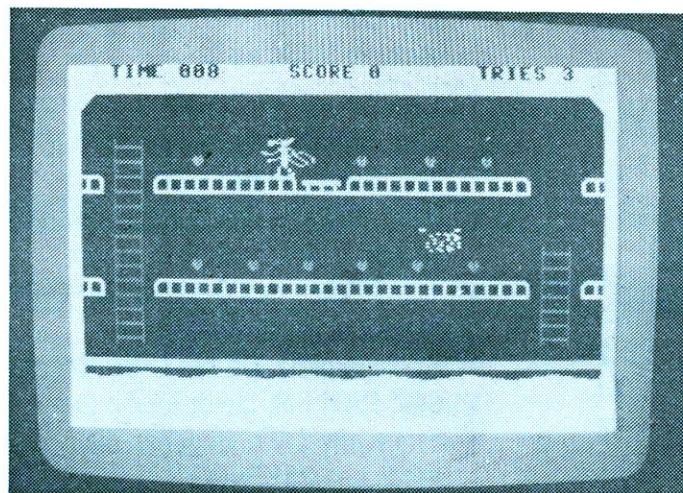
Programme Programme Programme Programme Programme Programme Programme Programme

Spiel und Anwenderprogramme für Ihren Commodore-Computer.

Wir haben für Sie u.a. ausgewählt:

★ Simons Newplot (C-64)

Die hervorragenden Eigenschaften des CBM 64 lassen sich nicht nur für Spiele verwenden. Benutzt man zum Programmieren noch eine Basic-Hilfe wie Simons Basic, so entstehen solche tollen Hilfsprogramme wie Simons Newplot mit dem man menügesteuert, mathematische Funktionen grafisch und farbig darstellen kann. Sie finden dieses tolle Anwenderprogramm auf Seite 25



★ Gold der Geister (VC-20)

Finden Sie das Gold der Geister und bringen es in Sicherheit.

★ Light Pen (VC-20)

Unergründlich und verschlungen sind die Wege, in denen Firmen die Preise für ihre Hardware errechnen.

Das Erweiterungen für Ihren Computer nicht teuer sein müssen, beweisen wir in dieser Ausgabe am Beispiel eines Lightpens. Sie finden die komplette Bauanleitung und Treibersoftware für VC-20 und CBM 64 auf Seite 21

★ Jumping Painter (VC-20)

Donkey Kong und Q-Bert sind tot... es lebe der Jumping Painter.

In diesem Heft eine Kombination dieser beiden weltbekannten Programme.



Hannover Messe'85

Hannover '85 –

ein Ausblick auf die Messe der Messen

Noch müssen sechs Monate ins Land ziehen, bis die Hannover-Messe '85 ihre Pforten öffnen wird, doch schon heute laufen die Vorbereitungen auf vollen Touren. Was für uns Fans der Informationstechnologie so alles ausgetüfelt und im April '85 präsentiert wird, erfahren Sie schon heute, wenn Sie Seite 4 ansteuern.

V-24 Schnittstellen- Umschalter

Besitzer einer umfangreichen Peripherie auf V-24-Basis finden mit diesem Gerät ein nützliches Requisit, das die tägliche Programmierarbeit wesentlich angenehmer macht.

Computer und Peripherie werden gemeinsam angeschlossen und können beliebig per Knopfdruck gekoppelt werden. Schauen Sie sich das Kästchen auf Seite 5 mal an!

Assemblerkurs (Teil 6)

In dieser Folge unseres Lehrganges für angehende Maschinensprache-Programmierer dreht sich wieder alles um Erweiterung des Editors. Wenn Sie den vorliegenden Teil durchgearbeitet haben, werden Sie in der Lage sein, reine Assemblerlistings über Ihren Drucker auszugeben, ohne den nur begrenzt leistungsfähigen "T"-Befehl einsetzen zu müssen.

Telex

Vorschau auf die Hannover-Messe '85

Mit dem Rückenwind eines alle bisherigen Ergebnisse übertreffenden Erfolges im April 1984 steuert der internationale Branchenverbund "Weltmarkt Elektronik" in eine neue Messe-Ära. **4**

Preissenkung für Typenraddrucker

V-24-Schnittstellenumschalter **5**

Adventure Ecke

Gruds in Space

Ein Adventure von Sirius, bei dem Sie Ihre Fähigkeiten als privater Händler in einem Handelsraumschiff unter Beweis stellen können. **6**

Software

Jumping Painter (VC-20)	7
Kaboom (VC-20)	14
Kniffel (VC-20)	16
Light Pen (VC-20)	21
Gold der Geister (VC-20)	23

Simons Neuplot (C-64)	25
Crazy Car (C-64)	35
Mensch ärgere Dich doch (C-64)	44
Programmverwaltung (C-64)	52

Werkstatt

Logik im Basicprogramm (Teil 6)

In der neuesten Folge unserer speziellen Serie für Freaks, die sich den letzten Schliff holen wollen, erfahren Sie wertvolle Dinge in Sachen Programmaufbau. Vom ersten Planspiel bis zum fertigen Basic-Listing ist es nicht weit, wenn man darauf **28**

verzichtet, blind drauf los zu tippen. Das Besondere: eine interessante Hausaufgabe, die innerhalb einer Woche zu bewältigen ist und Ihnen sicher einiges zu tüfteln geben wird.

Assembler-Kurs (Teil 6)

Auch diesmal bringen wir wieder eine Erweiterung für den Editor. **32**

Berichte

4. Internationale COMMODORE-Fachausstellung

Vom 6. bis 8. September 1984 fand in der Halle 1 des Frankfurter Messegeländes die 4. Internationale COMMODORE-Fachausstellung (CFA) statt. **30**

Speicherplatz geklaut? oder was passiert eigentlich mit dem 64K RAM des C-64?

Viele Käufer eines Commodore 64 wundern sich nach dem ersten Anschließen des Gerätes. Der C-64 meldet sich mit 38K freiem Speicherplatz. Da der C-64 jedoch 64K RAM hat, bleibt häufig die Frage offen: Wohin gehen die restlichen 26KByte? **56**

CM-Lexikon

Tips & Tricks

Softwarereviews

Cosmic Bouncer	
Raid on Bungeling Bay	
Megahawk	60

Basic Kurs Teil 5 **61**

Kleinanzeigen **62**

Rätselseite **63**

Impressum

Compute mit
erscheint wöchentlich im:
Roeske Verlag, Eschwege

Herausgeber:
Ralph Roeske

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

Freie Mitarbeiter:
St. Reusch, O. Panz, G. Okrouhly, J. Nielsen,
A. Knippschild, R. Hofmann, S. Kuehne,
M. Saur-Brasch, J. Frank

Gestaltung u. Fotografie:
Renate Wells, Gerd Köberich

Herstellung:
Roeske Verlag, Eschwege

Satz und Reproduktion:
Roeske Verlag, Eschwege

Druck:
Parzeller, Fulda

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

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

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

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

Erscheinungsweise:
Compute mit erscheint wöchentlich, Mittwoch

Anzeigenpreise:
Bitte Mediaunterlagen anfordern

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

© Roeske Verlag, Eschwege
Redaktion "Compute mit"

Bezugspreise:
Einzelheft: 2,80 DM

Autoren, Manuskripte:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen.
Sollte keine andere Vereinbarung getroffen sein, so gehen wir davon aus, daß Sie mit einem Honorar von DM 100,- pro abgedruckter Seite im Heft, einverstanden sind.
Bei Zusendung von Manuskripten und Software, erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger.
Rücksendung erfolgt nur gegen Erstattung der Unkosten. Zusendungen von Software zur Veröffentlichung soll bitte folgendes enthalten:
Kopierfähige Kassette oder Diskette mit dem Programm (Computer-Bezeichnung), von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinenlistings), evtl. Bildschirmfotos von einem Probelauf, ausführliche Programmbeschreibung, (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaues, Farbe, Grafik usw.)
Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.

C64/IEEE-488 Steckmodul

Dieser ausgereifte, weltweit erprobte IEEE-488-Modul eröffnet dem Commodore 64 über seinen parallelen Ausgang ungeahnte Einsatzmöglichkeiten wie:

große, IEEE-kompatible CBM-Peripherie am C-64, simultanen (seriell – VC/paralleler – IEEE) Datenverkehr. Konfliktfreie, speicherverschiebbliche Modulsoftware. Im Einsatz beispielsweise **in Schulen** ermöglicht der IEEE-488-Steckmodul problemlos Mehrbenutzersysteme am IEC-Bus wie auch durch die rationell genutzte Peripherie: z. B. zahlreiche Computer an einer Doppelfloppy.

In der Industrie bietet der IEEE-488-Steckmodul die Möglichkeit für preisgünstige IEC-Meß-/Steuersysteme mit dem Commodore 64 als Controller. Zu diesem Modul wird ein **Betriebshandbuch** geliefert, in dem Beschreibungen zu fast sämtlichen Anwendungsfällen mit Programmbeispielen, Belegungstabellen, Angaben zum erforderlichen Kabel- und Steckmaterial, Literatur etc. aufgeführt sind. Zusätzlich können zum IEEE-488-Steckmodul **Anwendungshilfen** wie u. a. Disketten mit Lesekennzeichen, Utility-Disketten usw. bezogen werden.

IEEE-Steckmodul für Commodore 64
einschließlich Betriebshandbuch DM 239,— inkl. MwSt.



te-wi

te-wi Verlag GmbH
Theo-Prosel-Weg 1
8000 München 40

Weiterführende Literatur...



NEU! C-64 Computerhandbuch

Ein Handbuch für jeden Erfahrungsstand: von der ersten Begegnung bis zum professionellen Einsatz des COMMODORE 64 bzw. 1541. Das Werk ist sehr bildreich und bietet somit eine schnelle Übersicht – als echtes Nachschlagewerk werden Sie es stets in der Nähe Ihres Computers finden.

Raeto West, ca. 400 Seiten, Softcover, DM 56,—, 4. Qu. 84



NEU! C-64 Akustik und Graphik

Ein planvoller Lehrgang – keine Beispielsammlung – in anschaulichem Stil – daher für jedes Alter. Dieses Werk eröffnet dem C-64-Benutzer die Welt der Graphiken und Klangbilder. Es enthält Programmbibliotheken und wird abgerundet durch zahlreiche Anhänge.

John Anderson, ca. 200 Seiten, Softcover, DM 49,—, 4. Qu. 84



6502 - Programmieren in Assembler

Dieses Buch behandelt ausführlich die Assemblersprachen-Programmierung für den weitverbreiteten Mikroprozessor 6502. Er steckt auch in Ihrem C-64.

Lance Leventhal, 704 Seiten, Softcover, DM 59,—



CBM Computer Handbuch

Dieses unentbehrliche Nachschlagewerk bietet eine wahre Fundgrube – mit einer schrittweisen Einführung bis hin zur Darstellung aller professionellen Möglichkeiten dieses beliebten Computers.

Osborne/Danahue, 544 Seiten, Softcover, DM 59,—



NEU! LOGO Computersprache für Kinder und Eltern

Dieses Buch beweist: **Jeder kann programmieren.** LOGO ist die Computersprache für Eltern und Kinder. Nicht umsonst wurde dieser Titel zum „Buch des Jahres 1983“ in den USA. LOGO ist das Ergebnis der Erforschung menschlicher Intelligenz; entwickelt von einem Pädagogen und Mathematikprofessor. LOGO ist die erste Computersprache, die bewußt Strategien menschlichen Denkens dient.

Daniel Watt, ca. 400 Seiten, Softcover, DM 59,—, 4. Qu. 84

Coupon

Hiermit bestelle(n) ich (wir):

Datum

Unterschrift

Name/Firma

Straße/Postfach

Ort

Der Sensible C-64
CP/M und WordStar
C-64 Programmsammlung
VisiCalc (mit CBM Diskette)
77 BASIC Programme
Mikrocomputer-Grundwissen
C-64 Computer für Kinder
VC20 Computer für Kinder

4. Q. 84, DM 29,80
DM 29,80
4. Q. 84, DM 29,80
DM 79,—
DM 39,—
DM 36,—
4. Q. 84, DM 29,80
4. Q. 84, DM 29,80



Neue Messe-Ära für Elektronik und Elektrotechnik

**Acht Hallen der Hannover-Messe '85 für die
Elektroindustrie – Mikroelektronik
durchdringt alle Hallen**

Mit dem Rückenwind eines alle bisherigen Ergebnisse übertreffenden Erfolges im April 1984 steuert der internationale Branchenverbund "Weltmarkt Elektronik und Elektrotechnik" in eine neue Messe-Ära. Von der Hannover-Messe '85 an (17. bis 24. April) stellt sich die größte Fachmesse der "Messe der Messen" in weiten Teilen strukturell verbessert vor. Über die Zielsetzung sind sich die ausstellende Industrie und die Deutsche Messe- und Ausstellungs-AG einig:

- 1.) Das weltweit einmalige Angebot an elektronischen und elektrotechnischen Investitions- und Gebrauchsgütern sowie Technologie-Know-how wird noch weiter ausgebaut, qualifiziert und konzentriert.
- 2.) Die Vorteile des Technologie- und Angebot-Verbundes innerhalb der Elektrobranche und darüber hinaus mit den benachbarten weiteren neun Fachmessen der Hannover-Messe werden noch stärker nutzbar gemacht.

Zur Hannover-Messe '85 erhöht sich die Netto-Ausstellungsfläche des "Weltmarktes Elektronik und Elektrotechnik" auf über 126 000 Quadratmeter. Darüber wurde die Halle 13 endgültig in den Elektrobereich einbezogen, und die Hallen 11 und 12 sind durch neue, breite Übergänge im ersten Obergeschoß noch funktioneller miteinander verbunden. Die nun-

mehr acht Elektrohallen (von 8 bis 15) und das Elektrofriegelände nehmen rund 1900 Aussteller, davon 550 (knapp 30 Prozent) aus dem Ausland auf.

Mit den strukturellen Anpassungen, die überall im Bereich und Umfeld der Hannover-Messe stattfinden, entspricht die Messe-AG dem aktuellen Wandel, den in der Industrie vor allem die Schlüsseltechnologie Mikroelektronik auslöst. Diese innovative Technologie wird im Rahmen der Fachmesse "Weltmarkt Elektronik und Elektrotechnik" im "microtronic – Innovationszentrum der Mikroelektronik" dargestellt. Sie durchdringt in vielfältigen Anwendungen auch alle anderen Schwerpunkte der Hannover-Messe. Einzelaussteller, themenorientierte Ausstellungszentren und ein "microtronic-Forum" geben einen kompletten Überblick über das umfassende Spektrum der Bauteile, Bauelemente, Baugruppen und Entwicklungssysteme, der Elektronik/Mikroelektronik sowie ihrer zahlreichen Einsatzmöglichkeiten. Das zugrundeliegende Drei-Säulen-Konzept ist speziell auf das Ziel ausgerichtet, die Einstiegsbarrieren kleinerer und mittlerer Unternehmen zu überwinden und mit der modernen Technik neue Märkte zu erschließen.

Die elektrotechnische Industrie ist Entwickler, Hersteller und in großem Maßstab auch Anwender der Mikroelektronik. Dementsprechend gehört

sie in allen acht Angebotschwerpunkten des "Weltmarktes Elektronik und Elektrotechnik" zum technischen Standard. Die Hannover-Messe '85 präsentiert das aktuelle Programm der "Elektrischen Energieversorgung, -umwandlung und -verteilung", die vollständige "Meß-, Prüf-, Steuerungs- und Regeltechnik" sowie die "Industrielle Automatisierungstechnik". Parallel dazu gilt das Angebot der "Nachrichtentechnik" weltweit als umfassendes Informationsforum dieses Technologiesektors; die fortschrittlichen Bausteine der moder-

nen Kommunikationssysteme schließen z.B. die Lichtfasertechnik und die Satelliten-Nachrichtentechnik ein. Weitere Schwerpunkte sind die Bereiche "Elektrische Installationsgeräte und -systeme" und "Sicherheitstechnik". Auf keinem anderen Messeplatz wird ein umfassenderes Angebot an moderner Installations-, Hausleit- und Sicherungstechnik speziell dem Elektrohandwerk und -handel näher gebracht. Darüber hinaus wendet sich die Gesamt-Präsentation an jegliche Anwender von Installationstechnik. Ebenso übergreifend spricht das neukon-



zipierte Programm "Fertigungstechnik und Zubehör für Elektronik und Elektrotechnik" die Industrie aber auch das Handwerk an. Die Maschinen und Einrichtungen tragen zur rationellen Be- und Verarbeitung sowie zur Serienfertigung der unterschiedlichen Produkte bei. Zum Zubehörangebot gehören eine umfangreiche Palette von Leerschranken und Gehäusen sowie Vorprodukte und Bauteile für die Elektronik und Elektrotechnik. Außerdem sind hier die Firmen mit Elektrowärmeanlagen konzentriert.

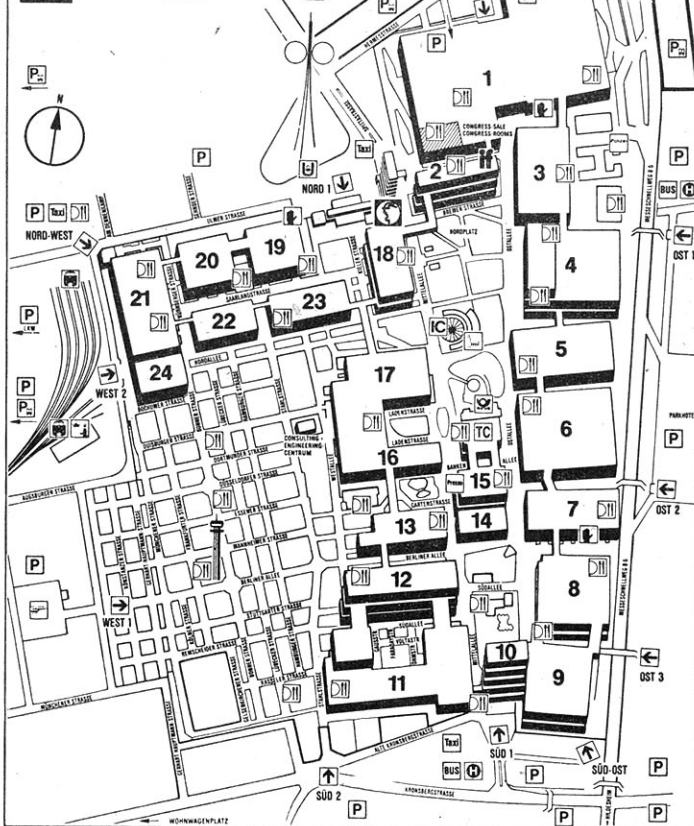
Den Reigen des größten Marktes elektronischer und elektrischer Produkte beschließt die "Weltlichtschau Leuchten und Lampen". Unter den wichtigsten internationalen Ordermessen für Fachhandel, Einrichtungshäuser und Innenarchitekten gehört sie weltweit zur Nr. 1. Die

Auszeichnung bezieht die technischen Leuchten und Lampen und nicht zuletzt das Zubehör dieser Branche mit ein, das hier komplett vorgestellt wird.

Zum ständig wachsenden Erfolg des "Weltmarktes Elektronik und Elektrotechnik" tragen unbestritten die intensiven Wechselbeziehungen im Verbund aller Fachmessen der Hannover-Messe bei. Die Kontakte reichen vom (A)nlagenbau bis zur (Z)uliefermesse und führten 1984 rund 450 000 (60%) von insgesamt 740 000 Besuchern der Hannover-Messe auf die Stände der internationalen Elektro-Industrie.

Mittwoch, 17. – Mittwoch, 24. April

Hannover Messe '85



Preissenkung für Typenraddrucker

Der Preis der **Olympia ESW 102** wurde bei iti-Datentechnik um DM 1.000,- auf DM 2.298,- inkl. MwSt. gesenkt. Dafür erhält man einen zuverlässigen Typenraddrucker aus westdeutscher Fertigung mit 17 Zeichen/Sekunde

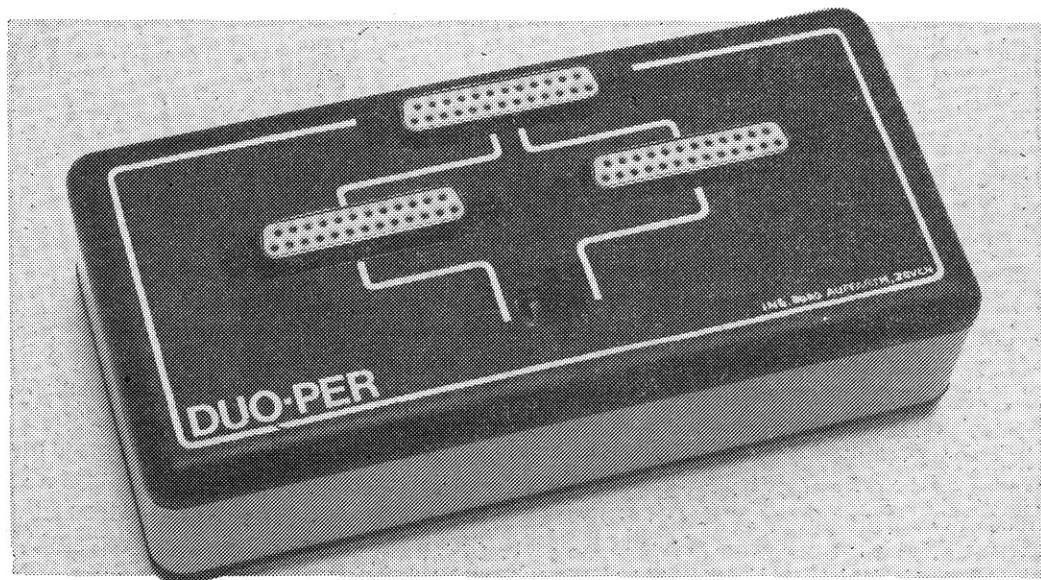
Schreibgeschwindigkeit, Druckwegoptimierung, DIN A3 Querformat, Farbbanden- und Papierendekennung. Alle Schnittstellen (Centronics, V.24, IEC-Bus) sind mit einem 4 KByte Puffer ausgestattet. Gegen Aufpreis ist ein

eingebautes Interface zum Commodore C-64 lieferbar. Ohne Aufpreis ist die ESW 102 mit den Zeichensätzen für Commodore, DEC, HP, IBM, Wang und anderen Rechnern von ITI ausstattbar.

Das reichhaltige Zubehör wie Traktor, Einzelblattzuführung (Einschacht- und Doppelschacht), Kabel, Typenräder und Farbbänder wird ebenso wie die Maschine selbst sofort ab Lager Leonberg geliefert.

V-24-Schnittstellenumschalter

Unter der Bezeichnung DUOPER hat das Ing.-Büro Auffarth einen Umschalter für V-24-Schnittstellen neu in sein Produktionsprogramm aufgenommen. Das Gerät hat drei 25 pol. D-Stecker, an die der Computer und zwei Peripheriegeräte angeschlossen werden. Mit dem eingebauten Schalter wird zwischen den Peripheriegeräten umgeschaltet. Es entfällt damit das lästige Umstecken der Peripheriegeräte, wenn abwechselnd eine Computerschnittstelle benutzt werden muß.

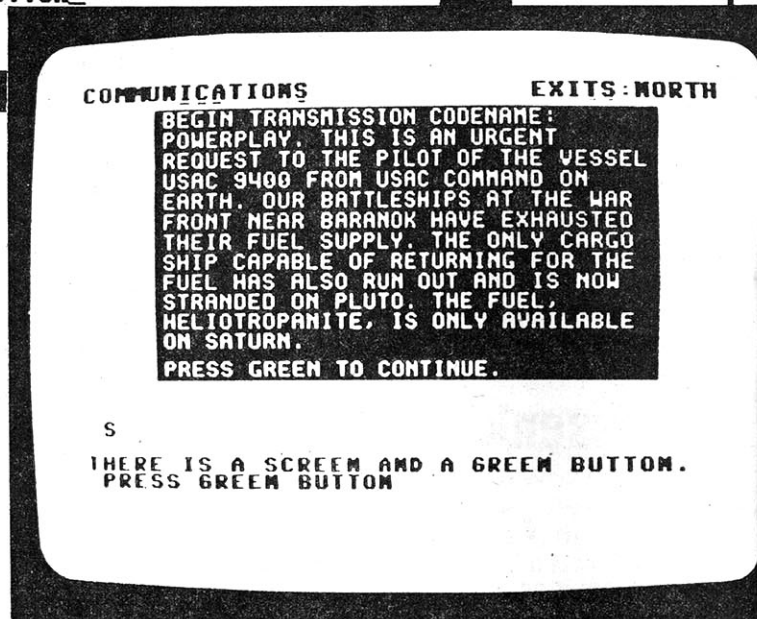
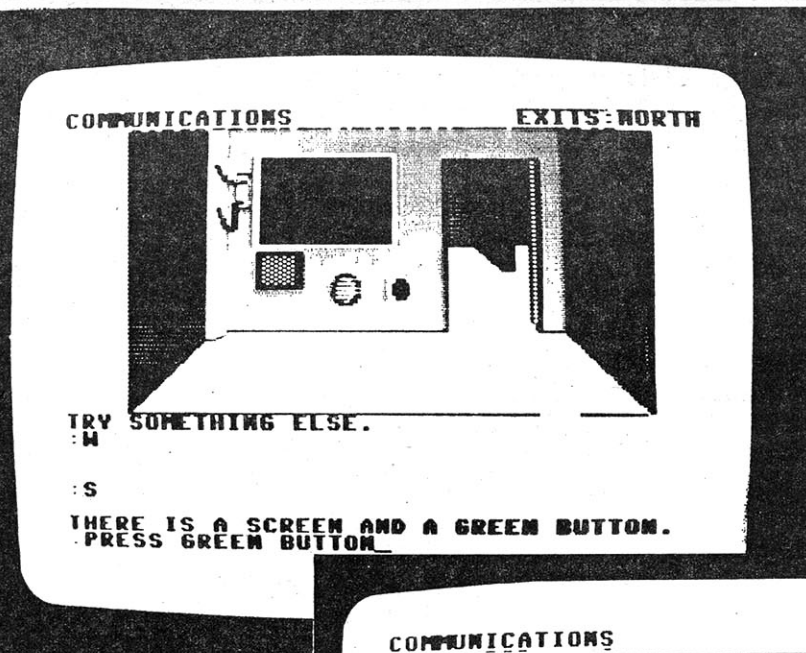


Gruds in Space

Nach "Blade of Blackpool" ein weiteres hervorragendes Adventure von Sirius. Der Spieler findet sich als privater Händler in seinem Handelsraumschiff. Ziel des Adventure ist es; eine Raumflotte mit Treibstoff zu versorgen. Diese sieht sich bedrohlichen Angriffen entgegen und kann wegen Treibstoffmangel nichts unternehmen. Die Bezahlung für diese Aufgabe ist gut, die Reise führt

zunächst zum Planeten Saturn, wo die Gruds leben. Hier können einige nützliche Ausrüstungsgegenstände gefunden werden, die auf dem Weg zum Ziel benötigt werden. Gruds in Space ist ein hervorragendes Adventure mit tollen Ton- und Grafikeffekten, das auch bei vorgegebener Lösung sehr lange zu spielen ist.

(sr)



soft service

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

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

Bestellungen Inland:

Gegen Einsendung eines Schecks oder Vorauszahlung auf unser Konto bei der Raiffeisenbank Eschwege eG, Bankleitzahl 522 603 85 Kto.-Nr. 245 8888 senden wir Ihnen die gewünschten Programme schnellstmöglich zu. Geben Sie bitte unbedingt die ge-

naue Bezeichnung der Kassette (z.B. Compute mit 37/84 C-64) und den Computer an.

Bestellungen Ausland:

Nur Vorkasse, Schein
Keine Schecks oder Überweisungen!

aus Compute mit 42/84

VC-20

K 14,- DM

Jumpin Painter
Kaboom
Kniffel
Light Pen
Gold der Geister

Commodore 64

K 14,- DM

Grazy Car
Horseguard
Simons Neuplot
Mensch ärgere Dich doch
Programm-Verwaltung

Jumping Painter

für den VC-20 + 16K

Inbetriebnahme des Programms:

Vor dem Eintippen und jedem Laden müssen der Zeichensatz und das Maschinenprogramm durch **POKE 642,28: SYS 64818** geschützt werden.

Spielablauf:

Nach dem Starten des Programms erscheint ein Titelbild, das für einige Zeit stehenbleibt. Währenddessen werden sowohl Zeichensatz als auch Maschinenprogramm erstellt. Daraufhin ertönt eine Melodie und es erscheint das Auswahlmenü. Durch Drücken von f1 wird das Spiel gestartet.

Die Spielphasen:

- 1) "Übungsphase"
- 2) In dieser Phase ist die Zeit sehr knapp bemessen, so daß es schwierig ist, allen Rostnagern (Ihren Feinden) aus dem Weg zu gehen.
- 3) Hier steckt die Gefahr wieder in der knappen Zeit und einer großen Fallstange.
- 4) In dieser Phase haben Sie zwar viel Zeit, aber die vielen Fallstangen sind auch nicht einfach!
- 5) Bietet etwas "Erholung", aber auch schwer genug!
- 6) Hier müssen Sie sich von einem Auf-

zug herauf- und herunterfahren lassen. (Besteigen des Aufzuges: Eingangstür betreten und den Joystick kurz nach unten drücken. Aussteigen: Knopf drücken).

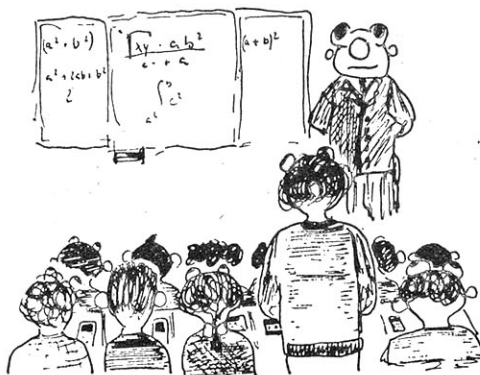
7) Eine einfachere Phase.

8) Die letzte Phase sieht sehr leicht aus, aber die Zeit ist extrem knapp!

Wenn Sie alle Phasen geschafft haben sollten, tritt ein Pfeil auf, der ab und zu einzelne Stücke des Gerüsts wieder entfärbt!

Nachdem Sie aber alle Leben verloren haben, erscheint wieder das Menü.

*Ich konnte meine Mathe-Aufgaben
nicht machen, Herr Lehrer.
Mein Akku-Ladegerät braucht 28 Stunden!*



Ich konnte meine Mathe-Aufgaben
nicht machen, Herr Lehrer.
Mein Akku-Ladegerät braucht 28 Stunden!

programme

```

260 IF(R1AND4)=. THEN400
270 IF(R1AND8)=. THEN450
300 AS=S:S=S+R:Q=Q-R*(ABS(R)=1):GETT$:IFT$="M" THENGOSUB5000:GOTO170:REM F4
310 IFPEEK(S)>127 THEN900
320 POKES1,-160*(ABS(R)=1):POKES+N,C:POKES,Z:C=PEEK(S+N):Z=PEEK(S)
330 POKES+N,1:POKES,64:POKES1,.
340 IFPEEK(S+22)=30 THENPOKES4,200:POKES+22,31:SC=SC+1:K=K-1:POKES4,,:IFK<1 THEN200
350 GOSUB600:ON-(H=6ANDQ=11)GOTO650:IFTM>.GOTO200
360 GOTO700
400 IFZ=660R(Z<32ANDPEEK(S+22)=66) THENR=-22:POKES1,130:GOTO300
410 GOTO300
450 IF(Z=660R(Z<32ANDPEEK(S+22)=660RPEEK(S+44)=66))ANDS<W+462ANDPEEK(S+44)<32 THEN470
460 GOTO300
470 R=22:POKES1,130:GOTO300
500 POKES,65:R=.
510 IF(R1AND16)=. ANDQ>0 THENR=3
520 IF(R2AND128)=. ANDQ<21 THENR=6
540 FORI=0TO2:POKES2,I*35+135:POKES+N,C:POKES,Z:S=S+R(I+R):Q=Q+Q(R)
550 IFQ<00RQ>21 THENQ=Q-Q(R):S=S-R(I+R):POKES2,0:GOTO200
560 C=PEEK(S+N):Z=PEEK(S):POKES+N,1:POKES,64+(IAND1):GOSUB600:IFZ>80ANDZ<87 THENGOSUB800
565 IFPEEK(S+22)<32 THENPOKES2,,:R=. :GOTO300
570 FORT=1TO99:NEXT:NEXT:POKES2,,:GOTO200
600 SYSC:IFRND(1)<0 THENGOSUB850
610 TM=TT-INT(TI/60):IFG>. THENG=0-1:IFG=. THENPOKE254,.
620 PRINT"###"TAB(11-LEN(STR$(SC)))SCCHR$(5-151*(TM>10))TAB(21-LEN(STR$(TM)))TM
630 RETURN
650 IF(PEEK(P1)AND8) THEN200
655 E=0:POKE38339,1
660 FORI=1TO6:GOSUB600:ON-(TM<1)GOTO360:FORT=1TO200:NEXT:IF(PEEK(P1)AND32)=0 THEN680
670 NEXTI:POKE38339-E*110,2:E=E+1AND3:POKE38339-E*110,1:GOTO660
680 Z=32:POKES,Z:S=W+473-E*110:POKES,65:FORI=160TO254:POKES3,I:POKES+N,IAND7
690 NEXT:POKES3,0:POKES+N,1:POKE38339-E*110,2:GOTO200
700 REM KAPUTT
710 FORI=254TO128STEP-1:POKES2,I:POKES+N,IAND7:NEXT:POKES2,0
720 POKES,68:POKES+N,1:LZ=LZ-1:POKES2,0:FORT=1TO999:NEXT
730 IFLZ=>0 THEN190
740 GOSUB5000:GOTO170
750 FORI=46TO0STEP-2:POKE36867,I:T=LOG(pi):NEXT
760 RETURN
770 FORI=0TO46STEP2:POKE36867,I:T=LOG(pi):NEXT
780 RETURN
800 REM GEGENSTAND
810 SC=SC+(87-2)*10:SYS4713
820 FORJ=254TO128STEP-1:POKES2,J:NEXT:FORJ=129TO254:POKES2,J:NEXT:POKES2,0
830 Z=32:G=20:POKE254,2:RETURN
850 REM ENTFERBUNG !
860 P=W+FNA(22)+22:Z1=PEEK(P):C1=PEEK(P+N):T=250:TT=TT+5
870 POKES2,T:POKEP,Z1:POKEP+N,C1:P=P+22:IFP>W+505 THENPOKES2,0:RETURN
880 Z1=PEEK(P):C1=PEEK(P+N):POKEP,69:POKEP+N,1:POKES+N,(T)AND7:IFZ1=31 THENZ1=30:K=K+1
890 T=T-2:GOTO870
900 POKES3,0:POKES2,0:POKES1,0:POKES+N,C:POKES,Z:IFG THEN930
920 POKES,32:POKES-1,32:POKES+1,32:FORI=254TO128STEP-.5:POKES4,I:NEXT:POKES4,0:GOTO700
930 POKES4,200:FORI=4652TO4683STEP4:X1=PEEK(I+2)/8:Y1=PEEK(I+3)
940 W1=W+X1+22*Y1:IFABS(S-W1)<2 THEN960
950 NEXT:POKES4,0:GOTO200
960 POKEI,0:POKES-1,32:POKES,64:POKES+N,1:POKES+1,32:Z=32
970 FORI=160TO254:POKES4,I:NEXT:POKES4,0
980 SC=SC+30:GOTO200
1000 REM BILDSCHIRM
1001 REM "I"=SHIFT B
1002 REM "-"=SHIFT C
1005 GOSUB750
1010 POKE36879,10+FNA(3)*2:POKEV,16*(FNA(14)+2)+10:POKE36869,205:GOSUB9150:POKE254,0
1020 ONHGOSUB1100,1200,1300,1400,1500,1600,1700,1800,1900
1030 FORI=W+484TOW+505:POKEI+N,15:POKEI,30:NEXT
1035 GOSUB770
1040 Z=32:C=1:AS=S:A=0:G=0:TT=TT-DX*2

```

```

1050 PRINT "SC: "; ZEIT:="TI$="000000"
1060 IFLX>0 THEN FORI=1 TO LX:POKEW+I-1,64:NEXT
1070 X=4652:FORI=1 TO LEN(A$)/6:POKE X,2+FNA(6):POKE X+1,FNA(2)
1080 POKE X+2,VAL(A$)*8:A$=MID$(A$,4):POKE X+3,VAL(A$):A$=MID$(A$,4):X=X+4:NEXT
1090 RETURN
1100 PRINT "X" SPC(18)"X":POKE F,15:PRINT "X"
1110 PRINT "I":PRINT "I":PRINT "I"
1120 FORI=1 TO 2:PRINT "I" SPC(8)"I":NEXT:PRINT "SPC(8)"
1130 PRINT "I" SPC(18)"I":PRINT "I" SPC(16)"X ":POKE F,15:PRINT "I":PRINT "I" SPC(18)"I"
1140 PRINT "I" X X ":POKE F,15:PRINT "I":PRINT "I"
1150 PRINT "I" "I":PRINT "I" "I":PRINT "I" "I":PRINT "I"
1160 FORI=1 TO 2:PRINT "I" SPC(18)"I":NEXT
1170 S=W+469:Q=7:K=104:TT=80:A$="10,15"
1180 RETURN
1200 POKE F,5:PRINT "X" SPC(18)"X":POKE F,15:PRINT "X"
1210 FORI=1 TO 3:PRINT SPC(10)"I":NEXT:PRINT "I"
1220 FORI=1 TO 2:PRINT "I" SPC(7)"I" SPC(8)"I":NEXT
1230 PRINT "I" SPC(6)"I" SPC(6)"I":PRINT "I"
1240 PRINT SPC(5)"I" SPC(7)"I":PRINT SPC(7)"I":PRINT "I"
1250 PRINT SPC(7)"I" SPC(8)"I":POKE F,15:PRINT SPC(10)"I":PRINT "I"
1260 PRINT SPC(10)"I":PRINT SPC(10)"I":PRINT "I" SPC(5)"I" SPC(6)"I"
1270 FORI=1 TO 2:PRINT "I" SPC(7)"I" SPC(8)"I":NEXT
1280 S=W+469:Q=7:K=104:TT=95:A$="10,02:03,15:19,15:03,06:19,06:"
1290 RETURN
1300 PRINT "I" SPC(9)"X":POKE F,15:PRINT "I"
1310 FORI=1 TO 2:PRINT "I" "I":NEXT:PRINT "I" "I"
1320 FORI=1 TO 2:PRINT "I" "I" SPC(13)"I":NEXT:PRINT "I" "I"
1330 PRINT "I" "I" SPC(9)"I":POKE F,15:PRINT "I" "I"
1340 PRINT "I" "I" SPC(13)"I":FORI=1 TO 3:PRINT "I" "I":NEXT
1350 PRINT "I" "I" SPC(5)"I" SPC(6)"I":POKE F,15
1360 PRINT "I" "I" SPC(5)"I":PRINT "I" "I"
1370 PRINT "X":POKE F,15:PRINT "I" SPC(19)"X":POKE F,15:PRINT "I"
1380 S=W+483:Q=21:K=82:TT=95:A$="03,02:16,05:11,08:17,11:12,15:18,18:"
1390 RETURN
1400 PRINT "I" SPC(18)"I":POKE F,13:PRINT "I"
1410 FORI=1 TO 6:PRINT "I" "I" "I" "I" "I" "I":NEXT
1420 PRINT "I" "I" "I" "I" "I" "I":PRINT "I" "I" "I" SPC(10):POKE F,15:PRINT "I";
1430 POKE F,13:PRINT "I" "I" "I" "I" "I" "I":PRINT "I" "I" "I" "I" "I" "I"
1440 FORI=1 TO 2:PRINT "I" "I" SPC(15)"I":NEXT:PRINT "I" "I" "I" "I" "I" "I";
1450 PRINT "I" "I":POKE F,13:PRINT "I" "I":POKE F,13:PRINT "I" SPC(6)"I"
1460 PRINT "I" "I" SPC(7)"I" SPC(6)"I":PRINT "I" "I" "I" "I" "I" "I";
1470 FORI=1 TO 2:PRINT "I" SPC(16)"I":NEXT
1480 S=W+473:Q=11:K=100:TT=170:A$="17,09:12,12:09,15:17,18:"
1490 RETURN
1500 POKE F,15:PRINT "I" SPC(21)"I":PRINT "I" "I" SPC(14)"I";
1510 PRINT "I" SPC(9)"I" "I":PRINT "I" "I" "I" "I" "I" "I":PRINT "I" "I"
1520 PRINT "I" "I" "I" "I" "I" "I":PRINT "I" "I" "I" "I" "I" "I"
1530 POKE F,15:PRINT "I" "I":PRINT "I" "I" "I" "I" "I" "I":PRINT "I" "I" "I" "I" "I" "I":POKE F,15
1540 PRINT "I" SPC(8)"I":PRINT "I" SPC(8)"I":PRINT "I" "I" "I" "I" "I" "I"
1550 PRINT "I" SPC(15)"I":PRINT "I" SPC(12)"I":PRINT "I" "I" "I" "I" "I" "I"
1560 PRINT "I" SPC(17)"I":POKE F,15:PRINT "I" "I"
1570 S=W+483:Q=21:K=102:TT=105:A$="03,02:19,03:14,04:10,05:16,11:14,18:"
1580 RETURN
1600 PRINT "I" SPC(15)"I":POKE F,11:PRINT "I" "I" "I" "I" "I" "I" SPC(9)"I"
1610 FORI=1 TO 2:PRINT "I" SPC(4)"I" SPC(9)"I":NEXT:PRINT "I" SPC(4)"I"
1620 FORI=1 TO 2:PRINT "I" SPC(5)"I" SPC(7)"I":NEXT:PRINT "I" "I" "I" "I" "I" "I" SPC(5):POKE F,11
1630 PRINT "I" "I" SPC(7)"I" SPC(7)"I":PRINT SPC(7)"I" "I" "I" "I" "I" "I"
1640 PRINT "I" "I" SPC(15)"I":PRINT "I" SPC(13)"I":POKE F,11
1650 PRINT "I" SPC(9)"I" "I":PRINT SPC(6)"I" SPC(11)"I":PRINT "I" "I" "I" "I" "I" "I"
1660 PRINT "I" SPC(13)"I":PRINT "I" SPC(14)"I":PRINT "I" "I" "I" SPC(10)"I"
1670 PRINT "I" SPC(15)"I":L$="I"
1680 PRINT "I" SPC(10)L$ "I" L$ "I" L$ "I" L$ "I"
1685 E=51:FORI=W+121 TO W+451 STEP 110:POKEI+I,N,2:POKEI,E:E=E-1:NEXT:POKEW+451,5
1690 S=W+483:Q=21:K=90:TT=110:A$="04,03:18,03:04,09:19,11:01,16:20,18:06,19:"
1695 RETURN
1700 PRINT "I" "I":POKE F,12:PRINT "I" "I":PRINT "I" SPC(6)"I":PRINT "I" SPC(10)"I"

```


[illegible]

```

3330 PRINT "MIES FOLGT EINE PUNKTE-MLISTE:";GOSUB3500
3340 POKE36879,14:PRINT "M ↑ #=> # = 1 PUNKT"
3350 PRINT "M # = 10 PUNKTE"
3360 PRINT "M # = 20 PUNKTE"
3370 PRINT "M # = 30 PUNKTE"
3380 POKEF,13:PRINT "M # = 40 PUNKTE"
3390 PRINT "M # = 50 PUNKTE"
3400 PRINT "M # = 60 PUNKTE"
3410 PRINT "M # = 30 PUNKTE"
3500 PRINT "M - BITTE EINE TASTE -M"
3510 IF PEEK(197)=64 AND PEEK(653)=0 AND (PEEK(P1) AND 32)=32 THEN 3510
3520 POKES3,220:FOR T=1 TO 20:NEXT:POKES3,0:RETURN
3600 GOSUB3000:POKE36869,205:L%=L1%
3610 PRINT "M F1 = SPIELSTART"
3620 PRINT "M F3 = ANLEITUNG"
3630 PRINT "M F5 = HIGHSCORE"
3640 PRINT "M F7 = LEBEN: #L%+1"
3642 PRINT "M F2 = LADE HIGH."
3645 PRINT "M F4 = SPIELENDEN":POKE198,0
3650 GETA$:IFA$=" "OR (PEEK(P1) AND 32)=0 THEN RETURN:REM F1
3660 IFA$=" "THEN GOSUB3200:GOTO3610:REM F3
3670 IFA$=" "THEN GOSUB4000:GOTO3610:REM F5
3680 IFA$=" "THEN L%=L%+1 AND 3:PRINT "M "SPC(14)L%+1:L1%=L%:REM F7
3685 IFA$=" "THEN GOSUB8100:GOTO3610:REM F2
3687 IFA$=" "THEN GOSUB8000:POKE36879,27:POKE36869,192:PRINT "M ";END:REM F4
3690 GOTO3650
4000 REM HIGHSCORE
4010 POKE36879,8
4020 PRINT "M HIGHSCORETABELLE"
4030 PRINT "M *****"
4040 FOR I=1 TO 10:PRINT:PRINT SPC(3-LEN(STR$(I)))"M I"IL. #";
4050 PRINT H$(I)TAB(21-LEN(STR$(H(I))))"M H(I);
4060 NEXT:GOTO3510
5000 REM EINTRAGEN
5010 GOSUB750:PRINT "M":IF SC<=H(10) THEN 770
5020 POKE36879,10:PRINT "M SIE HABEN EINEN DER M VORDEREN PLAEZTE ER- M REIC-
HT !"
5030 PRINT "M TRAGEN SIE NUN BITTE M IHREN NAMEN EIN:M":GOSUB770
5040 PRINT "M => -----":L=0:N$="":POKE198,0
5050 FOR I=1 TO 6 STEP 5
5060 PRINT "M "SPC(6)N$:POKEF,I:PRINT "-":GOSUB5500
5070 IFT$=CHR$(20) THEN PRINT "M":GOTO5040
5080 IFT$=CHR$(13) THEN 5200
5090 IFL>9 THEN 5200
5100 N$=N$+T$:L=LEN(N$):NEXT I:GOTO5050
5200 FOR I=1 TO 10:IF SC>H(I) THEN GOSUB5300:I=10
5210 NEXT I:RETURN
5300 FOR J=10 TO 1 STEP -1:H(J)=H(J-1):H$(J)=H$(J-1):NEXT
5310 H(I)=SC:H$(I)=N$:RETURN
5500 FOR T=1 TO 20:GETT$:IFT$=>"A" AND T$<="Z" OR T$=CHR$(13) OR T$=CHR$(20) OR T$=" " THEN T.
=99
5510 NEXT T:RETURN
6000 REM COPYRIGHT
6010 POKE36869,192:POKE36879,10:PRINT CHR$(8)CHR$(142)RND(-TI)
6020 PRINT "M "SPC(4)"/////////////////"
6030 PRINT SPC(4)"M S.R. SOFTWARE"
6040 PRINT "M PRASSENTIERT:"
6050 PRINT "M "
6060 PRINT "M JUMPING PAINTER"
6070 PRINT "M "
6080 PRINT "M COPYRIGHT JULI 1984"
6090 PRINT "M BY STEFAN REUSCH":RETURN
6999 END
7000 REM SPIELE MUSIK
7010 POKEV,PEEK(V) AND 240 OR 15
7020 FOR I=0 TO 42:POKES3,N%(I):POKES2,N%(I)
7030 IF H=9 THEN FOR T=1 TO D(I)*4:GOSUB2040:NEXT:GOTO7050
7040 FOR T=1 TO D(I)*350:NEXT
7050 POKES3,0:POKES2,0:FOR T=1 TO 30:NEXT:NEXT
7060 RETURN
8000 REM SAVE HIGH.
8010 PRINT "M 'RECORD' & 'PLAY'
8020 PRINT "M DRUECKEN."
8030 WAIT 37151,64,64:PRINT "M OK.":OPEN 1,1,1,"HI.SC."
8040 FOR I=1 TO 10:PRINT #1,H(I):PRINT #1,H$(I):NEXT

```


programme

```

8050 CLOSE1:RETURN
8100 REM LOAD HIGH.
8110 PRINT"PROGRAMM 'PLAY' DRUECKEN."
8120 WAIT37151,64,64:PRINT"OK.":OPEN1,1,0,"HI.SC.":PRINT"GEFUNDEN."
8130 FORI=1TO10:INPUT#1,H(I):INPUT#1,H*(I):NEXT
8140 CLOSE1:RETURN
9000 REM SPEICHERINIT
9020 FORI=5120TO5631:READA:POKEI,A:NEXT
9030 READA:IFA<0THEN9060
9040 FORI=0TO7:READB:POKE5120+I+A*8,B:NEXT
9050 GOTO9030
9060 :
9070 READA:IFA=-1THEN9120
9080 FORI=0TO7:READL:R=256*L:IF(RAND2)THENL=L/2:R=R/2
9090 FORJ=0TO3:C=5120+A*8+J*32+I:POKEC,L:POKEC+8,(R/256-INT(R/256))*256
9100 L=INT(L/4):R=INT(R/4):NEXT
9110 NEXT:GOTO9070
9120 :
9130 A=N:FORI=0TO43STEP2:POKE4608+I,RAND255:POKE4609+I,A/256:A=A+22:NEXT
9140 FORI=4684TO4867:READA:POKEI,A:NEXT
9150 FORI=4652TO4683:POKEI,.:NEXT
9160 RETURN
9500 REM MUSIK
9510 DATA195,1,207,1,215,1,225,3,228,3,223,1,219,.5,215,1.5,219,1.5,223,.5,219,1
,215,2
9520 DATA0,1,195,1,201,1,207,1,215,3,219,3,215,1.4,201,.6,201,1,201,3,0,3
9525 DATA215,1,207,1,215
9530 DATA1,225,3,219,3,215,1,207,1,201,1,195,3,201,3,207,1,201,1
9540 DATA207,1,215,3,219,2,0,1,201,1,207,1.4,201,.6,195,3
10000 REM NEUE ZEICHEN
10010 DATA060,102,110,110,096,098,060,000,024,060,102,126,102,102,102,000,124,10
2,102
10020 DATA124,102,102,124,000,060,102,096,096,096,102,060,000,120,108,102,102,10
2,108
10030 DATA120,000,126,096,096,120,096,096,126,000,126,096,096,120,096,096,096,00
0,060
10040 DATA102,096,110,102,102,060,000,102,102,102,126,102,102,102,000,060,024,02
4,024
10050 DATA024,024,060,000,030,012,012,012,012,108,056,000,102,108,120,112,120,10
8,102
10060 DATA000,096,096,096,096,096,096,126,000,099,119,127,107,099,099,099,000,10
2,118
10070 DATA126,126,110,102,102,000,060,102,102,102,102,102,060,000,124,102,102,12
4,096
10080 DATA096,096,000,060,102,102,102,102,060,014,000,124,102,102,124,120,108,10
2,000
10090 DATA060,102,096,060,006,102,060,000,126,024,024,024,024,024,024,000,102,10
2,102
10100 DATA102,102,102,060,000,102,102,102,102,102,060,024,000,099,099,099,107,12
7,119
10110 DATA099,000,102,102,060,024,060,102,102,000,102,102,102,060,024,024,024,00
0,126
10120 DATA006,012,024,048,096,126,000,060,048,048,048,048,048,060,000,012,018,04
8,124
10130 DATA048,098,252,000,060,012,012,012,012,012,060,000,000,024,060,126,024,02
4,024
10140 DATA024,000,016,048,127,127,048,016,000,000,000,000,000,000,000,000,000,02
4,024
10150 DATA024,024,000,000,024,000,102,102,102,000,000,000,000,000,102,102,255,10
2,255
10160 DATA102,102,000,024,062,096,060,006,124,024,000,098,102,012,024,048,102,07
0,000
10170 DATA060,102,060,056,103,102,063,000,006,012,024,000,000,000,000,000,012,02
4,048
10180 DATA048,048,024,012,000,048,024,012,012,012,024,048,000,000,102,060,255,06
0,102
10190 DATA000,000,000,024,024,126,024,024,000,000,000,000,000,000,000,024,024,04
8,000
10200 DATA000,000,126,000,000,000,000,000,000,000,000,000,024,024,000,000,003,00
6,012
10210 DATA024,048,096,000,060,102,110,118,102,102,060,000,024,024,056,024,024,02
4,126
10220 DATA000,060,102,006,012,048,096,126,000,060,102,006,028,006,102,060,000,00
6,014
10230 DATA030,102,127,006,006,000,126,096,124,006,006,102,060,000,060,102,096,12

```

```

4.102
10240 DATA102,060,000,126,102,012,024,024,024,024,000,060,102,102,060,102,102,06
0,000
10250 DATA060,102,102,062,006,102,060,000,000,000,024,000,000,024,000,000,000,00
0,024
10260 DATA000,000,024,024,048,014,024,048,096,048,024,014,000,000,000,126,000,12
6,000
10270 DATA000,000,112,024,012,006,012,024,112,000,060,102,006,012,024,000,024,00
0
10500 REM-----
10510 DATA30,255,15,240,15,240,15,240,15
10520 DATA31,255,255,255,255,255,255,255,255
10530 DATA64,126,219,255,126,36,66,36,231
10540 DATA65,126,219,255,126,36,231,0,0
10550 DATA66,65,105,65,65,65,105,65,65
10560 DATA67,40,40,40,40,40,40,40,40
10570 DATA95,255,255,63,63,15,15,3,3
10580 DATA105,255,255,252,252,240,240,192,192
10590 DATA81,60,102,251,253,253,255,126,60
10600 DATA82,64,64,252,255,252,64,64,0
10610 DATA83,0,20,84,32,32,32,32,32
10620 DATA84,136,136,142,137,142,136,248,112
10630 DATA85,40,68,40,16,40,68,68,68
10640 DATA86,62,127,8,8,8,8,40,16
10650 DATA68,16,56,16,16,16,56,124,254
10660 DATA69,60,24,24,24,90,126,60,24
10670 DATA76,6,6,6,6,6,6,6,6
10680 DATA77,96,96,96,96,96,96,96,96
10690 DATA78,0,7,5,5,5,4,7,6
10700 DATA79,0,255,210,215,211,87,255,0
10710 DATA80,0,224,32,96,96,96,224,96

12010 DATA165,250,10,169,185,0,18,133,251,185,1,18,133,252,165,249,74,74,74,24,1
01,251
12020 DATA133,251,144,2,230,252,96
12030 REM-----
12040 DATA162,28,189,44,18,133,247,208,3,76,250,18,189,45,18,133,248,189,46,18,1
33,249
12050 DATA189,47,18,133,250,32,76,18,169,32,160,0,145,251,200,145,251,165,248,20
8,22
12055 DATA198,249
12060 DATA32,76,18,160,22,177,251,201,32,208,6,169,1,133,248,230,249,184,80,19,2
30,249
12070 DATA32,76,18,160,23,177,251,201,32,208,6,169,0,133,248,198,249,165,248,157
,45,18
12080 DATA165,249,157,46,18,32,76,18,165,249,41,7,10,133,20,165,248,24,101,254,1
0,10,10
12090 DATA10,24,101,20,9,128,160,0,145,251,200,24,105,1,145,251,165,252,24,105,1
32,133
12100 DATA252,160,0,165,247,145,251,200,145,251,234,202,202,202,202,48,3,76,107
12110 DATA18,96

```

WERSIBOARD FÜR COMMODORE C 64/SX 64

MUSIC 64

PERSONAL
COMPUTER
MUSIC

Das WERSIBOARD-SYSTEM MUSIC 64 besteht aus einem Orgel-Manual im Commodore Design, einem Interface-Modul und zugehöriger Software. Gemeinsam mit dem Commodore VC 64 entsteht ein Musikinstrument mit bemerkenswerten Fähigkeiten.

Komplett-Paket bestehend aus:

KEYBOARD

- 49 Tasten, 4 Oktaven C-C
- PROFIFORMAT
- Gehäuse aus stoßfestem Kunststoff im Commodore-Design
- Interface-Modul mit Verbindungskabel, anschlussfertig

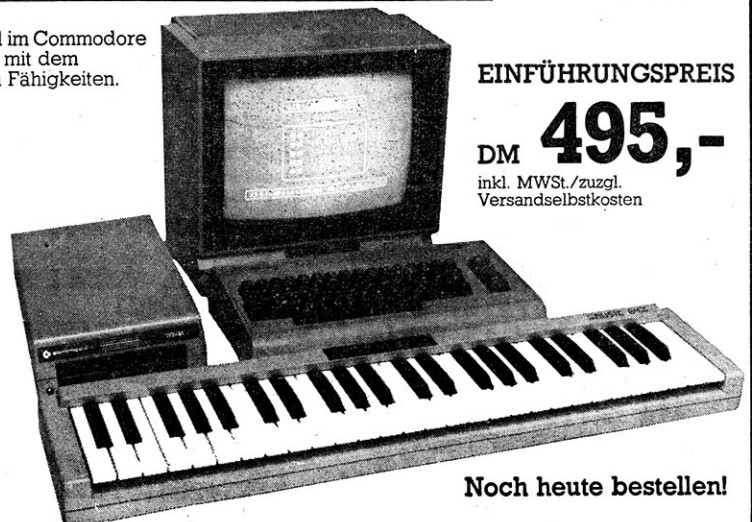
SOFTWARE

- auf 5 1/4" Diskette
- Programm MONO 64 - monophoner Synthesizer
- Programm POLY 64 - polyphones Keyboard
- Klangfarben direkt am PC veränderbar

WERSI Orgel- und Piano-Bausätze · Industriestraße · 5401 Halsenbach
Telefon (06747) 7131 · Telex 42323

BESTELL-COUPON abtrennen und senden an: WERSI electronic, Industriestraße, 5401 Halsenbach, Telefon 06747/7131
Hiermit bestelle ich/wir zu Ihren Allgemeinen Verkaufs-, Lieferungs-, Zahlungs- und Servicebedingungen zur sofortigen Lieferung:
☐ Stück WERSIBOARD MUSIC 64 Einzelpreis DM 495,- Gesamtpreis DM
Zahlung erfolgt: ☐ per Nachnahme ☐ per beigeliegender Scheck ☐ per Vorauskasse
☐ Übersenden Sie mir vor Lieferung die Rechnung ☐ Überweisungsbeleg ist beigelegt

Absender: _____
21 E _____ Ort/Datum _____ Unterschrift _____



EINFÜHRUNGSPREIS

DM **495,-**

inkl. MWSt./zugl. Versandselbstkosten

Noch heute bestellen!

Kaboom

für den VC-20 + 3K

Dieses tolle Grafikspiel entstand in Anlehnung an das gleichnamige bekannte Arcade-Spiel und hat das Vorbild sehr gut nachgeahmt.

Ihre Aufgabe ist es, als Gefängniswärter, Bomben mit einem Wassereimer aufzufangen, die ein Sträfling von der Gefängnismauer wirft.

Das Spiel ist maschinenunterstützt und wird ganz schön schnell.

Die Grafik und der Sound sind sehr gut gelungen.

Eine genaue Bedienungsanleitung ist im Programm enthalten.

```

0 GOTO15
1 "
2 "  KABOOM !!  "
3 "
4 "VC-20+3K+JOYST.
5 "
6 "STEFAN REUSCH
7 "MOEHLENKAMP 22
8 "2352 BORDESHOLM
9 "
10 STEUERUNG ERFOLGT MIT DEM JOYSTICK NACH LINKS UND RECHTS!
15 POKE51,0:POKE52,28:POKE55,0:POKE56,28:POKE251,207:POKE252,29:CLR:GOSUB2000
20 FORI=7631TO7679:POKEI,0:NEXT
30 READA:IFA=-1THEN50
40 FORI=0TO7:READB:POKE7168+A*8+I,B:NEXT:GOTO30
50 L=32:N=30720:M=829:E=253:S=1021:W=8120:V=36874:Q=7432
60 FORI=MTOM+144:READA:POKEI,A:NEXT:READA
70 FORI=7432TO7509:READA:POKEI,A:NEXT:READA
80 GOSUB2500:GOSUB3000
100 POKE36869,240:GOSUB1000:POKE36869,255:POKE36879,88:PRINT"J
110 SZ=0:SC=0:WZ=2:GOSUB5000:Q=.97
120 FORI=7768TO7789:POKEI+N,2:POKEI,9:NEXT
130 FORI=38466TO38487:POKEI,0:NEXT
140 POKES,9:POKES+1,0:POKES+2,9:POKE254,6
200 SYSM:IFRND(1)>QTHENGOSUB510
205 FORT=.TOU:NEXT:SYSO
210 X=PEEK(S)+W:IFPEEK(X)>8ANDPEEK(X+1)>8ANDPEEK(X+2)>8ANDPEEK(E)=.THEN200
220 IFPEEK(E)THEN300
230 POKEV+4,9:POKEV+3,252
232 SC=SC+1:SZ=SZ+1:IFPEEK(X+1)=8THENSZ=SC+1:SZ=SZ+1:POKEV+3,251
235 IFSZ>49THENSZ=SZ-50:IFWZ<3THENWZ=WZ+1:GOSUB500
240 Q=Q*.999:GOSUB5000:FORI=XTOX+2:IFPEEK(I-22)=7THENPOKEI-22,L
250 NEXT:POKEV+3,0:GOTO200
300 X=7724+PEEK(1023):POKEV+N,2:POKEV,16:POKEV+1+N,2:POKEV+1,17:POKEV+22+N,2
302 POKEV+22,18:POKEV+23+N,2:POKEV+23,19
305 FORI=10TO0STEP-.5:POKE36879,89:POKEV+4,1:POKEV+3,220
310 FORT=1TO50:NEXT:POKE36879,88:FORT=1TO50:NEXT:NEXT:POKEV+3,0
315 POKEV+N,1:POKEV,0:POKEV+1+N,1:POKEV+1,1:POKEV+22+N,0
317 POKEV+22,2:POKEV+23+N,0:POKEV+23,3
320 FORI=7790TO8163:POKEI,L:NEXT
330 WZ=WZ-1:IFWZ=>0THENGOSUB5000:GOTO200
340 PRINT"
345 IFSC=HITHENPRINT" *** SPIEL VORBEI ***":GOTO400
350 HI=SC:PRINT"  NEUER HIGHSCORE!
360 POKEV+4,15:FORI=1TO30:FORJ=250TO240STEP-1:POKEV+2,J:NEXT
370 FORJ=240TO250:POKEV+2,J:NEXT:POKEV+2,0:NEXT
400 PRINT"NOCH EIN SPIEL (J/N) ?":GOSUB5000:POKE198,0
410 GETT$:IFT$="J"THEN100
420 IFT$<>"N"THEN410
430 PRINT"TSCHUESS...":POKE36879,27:POKE36869,240:END
500 POKEV+3,0:POKEV+4,15:FORI=254TO128STEP-1:POKEV,I:NEXT:POKEV,0:RETURN

```

```

510 POKEV+4,15:POKEV,130:X=7790+PEEK(S+2)+RND(1)*2:POKEV+N,7:POKEV,7
520 POKEV+22+N,:POKEV+22,8:POKEV+1,RND(1)*2:POKEV,:RETURN
1000 POKE36879,13:PRINT"KABOOM",SPC(6)"KABOOM"
1010 PRINT"KABOOM - SCHNELLER ALS DER":FORT=1T0999:NEXT
1020 POKEV+3,200:FORI=15T00STEP-.1:POKEV+4,I:NEXT:POKEV+3,0
1030 PRINTSPC(7)"KABOOM!"
1040 PRINT"KABOOM (C) BY STEFAN REUSCH"
1050 PRINT"KABOOM HIGHSCORE:HI"
1060 PRINT"KABOOM LEVEL (1-3) ? ":POKE198,0:POKE204,0
1070 GETT$:T=VAL(T$):IFT<10T>3THEN1070
1080 POKE204,1:PRINTT$:U=60-T*20
1090 POKEV+4,9:FORI=128T0254:POKEV+1,I:NEXT:POKEV+1,0:RETURN
2000 PRINTCHR$(8)CHR$(142)RND(-TI):POKE36879,10
2005 PRINT"KABOOM ////////////////":PRINT"KABOOM S.R. SOFTWARE"
2010 PRINT"KABOOM PRÄSENTIERT":PRINT"KABOOM"
2020 PRINT"KABOOM *** KABOOM ***":PRINT"KABOOM"
2030 PRINT"KABOOM COPYRIGHT APRIL 1984":PRINT"KABOOM BY STEFAN REUSCH":RETURN
2500 POKE36878,6
2510 READX,Y:IFX=-1THENPOKE36875,0:RETURN
2515 IFX=-2THENPOKE36878,10:GOTO2510
2520 POKE36875,X:FORT=1T0270*Y:NEXT:POKE36875,0:GOTO2510
3000 POKE36869,255:POKE36879,110
3010 PRINT"KABOOM SPC(5)"KABOOM*",SPC(5)"KABOOM"
3020 PRINT"KABOOM IN DIESEM INTERESSANTEN SPIEL MUßEN SIE
3030 PRINT"KABOOM MIT IHRER WASSERWANNE (KABOOM) BOMBEN AUFFANGEN,
3040 PRINT"KABOOM DIE EIN STRICKFLING ÜBER DIE GEFÄHRLICHE MAUER
3050 PRINT"KABOOM WIRFT":PRINT"KABOOM WIRFT SICH JA GANZ EINFACH AN, ABER...
3060 GOSUB3500:POKE36879,42:PRINT"KABOOM PUNKTEZÄHLUNG:",
3070 PRINT"KABOOM BOMBE AUFGEFANGEN: 1P.":PRINT"KABOOM BOMBE IN DER MITTE DER
3080 PRINT"KABOOM WANNE AUFGEFANGEN: 2P.
3090 PRINT"KABOOM SIE HABEN ZU BEGINN 3 WANNEN ZUR VERFÜGUNG,
3100 PRINT"KABOOM ALLE 50P. GIBT ES EINERDAZU.
3500 PRINT"KABOOM - TASTE DRÜCKEN -":POKE198,0:WAIT198,1:RETURN
5000 PRINT"KABOOM KLMNO"RIGHT$(("00"+MID$(STR$(SC),2),3)" KLMNO")
5010 PRINTRIGHT$(("00"+MID$(STR$(HI),2),3)" " IFWXC=0THEN5030
5020 FORI=1TOW:POKE7702-I*2,10:NEXT
5030 RETURN
9000 DATA0,3,7,13,13,15,11,12,7
9010 DATA1,192,224,176,176,240,208,48,224
9020 DATA2,3,15,63,63,55,55,55,55
9030 DATA3,192,240,252,252,236,236,236,236
9040 DATA4,128,192,224,112,127,58,31,0
9050 DATA5,0,0,0,0,255,165,255,0
9060 DATA6,1,3,7,14,254,92,248,0
9070 DATA7,0,4,0,21,4,8,8,16
9080 DATA8,24,60,126,126,126,126,60,24
9090 DATA9,254,254,254,0,239,239,239,0
9100 DATA10,0,0,0,129,255,126,0,0
9110 DATA11,0,42,42,58,42,42,0,0
9115 DATA12,0,234,138,238,170,234,0,0
9120 DATA13,0,238,136,232,40,238,0,0
9130 DATA14,0,238,170,174,172,234,0,0
9140 DATA15,0,224,128,192,128,224,0,0
9150 DATA32,0,0,0,0,0,0,0,0
9160 DATA48,0,126,70,70,70,126,0,0
9170 DATA49,0,6,6,6,6,6,0,0
9180 DATA50,0,126,6,126,96,126,0,0
9190 DATA51,0,126,6,62,6,126,0,0
9200 DATA52,0,76,76,126,12,12,0,0
9210 DATA53,0,126,96,126,6,126,0,0
9220 DATA54,0,126,64,126,70,126,0,0
9230 DATA55,0,126,6,6,6,6,0,0
9240 DATA56,0,126,70,126,70,126,0,0
9250 DATA57,0,126,70,126,6,126,0,0
9260 DATA16,3,103,109,109,111,107,108,103
9270 DATA17,192,230,182,182,246,214,54,230
9280 DATA18,115,63,31,7,7,7,7,7
9290 DATA19,206,252,248,224,224,224,224,224
9300 DATA29,90,36,66,126,66,66,66,0
9310 DATA30,90,36,66,66,66,36,24,0
9320 DATA31,66,0,66,66,66,66,60,0,-1
9500 DATA174,255,3,169,32,157,44,30,157,45,30,157,66,30,157,67,30
9510 DATA173,254,3,208,11,224,0,208,6,160,1,140,254,3,36,202,201,1,208,11
9520 DATA224,20,208,6,160,0,140,254,3,36,232,142,255,3

```



```

9530 DATA169,0,157,44,30,169,1,157,45,30,169,2,157,66,30,169,3,157,67,30
9540 DATA162,205,189,0,151,157,22,151,189,0,31,157,22,31,202,208,241
9550 DATA162,147,189,109,150,157,131,150,189,109,30,157,131,30,202,208,241
9560 DATA162,22,169,32,157,109,30,202,208,250,32,8,29
9570 DATA162,22,169,0,133,253,189,205,31,201,7,208,10
9580 DATA169,1,133,253,169,32,157,183,31,96,202,208,236,96,-1
9600 DATA174,253,3,169,32,157,206,31,157,207,31,157,208,31
9610 DATA173,31,145,41,16,208,5,224,0,240,1,202,169,127,141,34,145,173,32,145
9620 DATA41,128,208,5,224,19,240,1,232,142,253,3,169,255,141,34,145
9630 DATA165,254,157,206,151,157,207,151,157,208,151,169,4,157,206,31
9640 DATA169,5,157,207,31,169,6,157,208,31,96,-1
10000 DATA195,.5,201,.5,207,1,215,1,215,1,219,1,215,1,207,1,195,1,195,.5,201,.5
10010 DATA207,1,207,1,201,1,195,1,201,3,195,.5,201,.5
10020 DATA207,1,215,1,215,1,219,1,215,1,207,1,195,1,195,.5,201,.5
10030 DATA207,1,207,1,201,1,201,1,195,3,195,.5,201,.5
10040 DATA207,1,215,1,215,1,219,1,215,1,207,1,195,1,195,.5,201,.5
10050 DATA207,1,207,1,201,1,195,1,201,3,195,.5,201,.5
10060 DATA207,1,215,1,215,1,219,1,215,1,207,1,195,1,195,.5,201,.5
10070 DATA207,1,207,1,201,1,201,1,195,3,0,1,-2,
10080 DATA209,2,209,2,219,1,219,2,219,1,215,1,215,1,207,1,195,1,201,3
10090 DATA195,.5,201,.5,207,1,215,1,215,1,219,1,215,1,207,1,195,1,195,.5,201,.5
10100 DATA207,1,207,1,201,1,195,1,201,3,0,1
10110 DATA209,2,209,2,219,1,219,2,219,1,215,1,215,1,207,1,195,1,201,3
10120 DATA195,.5,201,.5,207,1,215,1,215,1,219,1,215,1,207,1,195,1,195,.5,201,.5
10130 DATA207,1,207,1,201,1,201,1,195,3,-1,

```

Kniffel

für den VC-20 + 16K

"Kniffel" wurde schon von den Landsern im Ersten Weltkrieg, unter dem Namen "Hindenburgspiel", geschätzt. Sehr bekannt gewesen ist es vor allem in den zwanziger und dreißiger Jahren in Skandinavien unter dem Namen "Yahtsee".

Nachdem Anzahl und Namen der Spieler eingegeben worden sind, ruft der Computer automatisch für jeden Spieldurchgang die Namen auf.

Jeder Spieler hat maximal 3 Würfe mit bis zu 5 Würfeln pro Durchgang und muß versuchen, eine für ihn sinnvolle Zahlenkombination zu erreichen.

Hat man 3 Würfe verbraucht oder schon vorher eine geeignete Zahlenkombination erreicht, kann man nun die Punktzahl für diese Zahlenreihe durch Drücken der davorstehenden Taste eingetragen und gespeichert werden, wobei die gewürfelten Zahlen für die Kombination G-L noch geordnet werden müssen. Hierzu muß die Taste "Q" gedrückt werden und

die Zahlen wie folgt geordnet (mit Hilfe der Tasten 1-5; s.o.):

G-3er-Pasch: 55514 xxxyz
H-4er-Pasch: 333336 xxxxy
I-Full House: 66622 xxxxy
K-Straße (kl.): 23453 uvwxw
L-Straße (gr.): 12345 vwxyz

Falls die gewürfelte Zahlenkombination nicht in die Liste eingetragen werden kann, muß ein beliebiges, noch freies Zahlenfeld der Liste gestrichen werden (d.h. Punktzahl = 0). Dafür braucht nur die Taste "S" gedrückt werden, und anschließend die Taste des zu streichenden

Zahlenfeldes.

Nachdem der Spieler seine Zahlenkombination eingetragen hat, muß er die RETURN-Taste drücken, und der Computer ruft den nächsten Spieler auf.

Wenn alle 13 Durchgänge vorüber sind, zeigt der Computer automatisch die Punkte aller Spiele an und erstellt die Siegertabelle.

Falls ein Spieler zum zweitenmal einen KNIFFEL gewürfelt hat, muß er diesen und alle folgenden in der Zeile 0-Bonus eintragen, wofür er 100 Punkte bekommt. Dieser Bonus ist nur ein Zusatz und zählt nicht als eigener Durchgang.

```

0 REM VOR DEM EINGEBEN BITTE: POKE44,32:POKE8192,0:NEW
1 REM****KNIFFEL****
4 REM*****
5 POKE36879,110:POKE36869,205:POKE36878,10
6 PRINT"U":FORT=828T0881:READY:POKET,Y:NEXTT:SYS828
7 READY:IFY=-1THEN10
8 FORT=0T07:READU:POKET+5120+Y*8,U:NEXTT
9 GOTO7
10 PRINT"KNIFFEL"
11 PRINT" (C) 1984 BY"
12 PRINT"OLAF PANZ"
13 PRINT"ANLEITUNG ?"
20 GOSUB100
21 IFA$="J"THEN23
22 GOTO300
23 PRINT"KNIFFEL"
24 PRINT"KNIFFEL WIRD MIT 5 WUERFELN GESPIELT.SIE KOENNEN 3 MAL WUERFELN"
25 PRINT"VERSUCHEN SIE VERSCHIEDENE ZAHLENKOMBINATIONEN ZU BEKOMMEN."
26 GOSUB100
27 PRINT"KLEINE STRASSE IST EINE FOLGE VON 4 WUERFEN MIT AUFEINANDERFOLGENDEN"
28 PRINT"ZAUGENZAHL,Z.B.: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100":GOSUB100
29 PRINT" GROSSE STRASSE":PRINT"IST EINE FOLGE VON 5 WUERFEN MIT AUFEINANDERFOLGENDER"
30 PRINT"ZAUGENZAHL,Z.B.:"PRINT"1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100":GOSUB100
31 PRINT" OBERE GEWINNERKARTE":PRINT"HIER BEFINDEN SICH KAESTCHEN FUER EINER-"
32 PRINT"ZWEIER-,DREIER-,U.S.W.ZAUGEN. WENN EIN SPIELER HIER EINTRAEGT,AD-"
33 PRINT"DIERT ER NUR DIE WUERFEL MIT GLEICHER PUNKTZAHL,UND"
34 PRINT"TRAEGT ES IN DAS ENTSPRECHENDE KAESTCHEN EIN":GOSUB100
35 PRINT" WIRFT Z.B. EIN SPIELER":PRINT"1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100":PRINT"TRAEGT ER BEIM ZWEIER"
36 PRINT"DIE 8 EIN. WENN SIE AUF DER OBEREN GEWINN-KARTE 63 ODER"
37 PRINT"MEHR PUNKTE HABEN,BEKOMMEN SIE 35 PUNKTE BONUS":GOSUB100
38 PRINT"KNIFFEL":PRINT"HEISST DER WURF MIT 5 GLEICHEN ZAHLEN BELIBIGER ART,ER ERGIEBT"
39 PRINT"0 PUNKTE IM KAESTCHEN FUER KNIFFEL":GOSUB100
40 PRINT"CHANCE":PRINT"GIBT DEM SPIELER DIE MOEGLICHKEIT,BEI EINEM BELIBIGEN SPIEL-"
41 PRINT"GANG EINEN GEWINN ANZUSCHREIBEN,DER KEINER ANDERE BEDINGUNG ERFUELLT."
42 PRINT"ES WIRD DIE SUMME DER PUNKTZAHLN DER WUERFEL EINGETRAGEN":GOSUB100
43 PRINT"DREIERPASCH UND VIERERPASCH":PRINT"3 B.Z.W. 4 GLEICHE ZAHLEN."
44 PRINT"ES WIRD DIE SUMME ALLER ZAHLEN DER WUERFEL EINGETRAGEN":GOSUB100
45 PRINT"FULL HOUSE":PRINT"KANN MAN WAELLEN,WENN EINE BELIBIGE KOMBINATION VON 3 GLEICHEN UND 2 GLEICHEN ZAHLEN EINER ART GEWUERFELT WERDEN."
46 PRINT"FULL HOUSE GIBT 25 PUNKTE":GOSUB100:GOTO51
47 PRINT"ALLE KLAR?":GOSUB100
48 IFA$="N"THEN23
49 GOTO300
50 GOTO300
51 PRINT" WEGSTREICHEN":PRINT"WENN SIE EINE WUERFEL KOMBINATION HABEN,DIE IN KEINE DER GENANNTEN SPALTEN PASST,MUESSEN SIE EINE SPALTE WEGSTREICHEN."
52 PRINT"DAZU DRUECKEN SIE BEI DER FRAGE NACH DER SPALTE EINFACH RETURN."
53 PRINT"DANN GEBEN SIE BITTE DIE SPALTE EIN,DIE SIE WEGSTREICHEN"
54 PRINT"WOLLEN":GOSUB100
55 PRINT"ZUR COMPUTER VERSION":PRINT"VOLGENDE TASTENBELEGUNG GILT IMMER :
56
57 PRINT" "
58 PRINT" N "
59 PRINT" I "
60 PRINT" F11 = TON AUS"
61 PRINT" I "
62 PRINT" V "
63 PRINT" "
64 PRINT" "
65 PRINT" N "
66 PRINT" I "
67 PRINT" F31 = TON AN"
68 PRINT" I "
69 PRINT" V "
70 PRINT" "
71 GOSUB100
72 GOTO48
100 FORT=0T070:POKE36876,T+130:POKE36874,(70-T)+130:NEXTT:POKE36874,0:POKE36876,0
110 GETA$:IFA$=""THEN110
111 RETURN
200 DATA162,0,189,0,128,157,0,20,189,0,129,157,0,21,189,0,130,157,0,22
201 DATA189,0,131,157,0,23,189,0,132,157,0,24,189,0,133,157,0,25,189,0

```

```

202 DATA134,157,0,26,189,0,135,157,0,27,202,208,205,96
203 DATA65,255,129,129,153,153,129,129,255
204 DATA66,255,129,161,129,129,133,129,255
205 DATA67,255,129,133,145,129,161,129,255
206 DATA68,255,129,165,129,129,165,129,255
207 DATA69,255,165,129,145,129,165,129,255
208 DATA70,255,165,129,165,129,165,129,255
209 DATA256,250,0,79,76,65,70,32,32
210 DATA257,200,0,80,65,78,90,32,32
211 DATA258,150,0,79,76,65,70,32,32
212 DATA259,100,0,80,65,78,90,32,32
213 DATA118,0,0,0,56,68,68,69,69
214 DATA119,68,69,62,0,0,0,0,0
215 DATA120,0,0,0,0,0,128,67,68
216 DATA121,136,80,63,0,0,0,0,0
217 DATA122,0,4,10,10,18,18,18,146
218 DATA123,146,146,148,88,176,126,16,16
219 DATA124,60,66,66,66,124,64,64,67
220 DATA125,0,0,0,0,0,0,0,199
221 DATA126,0,0,0,0,0,0,0,224
222 DATA127,0,0,0,15,17,34,68,136
223 DATA114,68,72,80,80,111,64,64,64
224 DATA115,74,82,82,82,226,0,0,0
225 DATA116,17,18,18,20,25,0,0,0
226 DATA117,16,32,64,128,0,255,0,0
227 DATA128,128,129,130,132,136,144,160,224
228 DATA130,8,8,8,8,9,8,9,9
229 DATA132,0,119,132,4,4,4,4,135
230 DATA134,144,136,132,130,129,128,128,128
231 DATA136,137,73,73,41,41,25,25,9
232 DATA138,4,4,4,4,4,4,4,7
233 DATA129,128,12,12,10,10,9,9,9
234 DATA131,0,14,16,33,33,33,33,115
235 DATA135,8,8,8,8,8,136,72,40
236 DATA137,33,33,33,33,33,33,33,33
237 DATA139,32,32,32,32,32,32,32,191
238 DATA133,0,160,32,32,32,32,32,160
300 DATA-1
301 POKE36879,8: PRINT"XXXXXXXXXXXXXXXXXXXXABCDEFGHIJK"
302 PRINT"      (C) 1984 BY"
303 PRINT"      F  U  I  PRINT  U  I  H  I  "
304 PRINT"      ":POKE198,1:POKE631,131
READY.

```

```

1 POKE36879,8:POKE36869,205:POKE36878,5:PRINT"  "
5 DIMPA(10,50),PO(10,50):FORT=0T09:READQ:PO(T,0)=Q:FOR Y=1TOQ:READPO(T,Y),PA(T,Y)
6 NEXT Y,T:P=3
7 PRINT"XXXXXXXXXXXXXXXXXXXXABCDEFGHIJK"
8 PRINT"      (C) 1984 BY":POKE36879,8
9 H$="XXXXXXXXXXXXXXXXXXXX":U=1:M=9
10 PRINTLEFT$(H$,M)"XXXXXXXXXXXXXXXXXXXX":POKE646,INT(RND(1)*5)+1
11 PRINTLEFT$(H$,25-M)"XXXXXXXXXXXX":POKE646,INT(R
ND(1)*5)+1
12 M=M+U:IFM>160RM<9THENU=-U
13 L=L+1:IFL>50THENL=0:GOSUB1000:GOTO7
14 TY=1:GOSUB100:TY=0
15 IFA$=" "THEN10
16 POKE36879,110
17 GOTO200
100 REM
110 POKE36876,PO(P,M1)
111 GETA$=IFA$<>" "THEN116
112 FOR Y=0TOPA(P,M1):NEXT Y:POKE36876,0:M1=M1+1
113 IFM1>PO(P,0)THENM1=0
114 IFTY=1THENRETURN
115 GOTO110
116 IFA$=CHR$(133)THENPOKE36878,0:GOTO110
117 IFA$=CHR$(134)THENPOKE36878,5:GOTO110
118 POKE36876,0:RETURN
129 END
200 REM
220 PRINT"VIELLE SPIELER(2-9)?"
221 GOSUB100
222 A2=VAL(A$):IFA2<20RA2>9THEN221
223 DIMA(15,A2),NA(A2)
229 FORT=1TOVAL(A$)
230 PRINT"      IHR NAME SPIELER":T:PRINT"
"
231 GOSUB100:IFA$=CHR$(13)THEN234
232 NA$(T)=NA$(T)+A$:PRINT"      ":NA$(T);
233 GOTO231
234 IFLEN(NA$(T))>6THENNA$(T)=" ":GOTO230
235 NEXTT
240 PRINT"      ":FORRU=1TO13:FORPL=1TOA2
241 P=PL:IFP=3THENP=0
250 PRINT"      NAME: ";NA$(PL)

```

```

260 PRINT"  1 1 ER  " I":ABS(A(1,PL))
261 PRINT"  2 2 ER  " I":ABS(A(2,PL))
262 PRINT"  3 3 ER  " I":ABS(A(3,PL))
263 PRINT"  4 4 ER  " I":ABS(A(4,PL))
264 PRINT"  5 5 ER  " I":ABS(A(5,PL))
265 PRINT"  6 6 ER  " I":ABS(A(6,PL))

```

V P C	Völzke Computer Peripherie	V P C
	Eprom-Programmer V128 für C 20, C 64 u. SX-64 für Eproms 2508/16/32 u. 2758/16/32/64/128. Professionelle Ausführung m. komfortabler Treiber-Software auf Kas- sette: DM 249,-	
	Neu: Eprom-Programmer V128-G im Pult-Gehäuse DM 349,-	
	Uniment-C 64-Befehlserweiterung: über 50 zus. Befehle u. Funktionen für Assembler, Centronics-Druckanschl., Graphik, Sprite-, Sound- und Disketten-Anwendung; mit Beispielprogrammen u. ausführlicher Bedienungsanlei- tung DM 99,-	
	Diskette zzgl. DM 7,- UNIMENT-Steckmodul DM 199,-	
	Weiteres aus unserem Programm: — Eprom-Karten und -Löschgerät — 80-Zeichenkarten	
Info gegen Rückporto	Hagen Völzke, Ahornallee 4, 8023 Pullach Versandhandel Tel. 089/7934534	


```

266 PRINT"|"=====|"
267 PRINT"|" GESAMMT      |":A(7,PL)
268 PRINT"|"=====|"
269 PRINT"|" 7 DREIERPASCH |":ABS(A(8,PL))
270 PRINT"|" 8 VIERERPASCH |":ABS(A(9,PL))
271 PRINT"|" 9 FULL-HOUSE  |":ABS(A(10,PL))
272 PRINT"|" A KL. STRASSE |":ABS(A(11,PL))
273 PRINT"|" B GR. STRASSE |":ABS(A(12,PL))
274 PRINT"|" C KNIFFEL     |":ABS(A(13,PL))
275 PRINT"|" D CHANCE      |":ABS(A(14,PL))
276 PRINT"|"=====|"
277 PRINT"|" ENDSUMME      |":A(15,PL):PRINT"|" 1 2 3 4 5":GOTO300
290 T2=INT(RND(1)*6)+2:POKE4580+T1*2,T2+63:TY=1:GOSUB100:TY=0:RETURN
300 FORT=0T05:FORT1=1T05:GOSUB290:FORY=1T030:NEXTY,T1,T:WA=1
301 PRINT"=====NOCH EIN WURF ?":IFWA>2THEN320
302 GOSUB100
303 IFA$="N"THEN320
304 WA=WA+1:T1=0:T=1:PRINT"=====WELCHE WUERFEL? W-WURF";
305 GOSUB100
306 IFA$=CHR$(97)THEN310
307 IFVAL(A$)<10RVAL(A$)>5THEN305
308 T1(T)=VAL(A$):T=T+1:IFT>5THEN310
309 GOTO305
310 PRINT"=====":IF T=1THEN301
311 FORT2=0T07:FORT3=1T07-T1:T1=T1(T3):GOSUB290:FORT4=1T010:NEXTT4,T3,T2:GOTO301
320 REM
321 PRINT"=====KAESTCHEN ?(1-9/A-D)"
322 GOSUB100
323 A1=VAL(A$):IFA$="A"ORA$="B"ORA$="C"ORA$="D"ORA$=CHR$(13)THEN325
324 IFA1<10RA1>9THEN322
325 FORT=1T05:I(T)=(PEEK(4580+T*2))-64:NEXTT
326 ONA1GOTO330,340,350,360,370,380,390,400,410
327 GOTO420
330 IFA(1,PL)<>0THEN321
331 T=1:GOTO333
333 FORT1=1T05
336 IF I(T1)=TTHENA(T,PL)=A(T,PL)+T
337 NEXTT1
338 IFA(T,PL)<TTHENA(T,PL)=0:GOTO321
339 GOTO800
340 IFA(2,PL)<>0THEN321
341 T=2:GOTO333
350 IFA(3,PL)<>0THEN321
351 T=3:GOTO333
360 IFA(4,PL)<>0THEN321 ERROR
361 T=4:GOTO333
370 IFA(5,PL)<>0THEN321
371 T=5:GOTO333
380 IFA(6,PL)<>0THEN321
381 T=6:GOTO333
390 IFA(8,PL)<>0THEN321
391 FORT=1T06:T2=0:FORT1=1T05:IF I(T1)=TTHEN T2=T2+1
392 NEXTT1:IFT2=3THENFORY=1T05:A(8,PL)=A(8,PL)+I(E):NEXTE:GOTO800
393 NEXTT:GOTO321
400 IFA(9,PL)<>0THEN321
401 FORT=1T06:T2=0:FORT1=1T05:IF I(T1)=TTHEN T2=T2+1
402 NEXTT1:IFT2=4THENFORY=1T05:A(9,PL)=A(9,PL)+I(E):NEXTE:GOTO800
403 NEXTT:GOTO321
410 IFA(10,PL)<>0THEN321
411 FORT=1T06:FORY=1T06:T1=0:T2=0:FORY=1T05:IF I(U)=TTHEN T1=T1+1
412 IF I(U)=YTHEN T2=T2+1
413 NEXTU
414 IFT1=2ANDT2=3THENA(10,PL)=25:GOTO800
415 NEXTY,T:GOTO321
420 PRINT"|"A$
421 IFA$="B"THEN450
422 IFA$="C"THEN460
423 IFA$="D"THEN470
424 IFA$=CHR$(13)THEN480
425 IFA$="A"THEN440
440 IFA(11,PL)<>0THEN321
441 FORG=1T04:T2=0:T3=0:T4=0:T5=0:T6=0:FORT=1T06:IF I(T)=GANDT3=0THEN T2=T2+1:T3=1
442 IF I(T)=G+1ANDT4=0THEN T2=T2+10:T4=1
443 IF I(T)=G+2ANDT5=0THEN T2=T2+100:T5=1
444 IF I(T)=G+3ANDT6=0THEN T2=T2+1000:T6=1
446 IFT2=1111THENA(11,PL)=30:GOTO800
447 NEXTT,G:GOTO321
450 IFA(12,PL)<>0THEN321
451 FORG=1T04:T2=0:FORT=1T06:IF I(T)=GTHEN T2=T2+1
452 IF I(T)=G+1THEN T2=T2+10
453 IF I(T)=G+2THEN T2=T2+100
454 IF I(T)=G+3THEN T2=T2+1000
455 IF I(T)=G+4THEN T2=T2+10000
457 IFT2=11111THENA(12,PL)=40:GOTO800
458 NEXTT,G:GOTO321
460 T2=0:IFA(13,PL)<>0THEN321

```

```

461 FORT=1T06:FORT1=1T05:IF1(T1)=TTHENT2=T2+1
462 NEXTT1:IFT2=5THENA(13,PL)=50:GOTO800
463 T2=0:NEXTT:GOTO321
470 IFA(14,PL)<>0THEN321
471 FORT=1T05:A(14,PL)=A(14,PL)+I(T):NEXTT:GOTO800
480 PRINT"*****WEGSTREICHEN *****";FORT=0T010:TY=1:GOSUB100
:NEXTT
481 TY=0:PRINT"*****KAESTCHEN ? (1-9/A-D)"
482 GOSUB100
483 IFA$="A"ORA$="B"ORA$="C"ORA$="D"THEN490
484 A1=VAL(A$):IFA1<10RA1>13THEN482
485 A(A1,PL)=-1:GOTO800
490 IFA$="A"THEN496
491 IFA$="B"THEN498
492 IFA$="C"THEN500
493 A(14,PL)=-1
494 GOTO800
496 A(11,PL)=-1
497 GOTO800
498 A(12,PL)=-1
499 GOTO800
500 A(13,PL)=-1
501 GOTO800
800 A(15,PL)=0:M=0:A(7,PL)=0:FORT=1T06:A(7,PL)=ABS(A(7,PL))+A(T,PL):NEXTT
801 IFA(7,PL)>63THENA(7,PL)=A(7,PL)+35
810 FORT=7T014:A(15,PL)=ABS(A(15,PL))+A(T,PL):NEXTT
820 NEXTPL,RU
830 T1=0:T1$="":FORT=1T0PL-1:IFA(15,T)>A(15,T1)THENT1=T:T1$=NA$(T)
831 NEXTT
832 PRINT"*****IEGER:"T1$:PRINT"*****PUNKTE:";A(15,T1)
833 P=T1:GOSUB100
900 A1=-1:FORT=3T00STEP-1
901 IFPEEK(T*8+7168)+(PEEK(T*8+7169)*256)<A(15,T1)THENA1=T
902 NEXTT
903 IFA1=-1THENRUN
905 T3=7168+A1*8
906 POKET3,A(15,T1)AND255:POKET3+1,INT(A(15,T1)/256)
907 FORT=2T07:POKET+T3,32:NEXTT
908 FORT=2T0LEN(T1$)+1:POKET+T3,ASC(MID$(T1$,T-1,1)):NEXTT
909 GOSUB1000:RUN
1000 REM
1001 POKE36879,8:FORT=0T03
1002 HI(T)=PEEK(T*8+7168)+(PEEK(T*8+7169)*256)
1003 HI$(T)="":FORE=0T05:HI$(T)=HI$(T)+CHR$(PEEK(T*8+E+7170)):NEXTE
1004 NEXTT
1005 PRINT"*****HIGHSCORES*****"
1006 PRINT"*****"
1007 FORT=0T03:PRINT"*****";T+1:" ";HI$(T):" ";HI(T):POKE646,T+1:NEXTT
1008 FORT=0T020:TY=1:GOSUB100:NEXTT:TY=0:POKE36879,110:RETURN
60100 DATA0
60101 DATA201,250,201,500,201,250,209,250,215,250,219,250,219,250,219,750,228,25
0
60102 DATA228,250,228,250,228,500,225,250,219,1500,225,250,228,250,228,250,228,5
00
60103 DATA225,250,228,500,225,250,219,500,219,250,201,500,201,250,209,500,215,25
0
60104 DATA219,500,219,250,219,250,215,500,209,250,201,500,219,250,219,250,219,25
0
60105 DATA215,500,209,250,201,500,0,500
60106 DATA19
60107 DATA225,360,225,360,225,240,228,120,231,360,231,240,228,120,231,240,232,12
0
60108 DATA235,720,240,360,235,360,231,360,225,360,235,240,232,120,231,240,228,12
0
60109 DATA225,480,13
60110 DATA217,400,213,400,223,400,227,200,234,200,230,400,227,200,234,200,230,40
0
60111 DATA223,400,227,400,217,400,213,600,30
60112 DATA236,125,238,125,240,250,241,250,240,125,239,125,238,250,236,250
60113 DATA236,125,236,125,240,250,241,250,240,125,239,125,238,250,236,250
60114 DATA236,125,236,125,237,250,237,150,238,125,238,125,239,250,240,150,236,12
5,236
60115 DATA250,237,250,233,250,235,250,234,500
60116 DATA50
60117 DATA195,125,207,125,215,125,219,125,221,125,219,125,215,125,207,125,195,12
5,207,125
60118 DATA215,125,219,125,221,125,219,125,215,125,207,125,209,125,219,125,225,12
5,228,125
60119 DATA229,125,228,125,225,125,219,125,195,125,207,125,215,125,219,125,221,12
5,219
60120 DATA125,215,125,207,125,215,125,223,125,228,125,231,125,232,125,231,125,22
8
60121 DATA125,223,125,209,125,219,125,225,125,228,125,229,125,228,125,225,125,21
9,125
60122 DATA195,125,207,125

```

```

60123 DATA32
60124 DATA159,300,159,175,159,175,175,175,600
60125 DATA195,175,191,175,183,175,191,175,183,250,183,175,183,175,195,300,195,17
5
60126 DATA201,175,195,175,191,175,183,175,175,250,175,175,191,175
60127 DATA201,175,201,600,209,175,207,175,201,175,207,175,195,175,195,175,195,17
5,183
60128 DATA175,175,1000
60129 DATA22
60130 DATA191,200,191,200,191,200,207,900,223,900
60131 DATA2,200,219,200,217,200,213,200,231,900,223,900
60132 DATA0,200,219,200,217,200,213,200,231,900,223,900
60133 DATA219,200,217,200,219,200,213,1200,0,400
60134 DATA50
60135 DATA183,200,201,300,199,100,201,200,212,200,207,300,201,100,207,200,212,10
0,207,100
60136 DATA201,300,201,100,212,200,219,200,223,600,223,200,219,300,212,100,212,20
0,201,200
60137 DATA207,300,201,100,207,200,212,100,207,100,201,300
60138 DATA191,100,191,200,183,200,201,600,223,200,219,300,212,100,212,200,201,20
0,207,300
60139 DATA201,100,207,200,223,200,219,300,212,100,212,200,219,200,223,600,223,20
0,219
60140 DATA300,212,100,212,200,201,200,207,300
60141 DATA11
60142 DATA201,300,215,600,219,200,223,800,225,600,223,200,219,300,223,600,219,20
0,215,300
60143 DATA201,400,19
60144 DATA215,100,215,300,215,100,225,400,225,400,228,400,228,400,235,600,231,20
0,225,200,
60145 DATA100,235,100,231,300,225,100,219,400,232,800,228,300,223,100,225,800

```

LIGHT PEN

für den VC-20

Der Light-Pen ist auf sehr einfache Weise aufzubauen, da nur wenige Bauteile erforderlich sind.

Näheres bezüglich der Bauteile entnehmen Sie der Stückliste. Man kann mit dem Light-Pen eine Position am Bildschirm feststellen und mit Hilfe eines kleinen Programmes in den Computer bringen.

Da das Bild beim Fernsehgerät durch einen sich schnell bewegenden Lichtpunkt aufgebaut wird, kann man mit einem Fototransistor und einer Schaltung mit geringem Aufwand diesen Lichtpunkt erfassen und das so entstandene Signal sich zu nutze machen.

Das Signal wird in den Computer geschickt und dieser berechnet die Positionen für X und Y und speichert sie ab.

Die Abfrage dieser gespeicherten Po-

sitionen erfolgt mit dem Befehl PEEK. Der Light-Pen funktioniert nicht bei den Farben schwarz und rot und auch nicht am Bildschirmrand.

Weiter hat der Light-Pen einen Sensor, den man auf verschiedene Art nutzen kann.

Für den Bau des Light-Pens ist ein kleines Testprogramm beigelegt, welches helfen soll die Funktionen und die Arbeitsweise des Light-Pens kennenzulernen.

Stückliste:

4 Stk. Widerstände 47K
 2 Stk. Widerstände 10K
 1 Stk. Widerstand 100K
 1 Stk. Widerstand 220K
 1 Stk. Folienkond. 33nF
 1 Stk. Folienkond. 2N2
 1 Stk. Folienkond. u47

1 Stk. Tandalkond. ul
 3 Stk. Transistoren BC 337
 2 Stk. Transistoren BC 549
 1 Stk. Fototransistor BPY 61 BPW 22
 1 Stk. IC-74123
 1 Stk. IC-Sockel 16 pin
 1 Stk. Lochrasterplatine 5cm x 8cm
 1 Stk. Anschlußstecker für Joystick-board
 1 Kabel 2-polig/abgeschirmt
 1 Stk. Alter Kugelschreiber
 Der Aufbau ist unkompliziert, man muß nur darauf achten, daß der Fototransistor mindestens 3mm tief versenkt wird.

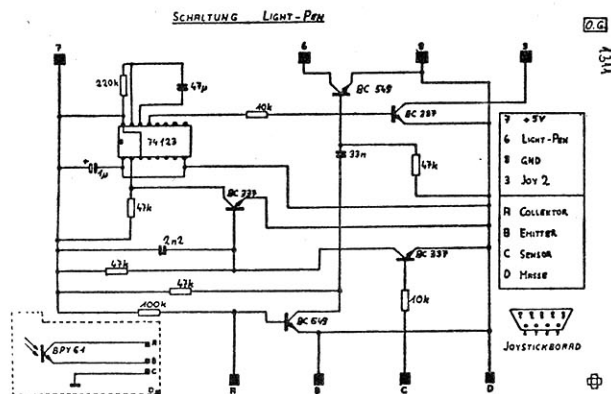
Die Schaltung selbst muß nicht in ein Gehäuse eingebaut werden. Man kann diese Schaltung auch noch erweitern, allerdings nach eigenem Ermessen!

schaltplan.


```

1 REM*****
5 REM* LIGHT-PEN *
6 REM*****
10 PRINT" "
20 KX=53:KY=36
30 R1=PEEK(648)*256:R2=37888+(PEEK(648)AND2)*256
40 RX=36870:RY=RX+1:RS=37137
50 PRINT"TESTPROGRAMM FUER "
60 PRINT" LIGHT-PEN"
70 PRINT" STECKEN SIE DEN LIGHT-PEN IN DEN JOYSTICK- BOARD."
80 PRINT" HABEN SIE DEN LIGHT- PEN ANGESTECKT ?. DANN:"
90 PRINT" SENSOR BERUEHREN "
100 GOSUB580
110 PRINT" DIE POKE ADRESSEN FUER DIE KOORDINATEN SIND FUER <X>=36870,"
120 PRINT" FUER <Y>=36871,"
130 PRINT" DER SENSOR KANN MIT DER ADRESSE 37137/BIT4 ABGEFRAGT WERDEN."
140 PRINT" S=PEEK(37137)AND16;"
150 PRINT" DAS IST:"
160 PRINT" 16 = BERUEHRT. 0 = NICHT BERUEHRT"
170 GOSUB560
180 S=PEEK(RS)AND16:IFS=0THEN180
190 PRINT" NACHFOLGENDES PROGR. MACHT DIE WERTE SICHTBAR."
200 PRINT" 10PRINT"CHR$(34)"CHR$(34)"PEEK(36870),PEEK(36871)"
210 PRINT" Fahren Sie mit dem PEN ueber den Bild- SCHIRM, JETZT SEHEN SIE D
IE:"
220 PRINT" KOORDINATEN."
230 S=PEEK(RS)AND16:IFS=16THENPRINT" PEEK(RX),PEEK(RY):GOTO230
240 S=PEEK(RS)AND16:IFS=0THEN240
250 PRINT" BERECHNUNG DES ZEICHENPLATZES AUF DEM BILD- SCHIRM"
260 PRINT" FOLGENDES PROGRAMM:"
270 PRINT" 10 KX=49:KY=36:RX=36870:RY=RX+1:RS=37137"
280 PRINT" 20 X=PEEK(RX):Y=PEEK(RY)"
290 PRINT" 30 X=INT((X-KX)/4):Y=INT((Y-KY)/4)"
300 PRINT" 40 IFX<0THENX=0"
310 PRINT" 50 IFX>21THENX=21"

```



```

320 PRINT"60 IFY<0THENY=0"
330 PRINT"70 IFY>22THENY=22"
340 GOSUB560
350 PRINT" DEMONSTRATION"
360 PRINT" Fahren Sie mit dem LIGHT-PEN ueber den BILDSCHIRM"
370 PRINT" WENN SIE DEN SENSOR BERUEHREN WIRD EIN PUNKT GESETZT."
380 PRINT" MIT SPACE WIRD DER SCHIRM GELOESCHT MIT RETURN STARTET DER
ST NEU"
390 GOSUB460
400 GETX$:IFX$="" THENPRINT" ":GOTO390
410 IFX$=CHR$(13)THEN:RUN
420 S=PEEK(RS)AND16:IFS=16THEN390
430 POKER1+A,46:POKER2+A,6
440 GOTO390
450 STOP
460 X1=PEEK(RX):X2=PEEK(RX):X3=PEEK(RX):X4=PEEK(RX):X5=PEEK(RX)
470 X=(X1+X2+X3+X4+X5)/5
480 Y=PEEK(RY):X=INT((X-KX)/4):Y=INT((Y-KY)/4)
490 IFX<0THENX=0
500 IFX>21THENX=21
510 IFY<0THENY=0
520 IFY>22THENY=22
530 A=Y*22+X:SG=PEEK(R1+A):SC=PEEK(R2+A)
540 POKER1+A,SG:POKER2+A,6:FORJ=1TO200:NEXT:POKER1+A,SG:POKER2+A,SC
550 RETURN
560 PRINT" SENSOR BERUEHREN "
570 S=PEEK(RS)AND16:IFS=0THEN570
580 S=PEEK(RS)AND16:IFS=16THEN580
590 RETURN
READY.

```

GOLD der GEISTER

für den VC-20 o.E.

Nach dem "RUN" erscheint auf dem Bildschirm ein Labyrinth. Dahinein werden 15 Goldmünzen verteilt. Auf dem Rand drumherum werden 8 Geister gesetzt. Sie können sich auch nur auf diesem Streifen bewegen. Wenn die Vorbereitungen beendet sind, ertönt ein kurzer Ton und die Zeit beginnt zu laufen. In der Mitte befindet sich die Figur. Sie wird folgendermaßen bewegt:

I: hoch; M: runter; J: links; K: rechts

Wenn man mit der Figur auf eine Goldmünze läuft, erklingt ein Signal, und die bis dahin verstrichene Zeit (in Sekunden steht rechts oben in der Ecke) wird zur Punktezahl addiert. Aber alle Punkte nutzen nur, wenn man in einer Minute aus dem Labyrinth herauskommt. Das geht folgendermaßen: Man läuft in eine Ecke des weißen Streifens (sofern die Geister das zulassen – man darf sich nicht von ihnen berüh-

ren lassen). Dann ertönt eine Melodie und man muß noch in eine zweite Ecke laufen. (Es muß eine andere sein, als die erste).

Die Geister sind nicht ungefährlich: Sie kommen auf den Spieler zu.

Natürlich nur wenn Sie können, aber wenn man von zwei Geistern eingeschlossen ist, ist das Spiel meistens vorbei. Zum Glück können die Geister nicht sehr gut sehen. Sie bemerken den Spieler erst, wenn er drei Felder oder näher an ihnen steht.

Noch ein Tip: Es ist empfehlenswert, gleich zu Beginn des Spieles in die erste Ecke zu laufen, dann die Münzen zu sammeln und danach in die zweite Ecke zu laufen, denn je älter die Münzen sind, desto wertvoller sind sie.

Wird während der Minute die zweite Ecke nicht angelaufen, so verfallen sämtliche Punkte.

Variablenliste:

a: Positionen der Geister 0-7

b: Nummer des Geistes, der dran ist

c: zählt, wieviele Geister schon gesetzt sind (Zeile 480 bis 500)

d: alter Inhalt des bestiegenen Feldes

e: gibt an, in welche Richtung der aktuelle Geist bewegt werden soll

f: Ziel des Geistes

g: Peek(f)

h: Laufvariable für Zeitschleifen

i: Punkte

j: Rekord

k: gedrückte Taste [Peek(197)]

l: Position der Spielfigur

m: gewünschter Schritt

n: Inhalt des Feldes, das von der Spielfigur bestiegen werden soll

o: alter Inhalt des von der Spielfigur bestiegenen Feldes

p: universelle Laufvariable

q: Anzahl angelaufener Ecken

r: zuletzt angelaufene Ecke

Diese Belegung der Variablen (außer c) gilt erst ab Zeile 510!
a und d sind Felder mit je 8 Stellen.

```

0 GOTO 160
10 #####
20 #GOLD DER GEISTER#
30 #
80 #
90 #####
160 POKE56,29:POKE55,0:CLR:DIMA(7),D(7):FORB=7424T07431:POKEB,0:NEXT
170 FORB=7488T07527:READC:POKEB,C:NEXT
180 DATA170,85,170,85,170,85,170,85,28,34,73,85,73,34,28,0,156,62,107
190 DATA127,127,127,127,107,0,0,0,24,24,0,0,0,28,34,65,65,65,34,28,0
200 POKE36879,0:POKE36869,255
210 PRINT"#####"
220 PRINT"#####"
230 PRINT"#####"
240 PRINT"#####"
250 PRINT"#####"
260 PRINT"#####"
270 PRINT"#####"
280 PRINT"#####"
290 PRINT"#####"
300 PRINT"#####"
310 PRINT"#####"
320 PRINT"#####"
330 PRINT"#####"
340 PRINT"#####"
350 PRINT"#####"
360 PRINT"#####"
370 PRINT"#####"
380 PRINT"#####"
390 PRINT"#####"
400 PRINT"#####"
410 PRINT"#####"
420 B=0:C=0
440 A=INT(RND(1)*506)+7680
450 IFPEEK(A)=160ANDPEEK(A+30720)=6THENPOKEA,43:POKEA+30720,7:B=B+1
460 IFB<15THEN440
480 A=INT(RND(1)*506)+7680
490 B=PEEK(A):IFB=400RB=230THEND(C)=B:POKEA,42:A(C)=A:C=C+1
500 IFC<8THEN480
510 GOSUB1120
520 FORB=0T07
530 E=INT(RND(1)*4)+1:GOSUB1380
540 ONEGOSUB1000,1030,1060,1090
550 K=PEEK(197)
560 IFK=12THENM=-22
570 IFK=36THENM=22
580 IFK=20THENM=-1
590 IFK=44THENM=1
600 IFK=64THENM=0:GOTO900
610 N=PEEK(L+M)
620 IFN<40ANDN>42ANDN<230ANDN<43ANDN<160THEN900
630 POKEL,0:0=N:L=L+M:POKEL,41:POKEL+30720,1:POKE36877,150:POKE36877,0
640 IF0=43THEN1160
650 IF0=40ANDL<0THEN1190
900 IF0=42ORPEEK(L+22)=42ORPEEK(L-22)=42ORPEEK(L-1)=42ORPEEK(L+1)=42THEN1180
960 IFTI#>"000059"THEN1180
970 PRINT"#####I"
980 PRINT,"#####MID$(TI#,5,2)
990 NEXT:GOTO520
1000 F=A(B)+22:G=PEEK(F):IFG=400RG=230THEN1020
1010 RETURN
1020 POKEA(B),D(B):D(B)=G:POKEF,42:A(B)=F:RETURN
1030 F=A(B)-22:G=PEEK(F):IFG=400RG=230THEN1020
1040 RETURN
1050 POKEA(B),D(B):D(B)=G:POKEF,42:A(B)=F:RETURN
1060 F=A(B)+1:G=PEEK(F):IFG=400RG=230THEN1020
1070 RETURN
1080 POKEA(B),D(B):D(B)=G:POKEF,42:A(B)=F:RETURN
1090 F=A(B)-1:G=PEEK(F):IFG=400RG=230THEN1020
1100 RETURN
1110 POKEA(B),D(B):D(B)=G:POKEF,42:A(B)=F:RETURN
1120 POKE36878,15:POKE36876,190:POKE36875,162:FORH=1T0700:NEXT:POKE36876,0:POKE3
6875,0
1130 TI#="000000":I=0:L=7954:PRINT"#####PUNKTE:", "#####ZEIT"::Q=0:R=7680:O=160
1150 RETURN
1160 FORP=1T03:POKE36876,235:FORH=1T050:NEXT:POKE36876,0:FORH=1T050:NEXT:NEXT
1170 I=I+VAL(MID$(TI#,5,2)):O=160:GOTO900
1180 POKEL,44:FORP=255T0127STEP-.3:POKE36874,P:NEXT:GOTO1220
1190 POKE36875,199:FORH=1T0300:NEXT:POKE36875,0:FORP=1T03:POKE36876,224+P:FORH=1
T0200
1200 NEXT:POKE36876,0:NEXT:Q=Q+1:IFQ=2THEN1250
1210 R=L:GOTO900
1220 POKE36879,170
1230 PRINT"##### SCHADE, DAS WAR NIX ! #####
***"

```



```

1240 I=0:GOTO1270
1250 POKE36879,109
1260 PRINT"*****GUT GEMACHT!*****"
***"
1270 POKE36869,240:IFI>JTHEN1350
1280 PRINT"NEUER REKORD"J
1290 PRINT"PFUNKTE"J
1300 PRINT"NOCHMAL ?":POKE198,0
1310 GETA$
1320 IFA$="J"THEN200
1330 IFA$="N"THENEND
1340 GOTO1310
1350 J=I:PRINT"NEUER REKORD !!!"
1360 FORP=255TO128STEP-1:POKE36875,P:NEXT:FORP=128TO255:POKE36875,P:NEXT:POKE368
75,0
1370 GOTO1290
1380 IFA(B)-2=LORA(B)-3=LTHENE=4
1390 IFA(B)+2=LORA(B)+3=LTHENE=3
1400 IFA(B)-44=LORA(B)-66=LTHENE=2
1410 IFA(B)+44=LORA(B)+66=LTHENE=1
1420 RETURN

```

Simons Neuplot

für den C-64 + *Simons Basic*

Nach dem Laden und Anstarten mit RUN, erscheint die Frage, ob Sie "Simon's Basic" geladen haben. Da das Programm viele "Simon's Basic"-Befehle benutzt, müssen Sie vorher "Simon's Basic" geladen haben, um mit dem Programm arbeiten zu können.

Wenn Sie die Frage mit 'J' beantworten, erscheint das Hauptmenue:

Zum Auswählen stehen 5 Programmunterpunkte zur Verfügung.

Beim Wählen von Punkt 1 erscheinen die momentanen Funktionen, und beim Drücken einer Taste (bitte nicht 'Return') erscheint ein weiteres Menue:

Beim Wählen von 1 werden Sie nach dem Darstellungsbereich der Funktionen gefragt; durch das Wählen kleiner Bereiche (0x1) auf X- und Y-Achse, werden die Funktionsverläufe in der Nähe des Ursprungs vergrößert, vernünftige Werte um sich einen Überblick über die Funktionen zu verschaffen, liegen je nach Funktion zwischen -30/30 und -5/5 auf X- und Y-Achse.

Der zweite Menüpunkt sollte nur gewählt werden, wenn eine Funktion be-

reits geplottet wurde, und man sich (aus welchen Gründen auch immer) die alte Grafik noch einmal ansehen will.

Nach Zeichnen bzw. Zeigen der Funktionen wartet das Programm auf einen Tastendruck und kehrt dann zum Hauptmenue zurück.

Beim Wählen des dritten Menüpunktes erfolgt ebenfalls ein Rücksprung ins Hauptmenue, z.B. wenn man irrtümlich die graphische Darstellung gewählt hat.

Das Wählen des zweiten Menüpunktes erlaubt die Ausgabe einer Wertetabelle. Dazu erscheinen die momentanen Funktionen und Sie wählen jeweils Funktion, den niedrigsten bzw. höchsten X-Wert und die Schrittzahl. Nach einer Rückfrage, ob alle Werte korrekt eingegeben wurden, werden dann die gewünschten X- und Funktionswerte ausgegeben. Nach einem Tastendruck kehren Sie wieder zum Hauptmenue zurück.

Die Hauptmenuepunkte 3 und 5 sind eigentlich im Programm gut erklärt und relativ absturzsicher, dürften also keine

großen Probleme aufwerfen.

Beim Wählen des vierten Punktes bietet sich die Gelegenheit, die vorhandenen Funktionen abzuändern, und zwar nach Drücken von F1 mit Hilfe der Cursorsteuerung. Dabei sollte man auch die Stringvariable FKT\$ nicht vergessen abzuändern, weil es sonst zu Mißverständnissen kommt, wenn das Programm die Funktionen zeigt. Nicht benötigte Funktionen werden als '0' definiert.

Nach erfolgreichem Abändern, drücken Sie bitte F7 und starten damit das Programm von vorn.

Zu Bemerken sei noch, daß jeder Unterprogrammpunkt durch eine verschiedene Farbe gekennzeichnet ist:

Graphische Darstellung: gelb

Ausgabe Wertetabelle: türkis

Hardcopy: orange

Änderung: hellblau

Programmende: violett

Anmerkung: In Zeile 3001 hat der Drucker nach dem reversen Herz ein reverses Pik (C= und 1) vergessen!

programme

```

0 REM *** FUNKTIONENPLOT ***
5 REM *****
6 REM DIESES PROGRAMM IST
7 REM NUR MIT SIMONS BASIC
8 REM LAUFFAEBIG !!!
9 REM *****
10 DEF FNA(X)=(1/12)*X^4-2*X^2      :FKT$(1)="(1/12)*X^4-2*X^2"
15 :
20 DEF FNB(X)=(1/3)*X^3-4*X        :FKT$(2)="(1/3)*X^3-4*X"
25 :
30 DEF FNC(X)=X^2-4                :FKT$(3)="X^2-4"
35 :
40 DEF FND(X)=0                    :FKT$(4)="0"
41 :
42 COLOUR 0,0
43 PRINT "*****" ACHTUNG ! ! !
44 PRINT "***** HABEN SIE 'SIMONS BASIC' GELADEN";
45 INPUT A$
46 IF A$="N" THEN PRINT "***** BITTE LADEN SIE ERST 'SIMONS BASIC' !" :END
47 IF A$(">")"J" THEN PRINT "*****":GOTO 44
48 :
50 PRINT "*****"
60 PRINT "***** FUNKTIONENPLOT*****"
70 PRINT "DRUECKE:***** 1 FUER GRAPH. DARSTELLUNG"
80 PRINT "***** 2 FUER AUSGABE WERTETABELLE"
90 PRINT "***** 3 FUER HARDCOPY AUF DRUCKER"
100 PRINT "***** 4 FUER ABAENDERN DER FUNKTIONEN"
110 PRINT "***** 5 FUER PROGRAMMENDE"
120 W$="":GETW$
130 IF W$="" THEN 120
140 IF W$="1" THEN CALL FUNKTIONENPLOT
150 IF W$="2" THEN CALL WERTETABELLE
160 IF W$="3" THEN CALL HARDCOPY
170 IF W$="4" THEN CALL AENDERUNG
180 IF W$="5" THEN CALL PROGRAMMENDE
190 GOTO 120
200 :
1000 PROC FUNKTIONENPLOT
1001 : PRINT "*****"TAB(13);"FUNKTIONSPLOT"
1010 : PRINT "***** FUNKTIONEN"
1020 : FOR I=1 TO 4
1021 : PRINT "*****" F(X)=";FKT$(I)
1022 : NEXT I
1023 : PRINT "*****" [TASTE]
1024 : POKE 198,0:WAIT 198,1
1025 : PRINT "***** DRUECKE"
1026 : PRINT "***** 1 FUER NEU PLOTTEN"
1027 : PRINT "***** 2 FUER ALTEN GRAPH NEU ZEIGEN"
1028 : PRINT "***** 3 FUER RUECKKEHR ZUM MENUE"
1029 : GET C$:IF C$="" THEN 1029
1030 : IF C$="1" THEN 1039
1031 : IF C$="2" THEN 1034
1032 : IF C$="3" THEN 50
1033 : GOTO 1029
1034 : CSET2
1035 : POKE 198,0:WAIT 198,1
1036 : NRM
1037 : GOTO 50
1039 : PRINT "***** DARSTELLUNGSBEREICH DER FUNKTIONEN*****"
1040 : INPUT " X MIN, X MAX: ";X1,X2
1050 : IF X1=0 OR X2=0 THEN 1000
1060 : INPUT " Y MIN, Y MAX: ";Y1,Y2
1070 : IF Y1=0 OR Y2=0 THEN 1000
1080 : HIRES5,0
1090 : YM=(Y2-Y1)/200
1100 : IF Y1>0 OR Y2<0 THEN 1141
1110 : Y=ABS(Y1)/YM
1120 : FOR X=0 TO 319:REM X-ACHSE
1130 : PLOT X,Y,1
1140 : NEXT X
1141 : ANZ=X2-X1
1142 : SRK=320/ANZ
1143 : FOR I=1 TO ANZ+1
1144 : PLOT XP,Y+1,1
1145 : PLOT XP,Y+2,1
1146 : XP=XP+SRK
1147 : NEXT I
1150 : CHAR311,Y+5,24,1,1
1160 : XM=(X2-X1)/320
1170 : IF X1>0 OR X2<0 THEN 1211
1180 : X=ABS(X1)/XM
1190 : FOR Y=0 TO 199:REM Y-ACHSE
1200 : PLOT X,Y,1
1210 : NEXT Y
1211 : ANZ=Y2-Y1
1212 : SRK=200/ANZ
1213 : FOR I=1 TO ANZ+1
1214 : PLOT X+1,YP,1
1215 : PLOT X+2,YP,1
1216 : YP=YP+SRK
1217 : NEXT I
1220 : CHARX+5,0,25,1,1
1230 : A$=" URSPRUNG -> RAND =" +STR$(X2)+" EINHEITEN"
1240 : TEXT 0,191,A$,1,1,10
1250 : FOR XL=X1 TO X2 STEPXM*3
1260 : Y(1)=FNA(XL)
1270 : Y(2)=FNB(XL)
1280 : Y(3)=FNC(XL)
1290 : Y(4)=FND(XL)
1300 : FOR K=1 TO 4
1310 : Y=(-Y(K)-Y1)/YM
1320 : IF Y<0 OR Y>199 THEN 1340
1330 : X=(XL-X1)/XM
1335 : PLOT X,Y,1
1340 : NEXT K
1350 : NEXT XL
1360 : POKE 198,0:WAIT 198,1
1370 : NRM
1380 GOTO 50
1390 :
2000 PROC WERTETABELLE

```

```

2001 : PRINT"          WERTETABELLE":PRINT" "
2002 : FOR I=1 TO 4
2003 :   PRINT"      F(X)=";FKT$(I)
2004 : NEXT I
2005 : PRINT" "
2010 : PRINT"          WELCHE FUNKTION [1-4]";:FETCH"1234",1,X$:FKT=VAL(X$)
2011 : PRINT"          "
2015 : PRINT"          WELCHER BEREICH:"
2020 : INPUT"  X-MIN.: ";MIN:PRINT" ";TAB(19);:INPUT"  X-MAX.: ";MAX
2030 : INPUT"  SCHRITZAHL: ";SRT
2040 : PRINT"          IHRE ANGABEN SIND:";PRINT"      FUNKTION: ";FKT
2050 : PRINT"  X-MIN. : ";MIN:PRINT"  X-MAX. : ";MAX:PRINT"  SCHRITT.: ";SRT
2060 : PRINT"          IST ALLES KORREKT [J/N] ?":K$=""
2070 : GETK$:IF K$="" THEN 2070
2080 : IF K$="N" THEN 2000
2090 : IF K$<>"J" THEN 2000
2100 : PRINT"          X-WERT      | FUNKTIONSWERT"
2110 : PRINT"          "
2120 : FOR I=MIN TO MAX STEPSRT
2130 :   IF FKT=1 THEN FW=FNA(I)
2140 :   IF FKT=2 THEN FW=FNB(I)
2150 :   IF FKT=3 THEN FW=FNC(I)
2160 :   IF FKT=4 THEN FW=FND(I)
2170 :   IF I=0 THEN FW=0
2180 :   A$="###.#####"
2190 :   PRINT:PRINT"      ";USEA$,STR$(I);PRINTTAB(20);:USEA$,STR$(FW)
2200 : NEXT I
2210 : PRINT:PRINT"          TASTE FUER ENDE"
2220 : T$="":GETT$
2230 : IF T$="" THEN 2220
2240 : GOTO 50
2250 :
3000 : PROC HARDCOPY
3001 : PRINT"          HARDCOPY"
3010 : PRINT"          NACH DRUECKEN EINER BELIEBIGEN TASTE ER"
3020 : PRINT"          FOLGT EINE HARDCOPY DER GRAFIK AUF DEN FOLGENDEN PRINTERN:"
3030 : PRINT"          - VC 1520                      - VC 1525"
3040 : PRINT"          - SEIKOSHA GP 100 VC                - EPSON RX-80/FX-80"
3050 : PRINT"          (MIT INTERFACE FUER C-64)"
3060 : PRINT"          ABER NICHT VC 1526 !!"
3070 : PRINT"          ";TAB(10);:ITASTE DRUECKEN!"
3080 : GETH$:IF H$="" THEN 3080
3090 : PRINT"          BITTE WARTEN SIE AUF DEN DRUCKER"
3100 : COPY
3110 : GOTO 50
3120 :
4000 : PROC AENDERUNG
4001 : PRINT"          ABAENDERN DER FUNKTIONEN"
4010 : PRINT"          NACH DEM DRUECKEN DER TASTE F1 WERDEN"
4020 : PRINT"          DIE FUNKTIONEN AUFGEListET."
4030 : PRINT"          BITTE AENDERN SIE SIE NACH BELIEBEN      MIT HILFE DER CURSORSTEUERUNG"
4040 : PRINT"          NACH ABGESCHLOSSENER AENDERUNG DRUECKEN SIE BITTE F7!"
4050 : POKE198,0:WAIT198,1
4060 : KEY1,"LIST10-40"+CHR$(13)
4070 : KEY7,"RUN"+CHR$(13)
4080 : PRINT"          >BITTE F1 DRUECKEN !      >WEITER MIT F7 !"
4090 : END
4100 :
5000 : PROC PROGRAMMENDE
5001 : PRINT"          PROGRAMMENDE"
5010 : PRINT"          "
5020 : PRINT"          DRUECKE"
5030 : PRINT"          1 FUER ENDE MIT NEW"
5040 : PRINT"          2 FUER ENDE OHNE NEW"
5050 : GETA$
5060 : IF A$="" THEN 5050
5070 : ELSE:GOTO 5080
5080 : A=VAL(A$)
5090 : IF A=1 THEN 5120
5100 : IF A=2 THEN 5200
5110 : GOTO 5050
5120 : PRINT"          WOLLEN SIE WIRKLICH"
5130 : PRINT"          DAS PROGRAMM LOESCHEN ?";
5140 : IF A$="N" THEN 5000
5150 : IF A$="J" THEN PRINT"          ":NEW
5160 : FETCH"JN",1,A$
5170 : IF A$="N" THEN 5000
5180 : IF A$="J" THEN PRINT"          ":NEW
5190 : END
5200 : PRINT"          BIS BALD ! ? !"
5210 : END
READY.

```


Teil 6

Logik im Basicprogramm

Wie Sie sich wohl erinnern werden, sprachen wir vor einer Woche über die einzelnen Schritte der Programmierung. Heute wollen wir diese Ausführung in Hinsicht auf den Basic-Interpreter unseres Computers erweitern.

Während wir mit unserem Rechner arbeiten, entwickelt sich bei uns von ganz allein eine bestimmte Vorstellung in Bezug auf die Abläufe im Innern des Computers. Ob diese Vorstellungen nun der technischen Realität entsprechen oder nicht, spielt keine Rolle. Wichtig ist nur, daß Sie in der Lage sind, die Funktion des Programmablaufes zu erklären.

Wir kommen in diesem Zusammenhang zum Modell-Begriff: Der große österreichische Physiker Ernst Mach prägte ihn für ein Ding, ein Schema oder eine Idee, die geeignet sind reale Vorgänge, deren direkte Erforschung uns mit gegenwärtigen Mitteln nicht möglich ist, geistig zu erklären. Bekannt sind zum Beispiel die verschiedenen Atommodelle, die den Menschen seit den Tagen des Demokrit von Abdera helfen, Vorgänge in der Welt des Kleinsten, das sich direkter Anschauung entzieht, plausibel deuten zu können.

Wichtig am Modell ist somit nicht unbedingt die Übereinstimmung mit dem Tatsächlichen, sondern mit den Symptomen der Realität.

Programmieren ist nun das für einen maschinellen Prozessor geeignete Aufbereiten und Durchformulieren eines Programm-Schemas. Obgleich die Problemanalyse, die Aufbereitung für den Prozessor und die syntaktische Umsetzung in eine dem Interpreter verständliche Sprache verschiedene Dinge sind, bilden sie doch eine semantische (d.h. sinnbezogene) Einheit - Wechselwirkungen sind selbstverständlich. Insbesondere wirken die Möglichkeiten der von uns benutzten Programmiersprache Basic, ihre Nähe bzw. Ferne zu bestimmten Problemstrukturen sowie die Vorstellungen unsererseits in Bezug auf den vom Rechner benutzten Prozessor positiv auf den Prozess der Programmbe-wältigung zurück.

Wichtig ist dabei auch unsere Bereitschaft, nicht jede Möglichkeit vonseiten des Interpreters zu nutzen, um uns die Sache so einfach wie möglich zu machen.

Versierte Programmierer und Informatiker sind nicht einig darüber, daß beispielsweise Sprungbefehle nur sehr zurückhaltend benutzt werden sollen, wenn man gut konzipierte Programme schreiben will. Wesentlich günstiger ist es, Entscheidungen an Ort und Stelle zu treffen, statt in Unterprogramme zu verzweigen, um lediglich eine oder zwei Bedingungen zu prüfen.

Hilfreich ist es in diesem Zusammenhang, die in Basic obligatorischen Zeilennummern zunächst wegzulassen und ein Programm ohne Zeilennummern und Sprungbefehle zu schreiben. Dies wird sicher manchem, der schon ein wenig Programmiererfahrung in Basic besitzt, lästig, ja vielleicht sogar unmöglich erscheinen. Die Erziehung der eigenen Person zu strukturiertem Denken (wie es erfolgreiche Programmierer auszeichnet) wird aber durch diese Verfahrensweise entscheidend gefördert.

Doch nicht nur der (vorläufige) Verzicht auf Programmzeilen, sondern ebenso die ständige Übersetzung von Basic-Befehlen in's Deutsche helfen Ihnen, sich der Möglichkeiten wie Handicaps einer höheren Programmiersprache bewußt zu werden.

Diese Übersetzungsarbeit ist schon deshalb sinnvoll, weil bei Formulierung der englischsprachigen Anweisung aus Rücksicht auf den Speicherplatzbedarf sowie den Komfort im Programm-Eintippen (lange Anweisungen sind umständlicher einzugeben) so knapp als irgend denkbar gehalten wurde. Diese Sparsamkeit bringt es mit sich, daß nicht bei jeder einzelnen Anweisung, bei jedem einzelnen Basic-Befehl der Sinn im Wort erkenntlich wird. Das Sichbewußt-

machen einer bestimmten Funktion und eine entsprechende Übersetzung hilft dann ungemein, eine Optimierung des eigenen Sprachschatzes zu erreichen.

Wir fassen dies bis hierher zusammen:

1. Wir beobachten den Rechner, seine Arbeitsweise, seine Reaktion auf bestimmte Eingaben unsererseits und entwickeln uns ein Modell von den internen Vorgängen im Gehäuse des Commodore.

2. Als Modell bezeichnen wir die Summe von Vorstellungen, die geeignet sind, bestimmte Geschehnisse, die wir sinnlich nicht direkt wahrnehmen können, wirklichkeitsnah zu beschreiben. Nicht die tatsächliche Übereinstimmung des Modelles mit der physischen Realität ist entscheidend, sondern die Eignung, bestimmte Abläufe treffend deuten zu helfen.

3. Komfort des Interpreters bedeutet nicht automatisch auch besseres Ergebnis in Sachen Programmierung. Im Gegenteil: Die Erfahrungen von Fachleuten zeigen, daß vielfach geraten ist, auf ein Ausschöpfen der programmtechnischen Möglichkeiten zu verzichten und stattdessen erhöhten Wert auf eine strukturierte Vorbereitung des Programmaufbaues zu legen.

a) Zeilennummern sollen zunächst eingespart und erst nach Fertigstellung des Programmes in einer eigenen Anpassungsphase nachgetragen werden.

b) Die im Computer implementierten Befehle und Anweisungen sollten auf ihren Funktionsgehalt hin analysiert in's Deutsche übersetzt werden.

c) Das Programm sollte zunächst unter Benutzung der deutschsprachigen Anweisung verfaßt werden. Die "Reinschrift" in Form eines Übersetzungsverfahrens gehört dann wieder in die Anpassungsphase.

Um insbesondere das unter Punkt drei Gesagte noch einmal zu verdeutlichen, will ich die einzelnen Schritte anhand eines kleinen Beispiels erklären.

Gehen Sie bitte davon aus ein Programm schreiben zu müssen, das die Fakultät einer gegebenen Zahl ermittelt. Denjenigen, die nicht tagtäglich mit derartigen Problemen konfrontiert werden, sei gesagt, daß man unter Fakultät in diesem Fall nicht den Teil einer Hochschule, sondern die Menge an Möglichkeiten begreift, eine bestimmte Zahl von Dingen zu ordnen. Der Mathematiker schreibt zum Beispiel $4! = 24$ und will damit zum Ausdruck bringen, daß 4 Dinge auf 24 verschiedene Weisen geordnet werden können. Die Fakultät einer Zahl (dargestellt durch ein Ausrufezeichen) errechnet sich, indem man sämtliche natürliche Zahlen bis zu 1 hinunter multipliziert und dabei mit derjenigen Zahl anfängt, deren 'Fakultät' errechnet werden soll. In unserem Beispiel müssen wir somit $4 \times 3 \times 2 \times 1$ rechnen, was 24 ergibt.

Doch nun zu unserem Programm:
Wir schreiben uns zunächst einmal auf, was geleistet werden soll:

1. Der Ausgangswert soll angefordert werden.
2. Die Fakultät des eingegebenen Wertes soll ermittelt werden.
3. Die Fakultät des Ausgangswertes soll angezeigt werden.

Diese drei elementaren Funktionen unseres Programmes können natürlich noch um die eine oder andere Ausgabe-Vorschrift oder Sicherungsanweisung (z.B. Begrenzung des Eingabewertes) erweitert werden.

Nun wird die Erstfassung des Programmes erstellt - mit deutschen Anweisungen (die sich freilich an den sprachlichen Möglichkeiten des Rechners orientieren) und ohne Zeilennummern.

Prüfe ob

Fällt Prüfung positiv aus so

$(a-1) < 1$ ist.

Zeige an

$a!$ = (aktueller a-Wert)
Beende das Programm!

Mache erneuten Durchlauf

Nun können wir immer noch in deutscher Fassung und ohne Zeilen-Nummern das

Programm in eine Art Pseudo-Basic übertragen.

Bemerkung

Programm zur Berechnung der Fakultät
A

Gebe ein

Für I \$ A bis 1 Schrittweite -1

Laß $A = A * (A-1)$ sein

Wenn $(a-1) < 1$ ist, dann drucke aus " $a!$ " sowie den augenblicklichen Wert der Variablen A.

* E n d e

(andernfalls)

nächstes I

Sehen Sie, da haben wir unser Pseudobasic-Programm!

Manchem mag die detaillierte Übertragung der englischen Basicanweisungen in unsere Muttersprache lächerlich erscheinen. Und angesichts des vergleichsweise simplen Programmes mag er auch mit einiger Berechtigung glauben, diese Phase überspringen zu können. Wenn wir aber erst einmal so weit gekommen sind, recht komplexe Listings zu erstellen, wird auch der Skeptiker von heute merken, wie hilfreich unser Vorgehen ist. Und gerade auch dem Lernenen prägt sich das vielfältige Instrumentarium der Programmiersprache Basic wesentlich gründlicher ein, wenn er die einzelnen Anweisungen nicht blind übernimmt, sondern sich durch die Übersetzung Sinn und Zweck jedes Befehls vergegenwärtigen kann.

Der Rest ist nun ein Kinderspiel...

10:REM PROGRAMM ZUR BERECHNUNG DER FAKULTÄT

20:INPUT A

30:FOR I=A TO 1 STEP -1

40:LET A=A*(A-1)

50:IF (A-1) < "1" THEN PRINT "a!"

="";A:END

60:NEXT I

Lesen Sie sich nun noch einmal beide Fassungen durch. Sie spüren sicher sofort, um wieviel plastischer, klarer, verständlicher die englische "Originalbasic-Version" geworden ist.

Als Hausaufgabe will ich Ihnen das folgende Problem stellen:

Sie erhalten den Auftrag, eine Digitaluhr für Ihren Computer zu programmieren, die folgende Leistungen aufweisen muß....

1. Stunden, Minuten und Sekunden sollen angezeigt werden.
2. Die Commodore-Uhr soll 24-Stunden-Anzeige besitzen.
3. Bei vollen Zeiten sind Nullwerte durch "00" kenntlich zu machen (also bei 16 Uhr z.B. durch "16:00:00" oder "16h-00m-00s").

Die Ganggenauigkeit unserer Computeruhr soll uns dabei weniger interessieren, obwohl es natürlich schön wäre, wenn sie zu allem Luxus auch noch zuverlässig würde.

Viel Spaß bis zum nächsten Mal!

Anmerkung

Fordere an

Eröffne eine Schleife

Setze a gleich

Programm zur Berechnung der Fakultät

Ausgangswert a

mit a Durchläufen

dem a-Wert des letzten Durchlaufes, multipliziert mit der gegenüber a nächstniedrigeren natürlichen Zahl

4. Internationale COMMODORE -Fachausstellung vom 6. bis 8. September 1984 in Frankfurt am Main

Vom 6. bis 8. September 1984 fand in der Halle 1 des Frankfurter Messegeländes die 4. Internationale COMMODORE-Fachausstellung (CFA) statt.

Gemeinsam mit dem Veranstalter, der COMMODORE BÜROMASCHINEN GMBH, präsentierten rund hundert Aussteller aus sechs Ländern auf ca. 4500 Quadratmetern das gesamte Produkt- und Leistungsangebot sowie die überzeugende Vielfalt von Anwendungsmöglichkeiten der Commodore-Produkte.

Jetzt

- Trend zur ernsthaften kommerziellen Nutzung
- Besucherzahl übertraf alle Erwartungen

Rückblick

Wie schon im Vorjahr, jedoch fundierter und im größerem Umfang wurde interessierten Besuchern an allen drei Tagen die Möglichkeit geboten, kostenlos an einer Reihe sorgsam ausgearbeiteter Seminare und Workshops teilzunehmen.

Besonderen Zuspruch fand dabei "COMMODORE--Papst" Jim Butterfield aus Kanada, der unter dem Motto "What's new COMMODORE" ausführliche Informationen der neuen Computer-Generation plus/4, C 16, C 116 vermittelte. Eine umfassende Multivisionsshow informierte mehrmals täglich über das reichhaltige Commodore-Programm und demonstrierte gleichzeitig, wie unterhaltsam und überzeugend sich solche Medien einsetzen lassen, wenn man damit umzugehen weiß.

Die CFA, immerhin die erste und einzige eigene Messe eines DV-Herstellers - also in relativ kurzer Zeit - nicht nur für den inländischen Markt, sondern insgesamt für die europäische Fachwelt eine Bedeutung erlangt, die sogar die Veranstalter bei allem Selbstvertrauen und ihrem

schon sprichwörtlichen Optimismus seinerzeit nicht erwarten durften. So zögern selbst kritische Experten und eher skeptische Fachjournalisten nicht länger, zuzugestehen, daß die CFA neben den Messen in Hannover und Köln mittlerweile zu den wichtigsten Fachveranstaltungen dieser Art gezählt werden muß.

Nach den von Jahr zu Jahr sprunghaft angestiegenen Besucherzahlen seit 1981 wurden für die diesjährige CFA erstmals mehr als zwanzigtausend Besucher erwartet. Am ersten Tag zählte man 5400 Besucher, am zweiten Tag weitere 6600, das Ergebnis für den dritten Tag beläuft sich auf 12500 Besucher, so daß die Gesamtzahl der Besucher in diesem Jahr mit 25500 zu beziffern ist. Zweimal mußte die Halle aus polizeilichen Gründen wegen Überfüllung vorübergehend geschlossen werden.

Trend:

Versucht man nach der Vielzahl von Eindrücken dieser drei Tage und besonders nach der bis zum Abend wachsenden Turbulenz des dritten Tages eine erste vorsichtige Bi-

lanz zu ziehen, so muß man zu dem Schluß kommen, daß nach den immensen Steigerungsraten der letzten Jahre und besonders des abgelaufenen Geschäftsjahres die Bedeutung des Microcomputermarktes für die Wirtschaft und ihre Zukunft gar nicht wichtig genug eingeschätzt werden kann, und der seit Jahren konstatierte und immer noch wachsende Boom seinen Höhepunkt erreicht hat. Neue Hardware-Serien (die COMMODORE-Geräte 116, 16, plus/4 und die neuen Geräte der 8000er Serie) und immer perfekter maßgefertigte Programme für jedes Anwenderbedürfnis tragen dazu bei, dem Markt Bereiche zu erschließen, von denen man noch vor wenigen Jahren nicht einmal zu träumen gewagt hätte. Schon läßt sich durch zukunftssträchtige Neuentwicklungen und hohe Fertigungszahlen unwahrscheinlich günstige Preiskalkulationen erzielen. Ausgereifte und zuverlässige Softwaresysteme für jeden Betriebsbereich können bereits günstiger als herkömmliches, vergleichsweise schwerfällige Büromaschinen angeboten werden, d.h. zum Preis von gängigem

elektronischen Spielzeug, lassen sich nun bereits ernsthafte, kommerziell nutzbare Programme erstellen und einsetzen, deren wichtigste Vorzüge, insbesondere für Laien und Newcomer gerade in ihrer einfachen und praktikablen Anwendungsweise und komfortablen benutzerfreundlichen Handhabung liegen.

Ein wichtiger Trend, der sich neben dem allgemein immer größeren Interesse und Öffentlichkeitsbewußtsein und den hohen Besucherzahlen bei der diesjährigen CFA deutlich abzeichnete, bestand in der Anwesenheit von Interessenten, denen es eindeutig um qualifizierte berufliche Anwendung von Microcomputern ging. Dies wurde von Besuchern und Ausstellern immer wieder bestätigt und durch Andrang und Nachfrage an Ständen, die solche Programme anboten, weithin sichtbar belegt.

Mitarbeiter der Volkshochschule Frankfurt hatten auf Einladung der COMMODORE GMBH einen markenunabhängigen Informationsstand eingerichtet. Auch hier übertraf die Menge der Anfrage an allen drei Tagen alle Erwartungen. Aus der Sicht

dieses gleichsam überparteilichen Standes konnte am Ende der Ausstellung der Trend zu gezielten professionellen Nutzung bestätigt werden.

Einsatzmöglichkeiten:

Gezeigt wurde eine nahezu unbegrenzte Zahl von Nutzungsmöglichkeiten in Organisation und Technik, die deutlich macht, daß die neue Generation von Microcomputern seit geraumer Zeit aus den Kinderschuhen herausgewachsen ist und es sich dabei längst nicht mehr um Werkzeuge für Fans und Fachleute handelt. Es lassen sich kaum noch Branchen und Betriebsbereiche denken, für die mittlerweile nicht eigenständige maßgeschneiderte Programme entwickelt wurden; die Anwendungsskala etwa der Gesellschaft für Energiesparteknik in Pfungstadt reicht von Industriesteuerungsanlagen für Brennöfen der Ziegelei- und Keramikindustrie, für Großküchen und Backwarenhersteller über die Meßwerterfassung und Fertigungskontrolle im Maschinenbau und der Kfz-Technik bis zum Programm von Systemen, die so passend und preiswert gestaltet sind, daß sich damit noch die Energieverbrauchsminde- rung im eigenen Dreipersonenhaushalt lohnt.

Die Firma MOOS & PARTNER in 7537 Remchingen bietet komplette und bis ins einzelne spezifizierte EDV-Programme nicht nur für Architekten, sondern auch für alle Handwerkszweige des Baugewerbes.

Diese Programme wurden Punkt für Punkt aus dem Alltag von Maler-, Spengler-, Dachdeckerbetrieben usw. erstellt und exakt auf die Anforderungen der zuständigen Landes-Innungsverbände abgestimmt. Der Karlsruher Firma SCHNEIDER & KOCH ist es in den letzten Jahren gelungen, mit ihrer eigens entwickelten Netzwerktechnik für Commodore-Rechner der Serien CBM 4000, 8000 und 700 die Kapazität dieser preiswerten Microcomputer bis in die Bereiche der mittleren Datentechnik (MDT) zu erweitern - und dies zu nur etwa ein Drittel des Preises herkömmlicher Systeme, so daß mittlerweile sogar namhafte Firmen der Großindustrie (Thyssen, Merck, Reynold's Tobacco, Böhler Stahl AG, Wacker Chemie und die Lufthansa) wichtige Bereiche ihrer Meßwerterfassung, Test- und Laborchemie, Materialwirtschaft und Finanzbuchhaltung über Schneider- und Koch-Systeme auf der Basis von Commodore-Rechnern abwickeln. Zu diesen anspruchsvollen SK-LAN-Systemen

bietet die Firma Schneider und Koch auf Anfrage in einer Referenzliste eigens konzipierte Software-Programme für praktisch jede Branche und Größenordnung.

Erwähnt werden muß noch, daß einige der renommiertesten pädagogischen Fachverlage in Deutschland (u.a. Langenscheidt; Otto Maier und Westermann) ihre neuen Lernprogramme vorstellten. Programme die nicht nur technisch perfekt sind, sondern auch vom pädagogisch-psychologischen Standpunkt aus mittlerweile so fundiert, daß sich damit umfangreiche Aus- und Fortbildungskurse geradezu spielerisch und als Entspannungsübung bewältigen lassen. Doch sollte man hier auch eine von der Methode wie vom Management her so junge Firma, wie die SM Software AG nicht vergessen, die mit ihren modular aufgebauten Soft-Learning-Kursen für den C-64 eine überzeugende neue Lernmethode zu entwickeln vermochte. Diese Lernmethode wurde nach neuesten tiefenpsychologischen Erkenntnissen auf audiovisueller Grundlage konzipiert und vermag weite Bereiche des Vor- und Unterbewußten für die interaktive Wissensvermittlung zu aktivieren. Ebenfalls für den C-64

wurde von der SM Software AG zusätzlich zu den bewährten Programmen dieser Firma mit der neuen Small Business-Serie die preiswerte und überzeugende Einsteiger-Software für Freiberufler und mittlere Betriebe entwickelt, die den entscheidenden Schritt zur Anwendung im eigenen Bereich bislang noch immer aufzuschieben vermochten.

Fazit:

Abschließend läßt sich sagen, daß mit der CFA 1984 endgültig der Beweis erbracht wurde, daß Microcomputer in unserem Alltag mittlerweile so nützliche und unerläßliche, jedoch erheblich vielseitigere Arbeitsgeräte sind, wie Telefon oder Schreibmaschine, Preis und Bedienungskomfort, sowie die Vielzahl erprobter fertiger Programme für jeden Zweck, gestatten ihren ebenso unprätentiösen wie selbstverständlichen Einsatz.

Der Erfolg der 4. Internationalen COMMODORE-Fachausstellung in Frankfurt am Main und besonders der Besucheransturm des dritten Tages lassen es angeraten erscheinen, die Ausstellung künftig in noch größerem Rahmen und evtl. für die Dauer von vier oder fünf Tagen zu präsentieren.

Worauf schon viele warten..... Compute mit jetzt im Abo

Roeske Verlag
Fuldaer Straße 6
3440 Eschwege

Raiffeisenbank Eschwege eG,
Bankleitzahl 522 603 85
Kto.-Nr. 245 8888

Auslandsabo nur per Euroscheck
(Unterschrift auf der Rückseite nicht vergessen)

Name

Straße

PLZ, Ort

Ich bezahle per Scheck oder Überweisung
☐ vierteljährlich à DM 39,-
☐ jährlich à DM 156,-

Mir ist bekannt, daß ich die Bestellung innerhalb von 8 Tagen widerrufen kann.

Unterschrift

Assemblerkurs Teil 6

Auch heute wieder eine Erweiterung für den Editor. Um ein Listing auf den Drucker ausgeben zu können, benutzen Sie bisher den Befehl "T". Damit bekamen Sie "aber" nur eine Hardcopy des Bildschirms. Um längere Assemblerlistings ausgeben zu können, war diese Routine ungeeignet. Diese neue Befehlsfolge erlaubt Ihnen, das reine Assemblerlisting auf den Drucker auszugeben.

Listing 1A

```
296 IF A$="LP" THEN GOSUB 15000:GOTO 2000
READY.

15000 OPEN 4,4:FORC=N1TON2:QC$=""
15010 QC$=RIGHT$(" "+STR$(C),3)+" "
15015 IF LEFT$(A$(C),1)=" " THEN QC$=QC$+" "
15020 FOR I=1 TO 8: I$=MID$(A$(C),I,1):QC$=QC$+I$: IF I$<>" " THEN NEXT I
15025 QC$=QC$+" "
15030 QC$=QC$+MID$(A$(C),I)
15060 PRINT#4,QC$:NEXT C:CLOSE 4:RETURN
READY.
```

Ab dieser Folge übrigens, werden wir immer ein oder mehrere Befehle ausführlich erklären, sodaß Sie in der Lage sind besser zu verstehen, wie ein Befehl wirkt.

Diesmal werden wir uns mit den Inkrementierbefehl (INC) befassen:
INC bedeutet Operand + 1

Welche Operanden sind möglich?

1. Eine absolut angegebene Speicherzelle

in MC-Code:

EE 2D 63 (hexadezimal)

High-Byte der Adresse

Low-Byte der Adresse

Befehlscode

das gleiche in Basic:

Poke 25389, Peek (25389)+1

2. Eine Speicherstelle auf der Seite 0 (das sind die Bytes mit den Adressen 0-255)

z.B. INC 30 (hexadezimal)

in MC-Code:

E6 1E

Adresse

Befehlscode

das gleiche in Basic:

Poke 30, Peek (30)+1

3. Eine absolut angegebene Speicherzelle + den Inhalt von X-Register (indiziert)

z.B.: INC 27934,X

in MC-Code:

FE 1E 6D

Highbyte der Adresse

Lowbyte der Adresse

Befehlscode

das gleiche in Basic:

Poke (27934+X), Peek (27934+X)+1

4. Eine Speicherstelle der Seite 0 + den Inhalt des X-Registers (indiziert)

z.B.: INC 27,X

in MC-Code:

F6 1B

Adresse

Befehlscode

das gleiche in Basic: Poke (27+X), Peek (27+X)+1

Außerdem werden das Zeroflag und das Vorzeichenflag beeinflusst. D.h., wird durch das Inkrementieren irgendeine Speicherzelle 0, so steht das Zeroflag auf 1, andernfalls auf 0. Das Vorzeichenflag ist eine Kopie des höchstwertigen Bits des inkrementierten Bytes.

Zurück zu unserem Kurs:

Diesmal ein etwas größeres Programm:

Listing 4A

```
1 WOLBY EQU 97
2 DFHBY EQU 648
3 ORG 828
4 LDA DFHBY
5 STA WOLBY+1
6 LDA #0
7 STA WOLBY
8 LDA #83
9 JSR ZEILE
10 LDX #23
11 JSR RAND
12 JSR ZEILE
13 RTS
14 ZEILE LDX #40
15 ZEIL1 STA (WOLBY),Y
16 INY
17 BNE WEIT
18 INC WOLBY+1
19 WEIT DEX
20 BNE ZEIL1
21 RTS
22 RAND STA (WOLBY),Y
23 PHA
24 TYA
25 CLC
26 ADC #39
27 TAY
28 BCC WEIT2
29 INC WOLBY+1
30 WEIT2 PLA
31 STA (WOLBY),Y
32 INY
33 BNE WEIT3
34 INC WOLBY+1
35 WEIT3 DEX
36 BNE RAND
37 RTS
```

Tippen Sie es in den Editor; save Sie es ab; assemblieren Sie es; und starten Sie es, **nachdem Sie den Bildschirm gelöscht haben**, mit Sys 828 **aber nur einmal**.

Sie sehen nun einen leeren Bildschirm, umrandet von Herzen und in der 3. Zeile steht Ready. Wäre das Programm nicht ausgegangen, dann würde das Ready auch fehlen.

Nun zum Erklären des Programmes:

Die ersten 2 Befehle:

WOLBY EQU 97

DFHBY EQU 648

Wir wissen schon, daß EQU ein Pseudo-Befehl ist, der einer Adresse einen Namen zuweist; nämlich in unserem Falle WOLBY (Wo Low Byte) = Adresse 97 DFHBY (D-File-High Byte) = Adresse 648

Labelnamen sind zwar beliebig, aber man sollte sie so wählen, daß erkenntlich wird, wozu sie da sind.

Zu DFHBY ist noch zu sagen, daß das Betriebssystem des Commodore in den Byte der Adresse 648 festlegt, wo der Bildschirm beginnt.

Die Bytes der Adressen 97-101 sind normalerweise für das Fließkomma-Akku #1 reserviert. Da wir in unserer Routine allerdings weder mit Fließkomma rechnen, noch ROM-Routinen benutzen, kann man dieses Feld ruhig als Speicherplatz in der Seite 0 (Zeropage) benutzen.

3. Befehl

ORG 828 dürfte mittlerweile bekannt sein. Übrigens beginnt bei 828 der Kasettenpuffer, der 191 Bytes lang ist.

Der 4. Befehl

LDA DFHBY bedeutet; lade den AKKU mit den Inhalt von DFHBY

Der 5. Befehl

STA WOLBY + 1 bedeutet; speichere den Inhalt von Akku in die Adresse WOLBY + 1 (=98)

Der 6. Befehl

LDA #0 bedeutet; lade den Akku mit 0 und der

7. Befehl

STA WOLBY bedeutet; speichere den Inhalt vom Akku nach WOLBY (= 97)

8. Befehl

LDA #83 bedeutet (siehe 6. Befehl) lade in den Akku 83 (= Code von einem Herz)

Der 9. Befehl

JSR ZEILE. Ein neuer Befehl, aber einfach zu erklären. Er ist wie in Basic der Befehl Gosub, nämlich "Springe zu einem Unterprogramm, von dem du mit RETURN wieder zurückkehren kannst". Mit dem Operand "ZEILE" geben wir die Adresse an, mit dem der 14. Befehl mit

dem Label "Zeile" beginnt. Den Return-Befehl kennen Sie auch schon – der RTS.

Wir benutzen ihn immer, um nach einem Maschinenprogramm zum Basicinterpreter zurückzukehren. Dies ist auch einfach zu erklären, denn der Basicinterpreter ist eigentlich nichts anderes, als ein Maschinenprogramm, von dem wir mit dem Sys-Befehl eine Maschinen-Unterroutine aufrufen können. Und von diesem Unterprogramm müssen wir natürlich mit RTS zurück zum Basicinterpreter.

Der 10. Befehl

LDX #23 ist ein Befehl den Sie kennen – lade das X-Register mit 23. Dieser Befehl gibt dem Unterprogramm "RAND", das mit dem nächsten Befehl aufgerufen wird die Mitteilung, daß es 23 mal eine Zeile Rand machen soll. Durch Minderung dieses Wertes kann auch eine kleine Umrundung erzielt werden.

Der 11. Befehl

JSR RAND. Dieser Befehl wurde beim 9. Befehl erklärt (Jump Subroutine).

Der 12. Befehl

JSR ZEILE siehe 9 und 11

Der 13. Befehl

RTS Das Programm ist zu Ende. Rücksprung zum Basicinterpreter.

Nun zu den verwendeten Unterprogramm ZEILE und RAND.

Zu **ZEILE**: Sobald dieses Unterprogramm aufgerufen wird, soll eine Zeile = 40 Herzen ausgegeben werden. Die Adresse der Anfangsposition (des 1. Herzens steht in Wolby und Wolby + 1) + den Y-Register. Im Akku steht, welches Zeichen verwendet werden soll.

Zum 14. Befehl

LDX #40. Keine Schwierigkeit. In das X-Register wird die Anzahl der auszugebenden Zeichen gebracht.

Der 15. Befehl

STA (WOLBY), Y wird zu der Adresse, die Wolby und das Y-Register angibt, der Wert von A(=Code eines Herzens) gespeichert.

Der 16. Befehl

INY Wie bei den INC-Befehlen, gibt es eine Erhöhung um 1. Diesmal allerdings das Y-Register. Falls dieses zu 0 werden sollte, muß man Wolby+1 um 1 er-

höhen. Dieses Problem wird mit den nächsten 2 Programmschritten erledigt.

Der 17. Befehl

BNE Weit (Branch on not equal).

Falls das Y-Register nicht 0 wird, dann wird die Erhöhung von Wolby+ einfach übersprungen

Der 18. Befehl

INC Wolby + 1. Das Byte der Adresse Wolby + 1 wird um 1 erhöht. In diesem Teil kommt man nur, wenn beim INY das Y-Register "0" wurde.

Der 19. Befehl

DEX: Das Register X wird dekrementiert. Dieser und der nächste Befehl sind dazu da, alle 40 Herzen auszugeben. Das DEX ändert nämlich auch das Zero-Flag, je nach Inhalt des X-Registers nach dem Dekrementieren.

Mit dem 20. Befehl

BNE ZEILE 1 wird je nachdem, ob sich in X-Register eine 0 befindet oder nicht, weitere Herzen ausgegeben.

Der 21. Befehl

Mit **RTS** wird zurückgesprungen; zum aufrufenden Programm.

Das **Unterprogramm RAND** ist etwas komplizierter:

Es soll folgendes tun: Im X-Register steht, in wieviel Zeilen links und rechts je ein Herz ausgegeben werden soll. Im A-Register steht, welches Zeichen (Herz).

Der 22. Befehl

STA (WOLBY), Y dürfte vom 15. Befehl her bekannt sein.

Der 23. Befehl

PHA. Ein neuer Befehl. Er heißt **PUSH AKKU**. Auf dem Stapel – ein von der CPU verwalteter RAM-Bereich – wird der Inhalt des Akku gespeichert.

Der 24. Befehl

TYA. Ein einfacher Befehl (Transfer Y to A). In das A-Register wird der Wert des Y-Registers kopiert.

Durch den 25. Befehl

CLC wird das Carry Flag auf Null gesetzt (Clear Carry). Wir müssen nämlich, um von der linken zur rechten Seite zu kommen, das Y-Register um 39 erhöhen. Dies erreichen wir durch den

26. Befehl

ADC #39 (Add with Carry Flag).

Würde man in den 25. Befehl das CLC vergessen, könnte es passieren, daß, wenn das Carry gesetzt ist, statt 39, 40 dazu addiert werden würde. Leider hat die 65XX-Familie nicht die Möglichkeit, ohne Carry zu addieren.

Der 27. Befehl

TAY befördert den Inhalt des Akku wieder in das Y-Register (Transfer Akku to Y-Register), wo es herkam.

Der TAY-Befehl ändert das Carry-Flag (Übertragsflag) **nicht**. Es ist aber eventuell durch den ADC-Befehl gesetzt worden; dann nämlich, wenn der Akku einen Übertrag durch die Addition bekommt, hatte (wenn Ergebnis größer als 255 war). Dann nämlich muß Wolby+1 wieder um 1 erhöht werden. Dieses Problem wird mit dem

28. Befehl

BCC WEIT2 (Branch on Carry Clear) gelöst: War **kein** Carry gesetzt, so überspringt er den INC-Befehl.

Der 29. Befehl

INC-Wolby+1 dürfte klar sein. Das Highbyte unseres Zeigers muß, wenn das Y-Register 255 überschreitet, um 1 erhöht werden.

30. Befehl

PLA auch ein neuer Befehl. Er bedeutet Pull Akku. Das heißt, daß das oberste Stapелеlement in den Akku copiert wird, also den Ursprungszustand wieder herstellen. Es ist der entgegengesetzte Befehl zu PHA. Mit dem

31. Befehl

STA (Wolby),Y wird, da unsere beiden Zeiger nun auf die rechte Seite deuten, das Herz (Inhalt des A-Registers) in den Bildschirmbereich hineingebracht. Danach wird mit dem

32. Befehl

INY das Y-Register um 1 erhöht, um von der rechten Seite eine Zeile tiefer auf den linken Rand zu zeigen.

Der 33. und 34. Befehl

BNE WEIT 3 bzw. **INC WOLBY +1** müßte nun jedem klar sein. Mit dem

35. Befehl

DEX wird das X-Register, das angibt, wie oft diese Prozedur wiederholt werden soll, um 1 erniedrigt. Dieser DEX-Befehl setzt, wenn das X-Register zu Null werden sollte, automatisch das Zero-Flag. Durch den

36. Befehl

BNE RAND wird das Unterprogramm RAND so oft wiederholt, bis alle 23 Zeilen abgearbeitet sind.

Der 37. Befehl

RTS sorgt dafür, daß wir wieder in unserem Hauptprogramm landen.

Da dieses Programm ziemlich lang und für Sie vielleicht unüberschaubar war, befassen wir uns im nächsten Kurs noch einmal damit. Außerdem werden wir dieses Programm noch erweitern, so daß 1. vorher der Bildschirm gelöscht wird und 2. daß man das Programm öfter benutzen kann.

Überlegen Sie sich bis zum nächsten Mal, wo der Fehler liegt, daß es nur 1 mal benutzt werden kann.

(hf)

TV-Ecke

Computer im Fernsehen

12.10.84
Microprozessoren-
Microcomputer Teil I
4. Struktur eines
Computers

SW3
21.15 h

19.10.84
Microprozessoren-
Microcomputer Teil I
5. Wie ein Chip
entsteht

SW3
21.15 h

Liebe Freunde von COMPUTE MIT....!

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

Korrektur zu Olivaro, C-64 aus Compute mit 40

```
1000 REM BASICLADER
1010 FOR I=51200 TO 51254: READ X: POKE I, X: NEXT I
1020 REM DATA MASCHINENPROGRAMM
1030 DATA 120,165,1,72,41,251,133,1,169,208,133,3
1040 DATA 169,48,133,5,160,0,132,2,132,4,162,32
1050 DATA 177,2,145,4,200,208,249,230,3,230,5,202
1060 DATA 208,242,104,133,1,173,24,208,41,241,9,12
1070 DATA 141,24,208,88,162,128,96
1080 SYS 51200: REM ZEICHEN UMFÄHIGKEITEN
1090 FOR X=0 TO 135: READ T: POKE 13312+X, T: NEXT T
1100 REM ZEICHEN-DATAS
```

CRAZY CAR

für den Commodore 64

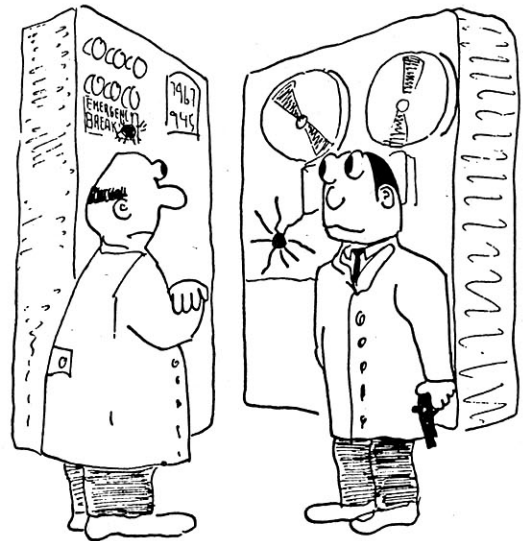
Wenn Sie das Spiel mit "RUN" gestartet haben, müssen Sie ein paar Minuten warten. Danach erscheint das Titelbild. Jetzt müssen Sie den Joystick bewegen, um weiter zu kommen. Wenn Sie das getan haben, erscheint das Spielbild. Dann geht es los. Sie bekommen nur Punkte, wenn das Spielometer grün ist

(ca. bei 215 Stundenkilometer). Es gibt keine Punkte mehr, wenn Sie schneller fahren, Sie sollten jedoch immer etwas über den 215 km/h liegen. In den Highscore kommen Sie nur, wenn Sie über 10000 Punkte haben. Danach können Sie "Fire" drücken, um noch ein Spiel zu bekommen.

```

10 REM =====
20 REM - THIS IS 'J' SOFTWARE -
25 REM - PRESENTING : -
35 REM - >>>> RALLY <<<< -
45 REM =====
75 REM =====
80 POKE53281,6 :POKE53280,6 :PRINT"J"
90 PRINT"***** PLUG JOYSTICK IN PORT 2 ***"
99 :
100 REM >>>>>> START OF GAME <<<<<<<<
101 :
110 READL
120 READA$
130 C=LEN(A$)
140 IFA$="*"THEN300
160 A=ASC(A$)-48
170 B=ASC(RIGHT$(A$,1))-48
180 N=B+7*(B>9)-(C=2)*(16*(A+7*(A>9)))
200 POKEL,N
210 L=L+1
220 GOTO120
289 :
290 REM >>>>>> SPRITE SHAPES <<<<<<<<<<<<
291 :
300 FORN=0TO62:READQ:POKE15*64+N,Q:NEXT
310 FORN=0TO62:READQ:POKE13*64+N,Q:NEXT
320 FORN=0TO62:READQ:POKE11*64+N,Q:NEXT
330 FORN=0TO62:READQ:POKE14*64+N,Q:NEXT
340 FORN=0TO62:READQ:POKE255*64+N,Q:NEXT
399 :
400 REM >>>>>> KARAK. GENERATOR <<<<<<
401 :
410 POKE56334,PEEK(56334)AND254:POKE1,PEEK(1)AND251
415 FORT=0TO2048:POKE12288+T,PEEK(53248+T):NEXT
430 READQ:IFQ=-1THEN450
440 FORI=0TO7:READZ:POKE12288+Q*8+I,Z:NEXT:GOTO430
450 POKE1,PEEK(1)OR4:POKE56334,PEEK(56334)OR1
460 POKE53272,(PEEK(53272)AND240)+12
499 :
500 REM >>>>>> STRINGE DEFINA. <<<<<<<<
501 :
510 DIMA$(25):FORI=1TO25:A$(I)="  " :NEXT
520 READA$(1):READA$(2):READA$(3):READA$(4):READA$(9):READA$(14):READA$(19)

```



Was sollte ich machen ?
Er wußte zuviel !

programme

```

530 READ$(24)=READ$(25)
540 FORI=1TO5:B*(I)=" 10000 CBM":NEXT
545 FORI=1TO3:READC$(I):NEXT:GOTO700
549 :
550 REM >>>>> VARIABLES DEFINE. <<<<<
551 :
560 POKE53003,163:POKE53004,24:POKE252,2:POKE253,7:POKE53006,70:POKE53007,2
570 FORT=52735T052994:POKET,INT(150*RND(1))+95:NEXT:POKE53012,3:POKE53013,4
599 :
600 REM >>>>>>> SPRITES <<<<<<<<<<
601 :
610 V=53248:FORT=2042T02047:POKET,14:NEXT:POKE2040,15
620 POKEV+39,6:POKEV+41,5:POKEV+42,7:POKEV+43,3:POKEV+44,1
630 POKEV+45,9:POKEV+46,12:POKEV+23,0:POKEV+29,0
640 POKE53248,173:POKE53249,200:POKE53252,150
645 POKE53253,30:POKE53254,196
650 POKE53255,65:POKE53256,128:POKE53257,190:POKE53258,221
660 POKE53260,247:POKE53261,140:POKE53262,95:POKE53263,110:POKE53259,235
670 RETURN
699 :
700 REM >>>>>>> FRONT PAGE <<<<<<<<<
701 :
710 PRINT"<";:POKE53281,0:POKE53280,9
720 PRINT"XXXXXXXXXX< "
730 PRINT"XXXXXXXXXX< "
740 PRINT"XXXXXXXXXX< "
750 PRINT"XXXXXXXXXX< "
760 PRINT"XXXXXXXXXX< "
770 PRINT"XXXXX< "
780 PRINT"XXXXX< "
790 PRINT"XXXXX< "
800 PRINT"XXXXX< "
810 PRINT"XXXXX< "
820 PRINT"XXXXX< "
830 PRINT"XXXXX< "
840 PRINT"XXXXX< "
850 PRINT"XXXXX< "
860 PRINT"XXXXX< "
870 PRINT"XXXXX< "
880 PRINT"XXXXX< "
890 PRINT"XXXXX< "
900 PRINT"XXXXXXXXXX< " BY JACOB NIELSEN 1984"
910 PRINT"XXXXXXXXXX< "
920 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXX< "
930 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXX< "
940 IFPEEK(56320)=127THEN940
950 POKE54276,0:POKE54277,15:POKE54278,0:POKE54296,15:POKE54276,17
960 FORT=0T0255STEP30:POKE54273,T:FORI=1TO20:NEXT:NEXT
970 FORT=255T055STEP-30:POKE54272,T NEXT:POKE54276,0
999 :
1000 REM >>>>>>> GAME <<<<<<<<<<<<
1001 :
1010 GOSUB550:PRINT"<";:POKE53281,11:POKE53280,14
1020 FORI=1TO25:PRINT"X ";
1030 PRINTA$(I):NEXT
1040 FORT=1978T01898STEP-40:POKET,160:NEXT:FORT=55450T055650STEP40:POKET,5:NEXT
1050 POKEV+21,255:POKE2023,160:POKE56295,6
1060 FORI=1TO3:FORT=1T01000:NEXT:PRINT"XXXXXXXXXXXXXXXXC*(I)":NEXT
1070 REM >>>>>>> START SKUD <<<<<<<<<<
1080 POKE54273,35:POKE54272,180:POKE54276,0:POKE54276,129
1090 FORI=15T00STEP-1:POKE54296,I:POKE54277,15:NEXT
1095 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXX"
1099 :
1100 REM>>>>>>> START SHUT <<<<<<<<<<
1101 :
1110 POKE54277,0:POKE54278,128:POKE54273,3:POKE54272,2:POKE54296,15
1120 POKE54274,2:POKE54275,2:POKE54276,0:POKE54276,65
1130 POKE53008,0:POKE53009,0:POKE53010,0:POKEV+30,0:SYS50000:REM + + + + GAME
1490 :
1495 REM >>>>> EXPLOSION <<<<<<<<<<
1497 :
1500 POKEV+39,9:POKE2040,255:POKE54276,0:POKEV+23,1:POKEV+29,1
1510 POKE54273,1:POKE54272,15:POKE54277,15:POKE54278,0:POKE54276,129
1520 FORT=15T01STEP-1:POKE54296,T:FORI=1TO8:POKEV+39,4:POKEV+39,11:POKEV+39,2
1530 POKE54272,3:NEXT:NEXT:POKE54276,0:POKEV+21,0
1540 FORT=1T01000:NEXT
1990 :
1995 REM >>>>>>> HI-SCORES <<<<<<<<<<
1997 :
2000 PRINT"<";:POKE53281,2:POKE53280,2
2010 FORT=1T024:PRINT"XX";
2020 PRINT"<";:FORT=1T03:PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";:NEXT
2030 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";
2040 FORT=1T02:PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";:NEXT
2100 SC=PEEK(53008)+256*PEEK(53009)+65536*PEEK(53010)
```



```

2105 SC$=STR$(SC)
2110 K=0:FOR I=1 TO 5:IF SC>VAL(LEFT$(B$(I),7)) THEN K=I:I=5
2120 NEXT I:IF K=0 THEN 2190
2130 PRINT "*****YOUR SCORE "SC$" IS NUM."K"ON":PRINT "*****THE TOP FIVE!"
2135 POKE54276,0:POKE54272,0:POKE54273,0:POKE54277,0:POKE54278,240:POKE54276,33
2140 POKE54296,15:FOR L=2 TO 17:FOR L1=1 TO 15:POKE54273,L*L1:NEXT L1,L:POKE54276,0
2150 PRINT "*****ENTER YOUR INITIALS "":OPEN#1,0:INPUT#1,I$:CLOSE 1:PRINT
2160 FOR I=4 TO K STEP -1:B$(I+1)=B$(I):NEXT B$(K)=SC$+" "+LEFT$(I$,3)
2170 FOR I=1 TO 5:PRINT "*****I "I"="LEFT$(B$(I),7),RIGHT$(B$(I),3):NEXT I:GOTO 2200
2190 PRINT "*****YOUR SCORE "SC$" IS NOT":PRINT "*****ON THE TOP FIVE"
2200 PRINT "*****PRESS FIRE BUTTON FOR ANOTHER GO"
2210 FORT=1 TO 500:IF PEEK(56320)=127 THEN NEXT:GOTO 700
2230 PRINT "J":FOR L=54276 TO 54296:POKE L,0:NEXT
2240 POKE54273,130:POKE54277,9:POKE54287,30:POKE54296,15
2250 POKE54276,21:FORT=1 TO 1000:NEXT:POKE54276,20:GOTO 1000
49996 :
49997 REM >>>> MASHINCODE DATA <<<<<<<
49998 :
50000 DATA 49991,20,55,C5,20,EA,C4,20,C6,C4
50005 DATA EE,0A,CF,AD,00,DC,A0,00,A2,00,4A,B0,01,88,4A,B0,01,C8,4A,B0,01
50010 DATA CA,4A,B0,01,E8,4A,8E,08,CF,8C,09,CF,AD,00,D0,18,6D,08,CF,C9,FC,F0,07
50020 DATA C9,5C,F0,03,4C,84,C3,60,8D,00,D0
50030 DATA AD,1E,D0,29,01,D0,F5
50040 DATA AD,08,CF,F0,0C,C9,01,F0,10,A9,0D,8D,F8,07,4C,AC,C3,A9,0F,8D,F8,07
50050 DATA 4C,AC,C3,A9,0B,8D,F8,07
50060 DATA AD,0E,CF,8D,02,D4,AD,0F,CF,8D,03,D4
50070 DATA A2,0E,AC,0B,CF,88,D0,FD,CA,D0,F7
50080 DATA A2,05,BD,00,D0,C9,F1,F0,0F,18,69,01,9D,00,D0,E8,E8,E0,11,D0,ED,4C,F1
50085 DATA C3
50090 DATA A9,01,9D,00,D0,AC,0A,CF,B9,00,CE,CA,9D,00,D0,E8,E8,E0,11,D0,D4
50100 DATA AD,09,CF,F0,02,D0,03,4C,47,C3,18,AD,0B,CF,6D,09,CF,C9,14,F0,07,C9,BB
50105 DATA F0
50110 DATA 03,8D,0B,CF,A9,20,A0,00,91,FC,84,FC,A9,07,85,FD,A2,08,A9,00,9D,57,31
50120 DATA CA,D0,FA,AD,09,CF,C9,01,F0,33,AD,0C,CF,18,69,01,C9,A8,F0,03,8D,0C,CF
50130 DATA 20,8E,C4,AD,0E,CF,18,69,03,8D,0E,CF,B0,03,4C,47,C3,AD,0F,CF,C9,03,F0
50135 DATA 06
50140 DATA EE,0F,CF,4C,47,C3,A9,F5,8D,0E,CF,4C,47,C3,AD,0C,CF,38,E9,01,F0,03,8D
50145 DATA 0C
50150 DATA CF,20,8E,C4,AD,0E,CF,38,E9,03,8D,0E,CF,90,03,4C,47,C3,AD,0F,CF,C9,02
50160 DATA F0,06,CE,0F,CF,4C,47,C3,A9,0A,8D,0E,CF,4C,47,C3
50170 DATA AE,0C,CF,A0,BA,E0,08,B0,02,90,17,A9,A0,91,FC,98,38,E9,28,A0,90,02,B0
50175 DATA 02
50180 DATA C6,FD,8A,38,E9,08,AA,4C,93,C4,A9,2B,91,FC,84,FC,A0,0A,A9,FF,E0,00,F0
50185 DATA 07
50190 DATA 99,55,31,88,CA,D0,F5,60
50200 DATA AD,0B,CF,C9,45,90,01,60,AD,10,CF,18,69,0A,8D,10,CF,F0,01,60,AD,11,CF
50210 DATA 18,69,01,8D,11,CF,B0,01,60,EE,12,CF,60
50220 DATA AD,14,CF,C9,03,F0,04,EE,14,CF,60,A9,01,8D,14,CF,EE,15,CF,AD,15,CF
50230 DATA C9,01,F0,1E,C9,02,F0,2B,C9,03,F0,38,A9,00,8D,15,CF,A9,E7,8D,5C,32
50240 DATA 8D,5D,32,A9,FF,8D,5E,32,8D,5F,32,60,A9,E7,8D,5E,32,8D,5F,32,A9,FF
50250 DATA 8D,58,32,8D,59,32,60,A9,E7,8D,58,32,8D,59,32,A9,FF,8D,5A,32,8D,5B,32
50260 DATA 60,A9,E7,8D,5A,32,8D,5B,32,A9,FF,8D,5C,32,8D,5D,32,60
50270 DATA AD,00,DC,C9,6F,F0,01,60,A5,C5,C9,04,F0,F9,4C,5D,C5,*
50294 :
50295 REM >>>> SPRITE DATA <<<<<<<<<<
50296 :
50300 DATA 3,24,192,3,60,192,3,255,192,3,255,192,3,60,192,3,24,192,0,24,0,0,60,0
50310 DATA 0,60,0,0,126,0,0,126,0,0,255,0,0,255,0,28,231,56,28,195,56,28,219,56
50320 DATA 31,219,248,31,195,248,28,255,56,28,126,56,28,0,56
50330 DATA 12,25,192,14,60,192,7,255,224,7,255,224,3,60,112,3,152,48,0,24,0,0,60
50360 DATA 0,0,60,0,0,126,0,0,126,0,0,255,0,0,255,0,28,231,56,28,195,56,28,219,56
50370 DATA 51,219,248,31,195,248,28,255,56,28,126,56,28,0,56
50400 DATA 3,152,48,3,60,112,7,255,224,7,255,224,14,60,192,12,25,192,0,24,0,0,60
50410 DATA 0,0,60,0,0,126,0,0,126,0,0,255,0,0,255,0,28,231,56,28,195,56,28,219,56
50420 DATA 31,219,248,31,195,248,28,255,56,28,126,56,28,0,56
50430 DATA 28,0,56,28,126,56,28,255,56,31,195,248,31,219,248,28,219,56
50440 DATA 28,195,56,28,231,56,0,255,0,0,255,0,0,126,0,0,126,0,0,60,0,0,60,0
50450 DATA 0,24,0,3,24,192,3,60,192,3,255,192,3,255,192,3,60,192,3,24,192
50460 DATA 0,16,4,0,16,12,0,56,24,0,124,56,0,254,120,3,238,208,31,199,176
50470 DATA 127,128,32,30,27,48,15,62,96,3,31,48,1,208,24,1,129,140,3,51,236
50480 DATA 3,123,62,7,239,14,7,206,3,15,132,0,15,4,0,6,0,0,4,0,0
50500 :
50502 :
50505 REM >>>> NEW KARAK. GENERATOR <<
50507 :
50509 :
50510 DATA 1,60,36,126,98,98,98,98,0
50520 DATA 2,124,36,62,50,50,50,126,0
50530 DATA 3,126,66,64,96,96,98,126,0
50540 DATA 4,126,34,34,50,50,50,126,0
50550 DATA 5,126,64,64,120,96,96,126,0
50560 DATA 6,126,64,64,126,96,96,96,0
50570 DATA 7,126,66,64,110,98,98,126,0
50580 DATA 8,66,66,66,126,98,98,98,0
50590 DATA 9,16,16,16,24,24,24,24,0
50600 DATA 10,2,2,2,6,6,70,60,0
50610 DATA 11,66,68,72,126,98,98,98,0
50620 DATA 12,64,64,64,96,96,96,126,0
50630 DATA 13,102,90,66,98,98,98,98,0
50640 DATA 14,114,74,74,106,106,106,102,0
50650 DATA 15,126,66,66,98,98,98,126,0
50660 DATA 16,126,66,66,126,96,96,96,0
50670 DATA 17,126,66,66,98,106,102,122,0
50680 DATA 18,126,66,66,126,100,98,98,0
50690 DATA 19,126,66,64,126,6,70,126,0

```

```

50700 DATA20,124,16,16,24,24,24,24,0
50710 DATA21,66,66,66,98,98,98,124,0
50720 DATA22,66,66,66,36,36,24,24,0
50730 DATA23,98,98,98,98,66,90,102,0
50740 DATA24,66,36,24,124,98,98,98,0
50750 DATA25,66,66,66,60,24,24,24,0
50760 DATA26,126,2,28,96,96,96,126,0
50770 DATA28,60,66,153,161,161,153,66,60
50780 DATA48,126,66,70,106,114,98,126,0
50790 DATA49,56,8,24,24,24,60,0
50800 DATA50,126,66,2,126,96,96,126,0
50810 DATA51,62,2,2,30,6,6,126,0
50820 DATA52,64,64,68,68,126,12,12,0
50830 DATA53,62,32,32,62,6,70,126,0
50840 DATA54,126,64,64,126,98,98,126,0
50850 DATA55,126,66,4,8,24,24,24,0
50860 DATA56,60,36,36,126,98,98,126,0
50870 DATA57,126,66,66,126,6,6,126,0
50880 DATA73,254,255,255,255,255,255,255
50890 DATA74,127,255,255,255,255,255,255
50900 DATA170,243,18,18,243,243,130,130,243
50910 DATA171,207,73,73,201,201,73,73,207
50920 DATA172,120,9,8,120,120,64,64,121
50930 DATA173,143,136,136,143,143,129,129,207
50940 DATA174,35,98,34,35,35,32,32,115
50950 DATA175,207,9,9,201,201,73,73,207
50960 DATA176,243,146,146,147,147,146,146,243
50970 DATA177,207,72,72,207,207,65,65,207
50980 DATA178,243,144,144,147,147,146,146,243
50990 DATA85,127,127,127,127,127,127,127
51000 DATA88,254,254,254,254,254,254,254
51010 DATA90,127,127,127,31,31,127,127,127
51020 DATA75,231,231,231,231,231,231,255,255
51030 DATA179,207,73,73,201,201,9,9,207
51050 DATA109,0,255,255,255,255,255,255,255
51110 DATA30,204,204,51,51,0,0,0,0
51120 DATA31,12,12,3,3,12,12,3,3
51130 DATA-1
51500 DATA"" ","%KM/H"% "%,%
51510 DATA"" %M G=3L-G ""%,% %M G=3L/%
51520 DATA"" %M G=3R2G ""
51530 DATA"" M,L,G ""
51540 DATA""FASTEN BELTS "","%N
51550 DATA""GO !!! ""
```

Horseguard

für den Commodore 64

Programmablauf:

1. Laden des Vorprogrammes mit LOAD von Cassette bzw. LOAD "HORSE-GUARD 1", 8 von Diskette.

2. Nach RUN wird gefragt, ob das Hauptprogramm von Cassette oder von Diskette nachgeladen werden soll. Nach entsprechender Eingabe werden die Daten für das Maschinenprogramm und für das Musikprogramm eingelesen. Dann wird das Hauptprogramm automatisch nachgeladen und selbst gestartet.

Nach dem Einlesen der DATAs für Sprites und den neuen Zeichensatz, währenddem ein Pferd über den Bildschirm läuft und eine 2-stimmige Version von "Oh, Susanna!" ertönt, erscheint die Anleitung

für das Spiel:

Sie müssen versuchen, mit Ihrem Cowboy möglichst viele Pferde einzufangen. Hierzu steht Ihnen ein Lasso zur Verfügung, das Sie nach rechts oben, rechts und rechts unten werfen können (Drücken des Feuerknopfes und des Joysticks in die entsprechende Richtung). Es wird der Joystick in Port 2 benutzt. Um das Spiel zu erschweren, können Sie Ihren Cowboy nur in der linken Bildschirmhälfte bewegen, und zwar in alle Richtungen, auch diagonal. Am Boden liegen außerdem noch zwei ausgebleichte Rinderschädel, bei deren Berührung Sie eines Ihrer drei Leben verlieren. Sie verlieren auch ein

Leben, wenn Sie eines der langsameren Pferde, also die, die nach links laufen, berühren. Alle 500 Punkte wird das Spiel schneller, das insgesamt 10 verschiedene Geschwindigkeiten (Levels) hat. Zur Punktezahl: Um ein Pferd einzufangen, müssen Sie es genau am Hals treffen. Dann gibt's 10 Punkte plus die Entfernung zwischen dem Cowboy und dem getroffenen Pferd, d.h. je größer der Abstand zum gefangenen Pferd ist, desto mehr Punkte gibt es. Nach Verlust aller 3 Leben kann ein neues Spiel per Knopfdruck begonnen werden; jedes Spiel beginnt mit der niedrigsten Geschwindigkeit (Level 0).

VORPROGRAMM

```

0 REM *****
1 REM *
2 REM * HORSEGUARD - VORPROGRAMM *
3 REM *
4 REM *****
5 REM *
6 REM * (C) 1984 BY A. KNIPSCHILD *
7 REM *
8 REM *****
9 :
10 PRINT"COMMON CASSETTE ODER DISKETTE NACHLADEN ?"
11 POKE198,0:WAIT198,1:GETDV$:IFDV$<>"C"ANDDV$<>"D"THEN11
12 IFDV$="C"THENNDV$="1"
13 IFDV$="D"THENNDV$="8"
15 PRINT"BITTE WARTEN, DATAS WERDEN EINGELESEN !"
20 FORA=16384TO18428:READB:C=C+B:POKEA,B:NEXT
30 IFNDV$=241835THENPRINT"FEHLER IN SPIELPGM.-DATAS":END
40 FORA=20480TO21436:READB:D=D+B:POKEA,B:NEXT
50 IFNDV$=73412THENPRINT"FEHLER IN MUSIKPGM.-DATAS":END
60 PRINT"COMP43,1:PF44,16:PF4096,0:NEW"
61 PRINT"MULT"CHR$(34)"HORSEGUARD 2"CHR$(34)", "DV$";
62 FORI=0TO5:POKE631+I,13:NEXT
63 POKE637,82:POKE638,213:POKE639,58:POKE640,13
64 POKE198,10:END
93 :
94 REM *****
95 REM *
96 REM * SPIEL-MASCHINENPROGRAMM *
97 REM *
98 REM *****
99 :
100 DATA173,57,192,240,1,96,173,58,192,240,3,32,48,70,169,255,141,57,192
101 DATA76,80,67,173,1,192,201,3,208,23,169,0,141,1,192,173,250,7,201,242
102 DATA240,6,238,250,7,76,52,64,169,235,141,250,7,238,1,192,234,173,3,192
103 DATA201,111,208,4,76,86,64,234,173,4,192,201,16,208,5,169,0,141,4,192
104 DATA32,96,64,76,229,64,0,32,48,68,76,229,64,0,0,0,169,224,141,2,220
105 DATA173,0,220,141,2,192,169,255,141,2,220,173,2,192,41,1,208,3,32,165
106 DATA64,173,2,192,41,2,208,3,32,181,64,173,2,192,41,4,208,3,32,197,64
107 DATA173,2,192,41,8,208,3,32,213,64,173,2,192,41,16,208,3,32,0,64,96,0
108 DATA0,173,5,208,201,107,240,6,56,233,1,141,5,208,96,0,0,173,5,208,201
109 DATA196,240,6,24,105,1,141,5,208,96,0,0,173,4,208,201,26,240,6,56,233
110 DATA1,141,4,208,96,0,0,173,4,208,201,92,240,6,24,105,1,141,4,208,96,0
111 DATA0,173,5,192,201,1,208,56,169,0,141,5,192,173,200,4,201,79,208,5,169
112 DATA80,76,23,65,173,200,4,201,80,208,5,169,114,76,23,65,173,200,4,201
113 DATA114,208,5,169,115,76,23,65,169,79,162,0,157,200,4,157,72,7,232,224
114 DATA80,208,245,238,5,192,234,173,250,7,24,105,9,201,251,208,2,169,243
115 DATA141,251,7,234,162,0,189,251,7,24,105,1,201,251,208,2,169,243,157
116 DATA252,7,232,224,4,208,236,234,162,0,189,6,192,208,109,189,32,192,157
117 DATA6,192,173,16,208,61,16,192,141,64,3,234,189,7,192,234,208,91,254
118 DATA6,208,173,64,3,240,63,189,6,208,201,90,208,72,56,169,255,253,16,192
119 DATA141,64,3,173,16,208,45,64,3,141,16,208,169,0,157,6,208,234,157,7
120 DATA192,234,234,32,250,65,234,234,234,234,173,27,212,201,107,48,249,201
121 DATA196,240,245,176,243,157,7,208,76,195,65,0,189,6,208,201,255,208,9
122 DATA173,16,208,29,16,192,141,16,208,76,240,65,0,222,6,208,173,64,3,208
123 DATA9,189,6,208,208,28,76,141,65,0,189,6,208,208,19,56,169,255,253,16
124 DATA192,141,64,3,173,16,208,45,64,3,141,16,208,234,232,232,224,10,240
125 DATA20,76,81,65,234,24,173,27,212,41,3,105,1,157,6,192,157,32,192,96
126 DATA0,162,0,222,6,192,232,232,224,10,208,247,234,32,128,71,234,234,234
127 DATA234,234,234,234,234,234,189,16,192,141,64,3,173,16,208,45,64,3,240
128 DATA13,189,6,208,201,5,48,6,169,128,157,7,192,234,76,144,70,234,234,234
129 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
130 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
131 DATA234,234,234,234,234,173,48,192,201,8,208,87,165,247,197,249,208,47
132 DATA165,248,197,250,208,41,160,0,169,32,145,247,200,145,247,173,27,212
133 DATA41,254,201,28,240,247,176,245,170,189,64,192,133,247,189,65,192,133
134 DATA248,189,96,192,133,249,189,97,192,133,250,169,0,141,48,192,56,165
135 DATA247,233,1,133,247,165,248,233,0,133,248,160,0,169,73,145,247,200
136 DATA169,74,145,247,200,169,32,145,247,238,48,192,234,173,49,192,201,8
137 DATA208,87,165,251,197,253,208,47,165,252,197,254,208,41,160,0,169,32
138 DATA145,251,200,145,251,173,27,212,41,254,201,28,240,247,176,245,170
139 DATA189,64,192,133,251,189,65,192,133,252,189,96,192,133,253,189,97,192
140 DATA133,254,169,0,141,49,192,56,165,251,233,1,133,251,165,252,233,0,133
141 DATA252,160,0,169,73,145,251,200,169,74,145,251,200,169,32,145,251,238
142 DATA49,192,234,169,0,141,31,208,173,31,208,41,4,240,7,169,2,141,80,3
143 DATA96,0,76,208,69,0,0,0,0,0,0,0,0,0,0,0,169,111,141,3,192,173,2,192
144 DATA41,26,240,20,173,2,192,41,25,240,77,173,2,192,41,24,208,3,76,240
145 DATA67,76,167,67,169,254,141,249,7,24,173,4,208,105,21,141,2,208,24,173
146 DATA5,208,105,7,141,3,208,24,173,4,208,105,24,141,0,208,24,173,5,208
147 DATA105,11,141,1,208,169,3,141,51,192,169,0,141,52,192,96,0,0,0,169,0
148 DATA141,3,192,96,0,0,0,169,252,141,249,7,24,173,4,208,105,15,141,2,208
149 DATA56,173,5,208,233,21,141,3,208,24,173,4,208,105,18,141,0,208,56,173

```



```

150 DATA5,208,233,25,141,1,208,169,1,141,51,192,169,0,141,52,192,96,0,0,0
151 DATA0,0,0,0,0,0,0,0,169,253,141,249,7,24,173,4,208,105,24,141,2,208
152 DATA56,173,5,208,233,6,141,3,208,24,173,4,208,105,32,141,0,208,56,173
153 DATA5,208,233,4,141,1,208,169,2,141,51,192,169,0,141,52,192,96,0,0,0
154 DATA0,0,0,0,0,0,0,0,173,52,192,201,2,240,3,76,224,68,169,0,141,52,192
155 DATA173,51,192,201,1,208,3,76,96,68,173,51,192,201,2,208,3,76,128,68
156 DATA173,51,192,201,3,208,102,76,160,68,0,0,0,56,173,0,208,237,4,208,201
157 DATA30,208,3,76,192,68,238,0,208,206,1,208,76,224,68,0,0,0,0,0,0,0
158 DATA0,56,173,0,208,237,4,208,201,48,208,3,76,192,68,238,0,208,76,224
159 DATA68,0,0,0,0,0,0,0,0,56,173,0,208,237,4,208,201,36,208,3,76
160 DATA192,68,238,0,208,238,1,208,76,224,68,0,0,0,0,0,0,0,0,0,0,169,0,141
161 DATA0,208,141,1,208,141,3,192,141,2,208,141,3,208,96,0,0,0,0,0,0,0,0
162 DATA0,0,0,0,0,0,162,0,173,16,208,61,16,192,221,16,192,208,3,76,70,69
163 DATA189,6,208,205,0,208,144,3,76,70,69,56,173,0,208,253,6,208,141,64
164 DATA3,173,64,3,201,16,144,3,76,70,69,169,1,205,64,3,144,3,76,70,69,232
165 DATA173,1,208,221,6,208,144,3,76,70,69,56,189,6,208,237,1,208,141,64
166 DATA3,173,64,3,201,10,144,3,76,70,69,169,2,205,64,3,144,3,76,70,69,76
167 DATA96,69,232,224,10,240,3,76,226,68,238,52,192,96,0,0,0,0,0,0,0,0
168 DATA0,0,0,0,0,202,142,53,192,138,74,170,189,42,208,141,40,208,169,251
169 DATA141,249,7,169,0,141,0,208,141,1,208,141,3,192,174,53,192,189,6,208
170 DATA141,2,208,189,7,208,141,3,208,169,1,141,54,192,169,255,141,57,192
171 DATA141,58,192,32,144,71,234,234,234,234,234,234,234,234,234,234,234
172 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
173 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
174 DATA76,129,70,0,0,0,173,58,192,208,3,76,250,69,173,54,192,240,3,76,247
175 DATA69,169,1,141,54,192,173,2,208,240,6,206,2,208,76,247,69,32,48,70
176 DATA76,250,69,0,206,54,192,173,2,192,41,16,240,5,169,0,141,57,192,169
177 DATA144,133,97,173,56,192,133,98,173,55,192,133,99,169,0,133,100,133
178 DATA101,32,221,189,162,0,189,1,1,240,7,157,34,4,232,76,31,70,76,112,70
179 DATA0,0,169,0,141,58,192,174,53,192,169,0,157,6,208,173,27,212,201,107
180 DATA48,249,201,196,240,245,176,243,157,7,208,173,27,212,41,3,105,1,157
181 DATA6,192,157,32,192,169,255,141,21,208,169,0,141,40,208,157,7,192,141
182 DATA2,208,141,3,208,96,0,0,162,246,160,0,200,208,253,232,208,248,76,0
183 DATA71,0,0,0,0,169,0,157,6,208,157,7,208,96,0,0,0,0,0,0,162,0,189,7,208
184 DATA205,5,208,48,14,56,189,7,208,237,5,208,201,18,48,17,76,241,70,56
185 DATA173,5,208,253,7,208,201,18,48,3,76,241,70,173,16,208,61,16,192,240
186 DATA3,76,241,70,189,6,208,205,4,208,48,14,56,189,6,208,237,4,208,201
187 DATA21,48,17,76,241,70,56,173,4,208,253,6,208,201,21,48,3,76,241,70,189
188 DATA7,192,240,7,169,1,141,80,3,96,234,232,232,224,10,208,155,76,106,66
189 DATA0,0,0,0,0,0,173,59,192,201,5,208,105,169,0,141,59,192,169,128,141
190 DATA4,212,173,60,192,201,5,208,18,169,1,141,60,192,169,50,141,1,212,169
191 DATA129,141,4,212,76,112,71,173,60,192,201,1,208,18,169,2,141,60,192
192 DATA169,100,141,1,212,169,129,141,4,212,76,112,71,173,60,192,201,2,208
193 DATA18,169,3,141,60,192,169,200,141,1,212,169,129,141,4,212,76,112,71
194 DATA173,60,192,201,4,208,8,169,4,141,60,192,76,112,71,169,5,141,60,192
195 DATA38,59,192,56,173,113,70,233,198,141,78,4,76,22,64,0,173,27,212,201
196 DATA9,240,249,176,247,74,10,170,96,0,0,0,56,189,6,208,237,4,208,141,64
197 DATA3,24,173,64,3,105,10,141,64,3,24,173,55,192,109,64,3,141,55,192,173
198 DATA56,192,105,0,141,62,192,24,173,61,192,109,64,3,141,61,192,173,62
199 DATA192,105,0,141,62,192,173,62,192,201,1,240,7,201,2,240,15,76,252,71
200 DATA173,61,192,201,244,240,5,176,3,76,252,71,56,173,61,192,233,244,141
201 DATA61,192,173,62,192,233,1,141,62,192,173,113,70,201,255,240,3,238,113
202 DATA70,96
493
494 REM *****
495 REM *
496 REM * MUSIK-MASCHINENPROGRAMM *
497 REM *
498 REM *****
499 :
500 DATA169,226,133,247,169,80,133,248,169,65,133,249,169,82,133,250,169
501 DATA0,133,251,133,252,169,1,133,253,133,254,169,9,141,5,212,141,12,212
502 DATA169,0,141,6,212,141,13,212,169,33,133,148,169,17,133,149,169,15,141
503 DATA24,212,120,169,71,141,20,3,169,80,141,21,3,88,96,0,198,253,165,253
504 DATA208,55,169,0,141,4,212,164,251,177,247,201,255,208,7,169,0,133,170
505 DATA76,186,80,141,0,212,200,208,2,230,248,177,247,141,1,212,200,208,2
506 DATA230,248,177,247,133,253,200,208,2,230,248,132,251,165,148,141,4,212
507 DATA234,198,254,165,254,208,44,169,0,141,11,212,164,252,177,249,141,7
508 DATA212,200,208,2,230,250,177,249,141,8,212,200,208,2,230,250,177,249
509 DATA133,254,200,208,2,230,250,132,252,165,149,141,11,212,234,76,49,234
510 DATA0,169,0,141,4,212,141,11,212,169,226,133,247,169,80,133,248,169,65
511 DATA133,249,169,82,133,250,169,0,133,251,133,252,169,1,133,253,133,254
512 DATA76,49,234,0,0,207,34,10,18,39,10,219,43,20,39,52,20,39,52,20,138,58
513 DATA20,39,52,20,219,43,20,207,34,20,207,34,10,18,39,10,219,43,20,39
514 DATA43,20,18,39,20,207,34,20,18,39,60,207,34,10,18,39,10,219,43,20,39
515 DATA52,20,39,52,20,138,58,20,39,52,20,219,43,20,207,34,20,207,34,10,18
516 DATA39,10,219,43,20,219,43,20,18,39,20,18,39,20,207,34,60,207,34,10,18
517 DATA39,10,219,43,20,39,52,20,39,52,20,138,58,20,39,52,20,219,43,20,207
518 DATA34,20,207,34,10,18,39,10,219,43,20,219,43,20,18,39,20,207,34,20,18
519 DATA39,60,207,34,10,18,39,10,219,43,20,39,52,20,39,52,20,138,58,20,39
520 DATA52,20,219,43,20,207,34,20,207,34,10,18,39,10,219,43,20,219,43,20
521 DATA18,39,20,18,39,20,207,34,80,118,46,40,118,46,40,138,58,20,138,58
522 DATA40,138,58,20,39,52,20,39,52,20,219,43,20,207,34,20,18,39,60,207,34
523 DATA10,18,39,10,219,43,20,39,52,20,39,52,20,138,58,20,39,52,20,219,43
524 DATA20,207,34,20,207,34,10,18,39,10,219,43,20,219,43,20,18,39,20,18,39
525 DATA20,207,34,80,118,46,40,118,46,40,138,58,20,138,58,40,138,58,20,39

```

```

526 DATA52,20,39,52,20,219,43,20,207,34,20,18,39,60,207,34,10,18,39,10,219
527 DATA43,20,39,52,20,39,52,20,138,58,20,39,52,20,219,43,20,207,34,20,207
528 DATA34,10,18,39,10,219,43,20,219,43,20,18,39,20,18,39,20,207,34,160,255
529 DATA255,255,0,0,20,103,17,20,20,26,20,237,21,20,20,26,20,103,17,20,20
530 DATA26,20,237,21,20,0,0,20,103,17,20,20,26,20,237,21,20,20,26,20,137
531 DATA19,20,20,26,20,20,26,20,0,0,20,103,17,20,20,26,20,237,21,20,20,26
532 DATA20,103,17,20,20,26,20,237,21,20,0,0,20,20,26,40,20,26,40,103,17,20
533 DATA20,26,10,237,21,10,103,17,20,0,0,20,103,17,20,20,26,20,237,21,20
534 DATA20,26,20,103,17,20,20,26,20,237,21,20,0,0,20,103,17,20,20,26,20,237
535 DATA21,20,20,26,20,137,19,20,20,26,20,20,26,20,0,0,20,103,17,20,20,26
536 DATA20,237,21,20,20,26,20,103,17,20,20,26,20,237,21,20,0,0,20,20,26,40
537 DATA20,26,40,103,17,20,20,26,10,237,21,10,103,17,20,0,0,20,59,23,20,207
538 DATA34,20,69,29,20,207,34,20,59,23,20,207,34,20,69,29,20,207,34,20,237
539 DATA21,40,20,26,40,137,19,20,20,26,20,20,26,20,0,0,20,103,17,20,20,26
540 DATA20,237,21,20,20,26,20,103,17,20,20,26,20,237,21,20,0,0,20,20,26,40
541 DATA20,26,40,103,17,20,20,26,10,237,21,10,103,17,20,0,0,20,59,23,20,207
542 DATA34,20,69,29,20,207,34,20,59,23,20,207,34,20,69,29,20,207,34,20,237
543 DATA21,40,20,26,40,137,19,20,20,26,20,20,26,20,0,0,20,103,17,20,20,26
544 DATA20,237,21,20,20,26,20,103,17,20,20,26,20,237,21,20,0,0,20,20,26,40
545 DATA20,26,40,103,17,20,20,26,10,237,21,10,103,17,120,0,0,0,0,0,0,120
546 DATA169,49,141,20,3,169,234,141,21,3,88,96

```

READY.

HAUPTPROGRAMM

[illegible]

[illegible]


```

590 IFLIX=0THEN700
600 PRINT"DRUECK FIRE TO CONTINUE"
610 POKE56322,224:I=PEEK(56320):IF(IAND16)<>0THEN610
620 POKE56322,255
630 POKE864,PEEK(49207):POKE865,PEEK(49208)
640 POKE866,PEEK(49213):POKE867,PEEK(49214)
650 FORI=0TO62:POKE49153+I,0:NEXT
700 FORI=0TO62:POKE49153+I,0:NEXT
710 PRINT"MIT TAB(15)"GAME OVER":PRINTTAB(15)
720 PRINT"DRUECK FIRE TO START NEW GAME"
730 POKE56322,224:I=PEEK(56320):IF(IAND16)<>0THEN730
740 POKE56322,255:GOTO85:REM 255563214
990 :
991 REM 255563214
992 :
1000 POKE53281,247:POKE53272,18
1005 PRINT"HOERSEGUARD @ BY A.KNIPSCHILD";
1010 PRINT"
1020 PRINT"VERSUCHEN SIE MOEGLICHST VIELE PFERDE"
1030 PRINT"MIT DEM LASSO EINZUFANGEN.
1040 PRINT"ASIE STEuern IHREN REITER MIT DEM JOY-"
1050 PRINT"STICK IN PORT 2 IM LINKEN TEIL DES BILD-"
1060 PRINT"SCHIRMS UND KOENNEN DAS LASSO NACH"
1070 PRINT"RECHTS OBEN, RECHTS UND RECHTS UNTEN"
1080 PRINT"WERFEN (JOYSTICK IN GEWUNSCHETE RICHTUNG
1090 PRINT"UND FEUERKNOPF DRUECKEN).
1100 PRINT"MITASTE DRUECKEN":PRINT"
1110 POKE198,0:WAIT198,1:POKE198,0
1120 PRINT"WEICHEN SIE LANGSAMEREN PFERDEN UND "
1130 PRINT"DEN RINDERKNOCHEN AM BODEN AUS !"
1140 PRINT"ALLE 500 PUNKTE STEIGT DER SCHWIERIGKEITSGRAD, D.H. DAS SPIEL WIRD SCHNELLER.
1150 PRINT"MP.S.: SIE MUESSEN DIE PFERDE GENAU AM HALS TREFFEN. JE WEITER SIE VON DEM GEHTROFFENEN PFERD ENTFERNT SIND, DESTO"
1200 PRINT"MEHR PUNKTE GIBT'S !
1210 PRINT"DRUECKEN SIE EINE TASTE UM ZU BEGINNEN"
1220 PRINT"
1230 POKE198,0:WAIT198,1:POKE198,0:RETURN
19990 :
19991 REM 255563214
19992 :
20000 READA:IFA=-1THENPOKEV+21,0:RETURN
20001 FORJ=0TO7:READB:POKE2048+A*8+J,B:Z=Z+2.2:I=PEEK(2040)+1:IFI=243THENI=235
20002 POKE2040,I:IFZ>390THENZ=0
20003 POKEV,ZAND255:POKEV+16,Z/256:NEXT:GOTO20000
20007 DATA,60,66,153,161,161,153,66,60
20008 DATA1,16,40,68,124,68,68,238,
20009 DATA2,252,66,66,124,66,66,252,
20010 DATA3,62,66,130,128,128,66,60,
20011 DATA4,248,68,66,66,66,68,248,
20012 DATA5,254,66,72,120,72,66,254,
20013 DATA6,254,66,72,120,72,64,224,
20014 DATA7,62,66,128,143,130,66,62,
20015 DATA8,238,68,68,124,68,68,238,

```

```

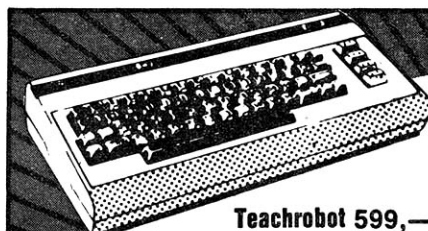
20016 DATA9,56,16,16,16,16,16,56,
20017 DATA10,14,4,4,4,228,68,56,
20018 DATA11,238,72,80,104,68,68,238,
20019 DATA12,224,64,64,64,66,66,254,
20020 DATA13,198,108,84,68,68,68,238,
20021 DATA14,206,68,100,84,76,68,228,
20022 DATA15,56,68,130,130,130,68,56,
20023 DATA16,252,66,66,124,64,64,224,
20024 DATA17,56,68,130,130,146,76,61,2
20025 DATA18,252,66,66,124,80,72,230,
20026 DATA19,126,130,128,124,2,130,252,
20027 DATA20,254,146,16,16,16,16,124,
20028 DATA21,238,68,68,68,68,68,56,
20029 DATA22,238,68,68,68,68,40,16,
20030 DATA23,238,68,68,68,84,108,68,
20031 DATA24,238,68,40,16,40,68,238,
20032 DATA25,238,68,40,16,16,16,56,
20033 DATA26,254,132,8,16,32,66,254,
20034 DATA48,56,68,130,130,130,68,56,
20035 DATA49,8,24,40,72,8,8,8,
20036 DATA50,124,130,2,60,64,128,254,
20037 DATA51,254,4,8,28,2,130,124,
20038 DATA52,20,20,36,68,254,4,4,
20039 DATA53,254,128,188,194,2,130,124,
20040 DATA54,124,130,184,196,130,130,124,
20041 DATA55,254,2,2,12,16,16,16,
20042 DATA56,56,68,68,124,130,130,124,
20043 DATA57,124,130,130,126,2,132,120,
20044 DATA36,40,126,168,124,42,252,40,
20045 DATA73,2,1,3,14,31,63,56,
20046 DATA74,8,8,240,228,230,200,48,16
20047 DATA79,17,17,17,255,17,17,17,17
20048 DATA80,34,34,34,255,34,34,34,34
20049 DATA114,68,68,68,255,68,68,68,68
20050 DATA115,136,136,136,255,136,136,136,136

```

```

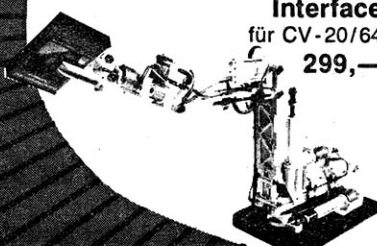
20051 DATA-1
20990 :
20991 REM 255563214
20992 :
21000 DATA1343,1383,1423,1463,1503,1543,1583,1623,1663,1703,1743,1783,1823,1863
21005 DATA1304,1344,1384,1424,1464,1504,1544,1584,1624,1664,1704,1744,1784,1824

```



Teachrobot 599,—
6 Achsen mit Wegnehmer
Nutzlast 200g (Bausatz)

Fertiggerät 699,—
Interface
für CV-20/64
299,—



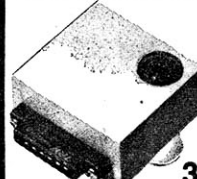
Info 3/84: 1,- Porto in Briefm.
Alle Preise inklusiv Mehrwertsteuer. 6 Monate
Garantie. Versand erf. per NN oder Vorkasse.
Händleranfragen erwünscht

Für Commodore VC-20/64

Speichervollausbau für VC-20

32/27 KByte-Modul

Ersetzt 3 + 8 + 16 KByte oder 8 + 8 + 16 KB
kompakt in einem Modul! Voll schaltbar!



39,—
Recorderinterface
Schließt Ihren Recorder an
VC-20 oder C-64.
Inklusive Motorsteuerung!



88,—
Datenrecorder
mit Anschlußkabel steckfertig
Sensationspreis!

80-Zeichenkarte
für C 64 279,—
40/80-Zeichenkarte (20) 219,—
Eprommer VII (20/64) 179,—
programmiert 2508, 2516, 2716, 2532,
2732. Betriebsber. incl. Steuersoftware!
Eprommer VIII (20/64) 249,—
wie oben, auch für 2764, 27128 geeignet.
Centronics Intf. (26/64) 169,—
schließt centr. komp. Drucker an VC's
Grandmaster (20/64) 79,—
Superstarkes Schach!
10 er Tastaturen:
Montage ohne löten
T1: 119,—
T2: 179,—

ROM-Modul 39,— für
2 EPROM'S-2716,
2732, 2764
mit Gehäuse

Klaus Jeschke
Hard-, Software
Im Birkenfeld 3 n
6233 Kelkheim
☎ (06198) 7523

Mensch ärgere Dich doch für den Commodore 64

Programmablauf:

Wenn man das Programm mit RUN startet, dann wird zuerst der SID initialisiert (resetiert), indem alle Register auf 0 gesetzt werden. Nach der Dimensionierung der Felder und der Zuweisung von konstanten Variablen springt das Programm in das UP zur Begrüßung. Man kann diese übergehen, falls sie zu lange erscheint, indem man in Zeile 3 "**POKE 2024,1**" schreibt.

Nach dem Drücken einer Taste erscheint ein Mini-Menü auf dem Bildschirm, welches abfragt, ob man das Spiel normal starten möchte oder den Spielstand von Band laden will. Arbeitet man mit einem Diskettenlaufwerk, so ist als einzige Änderung in Zeile 21 die Zahl 1 durch 8 oder 9 zu ersetzen.

Nun drückt man entweder 1 oder 2. Hat man den normalen Spielbeginn gewählt, erfolgen nun weitere Abfragen:

An dem Spiel können 1-4 Personen teilnehmen. Gibt man aber als Spieleranzahl eine 0 ein, so folgt ein Demonstrationsspiel, in dem 4 Computer gegeneinander spielen.

Tut man das nicht, so wird gefragt, mit wievielen Spielfiguren pro Spieler gespielt werden soll. Hier sind Eingaben von 2-4 möglich. Danach erfolgt die Eingabe der Spielernamen durch INPUT-Statements. Dabei kann man den Computer beliebig oft mitspielen lassen. Man muß jedoch darauf achten, daß die Anzahl der eingegebenen Namen außer dem Computer mit der vorhin eingegebenen Zahl übereinstimmt. Sonst zeigt der Computer unten an, wieviele Spielernamen fehlen. Spielt man alleine so ist es wohl selbstverständlich, daß der Computer mitspielen muß, ansonsten Fehlermeldung.

Wenn man dies überstanden hat, so wird das vertraute Spielfeld aufgebaut. Das Spiel wird durch den zauberhaften Klang der Big-Ben Glocken eröffnet.

Hat man bei dem Mini-Menü Punkt Nr. 2 gewählt, so wird bei Verwendung einer Datasette noch abgefragt, ob das Band an die richtige Stelle gespult werden soll. Wenn man hier etwas eingibt, nützt es eigentlich nichts. Die Abfrage dient ledig-

lich dazu, daß nach dem Eingeben der "2" nicht gleich die Datasette losläuft und ein Chaos hervorruft. Nach dem Drücken von RETURN erscheint die Aufforderung PRESS, falls dies nicht schon getan wurde. Darauf werden sämtliche Daten, die das Programm braucht, geladen: Spielernamen, Positionen der Figuren, wer gerade an der Reihe ist, ob einer schon im Ziel ist.

Bei der Verwendung einer Floppy läuft diese sofort nach Drücken der "2" an, falls sie angeschlossen ist.

Nach der Big-Ben Melodie beim normalen Spielbeginn wird zufällig ausgewählt, wer beginnt. Danach ist einer der runden Kästen weiß.

Wenn man den Spielstand geladen hat, fahren zuerst alle Figuren an ihre Plätze, wonach der Kasten weiß ist, dessen Spieler an der Reihe war.

Dieser "Kasten" wird nun als Würfel benutzt.

Wenn man jetzt auf irgendeine Taste drückt, würfelt man. Der Computer gibt dabei in schneller Folge Zufallszahlen hintereinander aus.

Läßt man die Taste los, so bleibt die Zahl im Würfel stehen, die man gewürfelt hat. Nun drückt man eine Zahl von 1 bis 4 und wählt somit die Figur, mit der man ziehen will. Die Figuren sind durch farbige inverse Zahlzeichen dargestellt. Ein Spiel auf dem S/W-Gerät ist daher sinnlos. Der Computer läßt bei der Eingabe nur die Zahlen zu, bei denen ein Ziehen mit der Figur möglich ist. Andere Eingaben werden ignoriert.

Sonderfälle:

- Der würfelnde Spieler hat alle Figuren im "Haus" und hat keine "6" gewürfelt: ER darf 3 mal würfeln.
- Ein Spieler kann mit keiner Figur ziehen und hat alle Würfe verbraucht: Der nächste kommt dran.
- Ein Spieler ist "fertig": An Stelle seines Würfels wird eine Digitaluhr als Platzierung angezeigt. Er wird im folgenden beim Würfeln übergangen.
- Alle Spieler außer einem sind fertig: Das Spiel wird abgebrochen und beim noch "aktiven" Spieler die schlechteste Pla-

zierung angegeben.

Tritt keiner dieser Sonderfälle auf, so zieht die Figur. Das Werfen von Figuren anderer Spieler wird optisch und akustisch angezeigt. Kommt der Computer an die Reihe, so würfelt er automatisch. Darauf wählt er automatisch eine Figur. Er hat dabei eine gewisse Strategie, ist aber auch von der zufällig gewürfelten Zahl abhängig.

Zusätzliche Einrichtungen:

1.) Drückt man, anstatt mit einer beliebigen Taste zu würfeln, die Funktionstaste 1, so kann man den Spielstand abspeichern. Mit einer Datasette erhält man die Meldung PRESS RECORD & PLAY, beim Benutzen eines Disklaufwerkes "TASTE DRUECKEN". Nach dem Abspeichern kann man das Spiel fortsetzen.

2. Drückt man an der gleichen Stelle "f3", so wird das Spiel unterbrochen. Der Computer rechnet nun aus, wie weit jeder schon gekommen ist, auch wenn er noch nicht im Ziel ist, und gibt danach die Platzierungen auf einen Blick aus. Währenddessen erscheint im Takt einer Melodie blinkend die große Frage: Noch ein Spiel?, die man mit "j" bzw. "n" beantwortet. Falls Sie "n" eingeben, schalten Sie den Computer nicht gleich aus, sondern "listen" Sie das Programm doch einmal. Es ist zum Glück noch da.

Ansonsten gelten die normalen Spielregeln von Mensch, ärgere Dich nicht.

Die wichtigsten Variablen:

SI: Basisadresse des SID

GN: Geräteummer

M(X): Einzelne Melodietöne (Begrüßungsmelodie)

T(X): Tonreihe der benutzten Töne

B(X,Y): Baßöne

SF(X,Y): Positionen der Spielfiguren (0-40)

PF(X,Y): Bildschirmadressen der Spielfiguren

MK(X,Y): Merkvariablen zum Zwischenspeichern des Bildschirmcodes, der vorher anstelle der Spielfigur war.

MF(X,Y): wie MK(X,Y), jedoch Farbcode

programme

```

1201 REM ZIELFELDER
1202 REM -----
1210 FORI=1203TO1443STEP80
1220 POKEI,87:POKESI+I,10:NEXT
1230 FORI=1603TO1843STEP80
1240 POKEI,87:POKESI+I,5:NEXT
1250 FORI=1515TO1521STEP2
1260 POKEI,87:POKESI+I,7:NEXT
1270 FORI=1525TO1531STEP2
1280 POKEI,87:POKESI+I,14:NEXT
1290 W$=" ",XXXXXXXXXX,XXXXXXXXXX,XXXXXXXXXX,XXXXXXXXXX"
1340 REM -----
1341 REM NAMEN
1342 REM -----
1350 POKESI+6,240:POKESI+4,17
1351 FORI=1TO4:IFF$(I)=""THENF$(I)=CHR$(32)
1352 NEXT
1360 FORI=1TOLEN(F$(1))
1370 POKE1023+I,ASC(MID$(F$(1),I,1))AND63
1380 POKE55295+I,7:GOSUB1480:NEXT
1390 SA=1063-LEN(F$(2)):FORI=1TOLEN(F$(2))
1400 POKESA+I,ASC(MID$(F$(2),I,1))AND63
1410 POKESI+SA+I,10:GOSUB1480:NEXT
1420 FORI=1TOLEN(F$(4))
1430 POKE1983+I,ASC(MID$(F$(4),I,1))AND63
1440 POKE56255+I,5:GOSUB1480:NEXT
1450 SA=2023-LEN(F$(3)):FORI=1TOLEN(F$(3))
1460 POKESA+I,ASC(MID$(F$(3),I,1))AND63
1470 POKESI+SA+I,14:GOSUB1480:NEXT
1475 POKESI+5,11:POKESI+6,0:GOTO1530
1480 FORX=40TO80-I*4
1490 POKESI+1,X:NEXTX:RETURN
1500 REM -----
1510 REM BIG-BEN GLOCKE
1520 REM -----
1530 FORW=1TO2000:NEXT
1540 POKESI+4,21
1550 DATA10,8,9,6,6,9,10,8,9,10,9,6,8,9,10,8
1555 DIMFR(15):RESTORE:FORI=1TO4:READUE:NEXT:FORI=0TO15:READFR(I):NEXT
1560 SL=3:FOR$=0TO23:POKESI+S,0:NEXT:POKESI+24,12
1570 FORI=0TO14STEP7:POKESI+I+5,76:POKESI+I+3,8:NEXT
1580 J=-1:FORI=0TO15:J=J+1:IFJ=3THENJ=0
1590 POKESI+4+J*7,0:POKESI+4+J*7,65:POKESI+1+J*7,FR(I)
1600 FORT=TTOT+1STEP.09:POKESI+J*7,ABS(SIN(T*2)*30):NEXTT
1605 II=(I+1)/4:IFII=INT(II)ANDR1=0THENR1=1:GOTO1600
1610 R1=0:NEXTI:S=SL:POKESI+11,33:POKESI+18,33:RETURN
2000 REM -----
2001 REM WER BEGINNT ?
2002 REM -----
2010 POKESI+4,0:POKESI+5,0:POKESI+6,240:POKESI+4,33
2020 X=40+INT(RND(1)*4)+1
2030 F$(0)="" :S=0:FORI=1TOX:GOSUB2125:NEXT
2040 POKESI+5,10:POKESI+6,0:FORW=1TO2000:NEXT:RETURN
2100 REM -----
2110 REM SPIELERWECHSEL
2120 REM -----
2125 ME=3
2130 S=S+1:IFS=5THENS=1
2135 IFF$(S)=""ORASC(F$(S))>100THEN2130
2136 IFF$(ME)<>" "ANDASC(F$(ME))<101THEN2140
2137 ME=ME-1:IFME<1THENME=4
2138 GOTO2136
2140 ONMEGOSUB2150,2160,2170,2180
2145 ONSGOTO2190,2200,2210,2220
2150 PRINT"SPIELER";W$:RETURN
2160 PRINT"SPIELER"TAB(34)" ";W$:RETURN
2170 POKE214,18:PRINT:PRINT"SPIELER"TAB(34);W$:RETURN
2180 POKE214,18:PRINT:PRINT"SPIELER"TAB(34);W$:RETURN
2190 PRINT"SPIELER";W$:POKESI+1,12:RETURN
2200 PRINT"SPIELER"TAB(34)" ";W$:POKESI+1,16:RETURN
2210 POKE214,18:PRINT:PRINT"SPIELER"TAB(34);W$:POKESI+1,20:RETURN
2220 POKE214,18:PRINT:PRINT"SPIELER"TAB(34);W$:POKESI+1,24:RETURN
2300 REM -----
2310 REM DREIMAL WUERFELN ?
2320 REM -----
2330 WA=1:FORI=1TO2
2340 IFSF(S,I)=0ORSF(S,I)>40THENNEXTI:WA=3
2350 RETURN
2400 REM -----
2410 REM WUERFELN
2420 REM -----
2430 ON S GOSUB2450,2460,2470,2480
2440 IFMP=1THENPRINT"SPIELER";GOTO3480
2445 GOTO2490
2450 PRINT"SPIELER";:RETURN
2460 PRINT"SPIELER"TAB(35);:RETURN

```

```

2470 POKE214,19:PRINT:PRINTTAB(35):RETURN
2480 POKE214,19:PRINT:PRINT"III":RETURN
2490 IFF$(S)<>"COMPUTER"THENPOKE198,0:WAIT198,1:GOTO2500
2495 FORN=1TO500:NEXT
2500 ZC=PEEK(203):IFZC=40RZC=5THEN5030
2502 POKE51+4,0:POKE51+5,0:POKE51+6,240:POKE51+4,17:PRINT"III";
2505 IFF$(S)="COMPUTER"THENFORCW=1TO10
2510 Q=INT(RND(1)*6)+1
2520 PRINTW$(Q);W$(Q);
2530 POKE51+1,64+Q*Q
2535 IFF$(S)="COMPUTER"THENNEXTCW:GOTO2550
2540 IFPEEK(203)<>64THENGOTO2510
2550 POKE51+5,10:POKE51+6,0
2555 MO=0:FI=0:UN=0:FJ=0:IFQ=6THENGOSUB8030:IFMO<>0THEN2680
2560 REM -----
2570 REM KANN EINE FIGUR ZIEHEN ?
2580 REM -----
2590 FORI=1TO2
2600 IFSF(S,I)=0THEN2647
2610 REM -----
2611 REM PROBEDURCHLAUF EINES ZUGES
2612 REM -----
2620 P=SF(S,I):FORPR=1TOQ
2630 P=P+1:IFP=41THENP=1
2635 IFF=-9+10*STHENP=P+99
2640 IFF$(S)="COMPUTER"THENGOSUB4100
2641 NEXTPR:IFP>93+10*STHEN2647
2643 FORJ=1TO2:IFSF(S,J)=PTHEN2647
2644 NEXTJ
2645 MO=MOOR2*I:GOSUB4830
2647 NEXTI:IFMO=0THENF=0:RETURN
2650 REM -----
2660 REM EINGABE:WELCHE FIGUR ZIEHT ?
2670 REM -----
2680 POKE51+4,0:POKE51+1,0:POKE51+5,0:POKE51+6,60:POKE51+4,17
2685 IFF$(S)="COMPUTER"THEN4060
2690 GETA#F=VAL(A#):IFF=0ORF>2THENGOSUB2710:GOTO2690
2700 IF(MOAND(2*F))=0THEN2685
2705 RETURN
2710 X=X+1:IFX=10THENPOKE51+1,64:RETURN
2720 IFX=20THENPOKE51+1,0:RETURN
2730 IFX=30THENPOKE51+1,32:RETURN
2740 IFX=40THENX=0:POKE51+1,0:RETURN
2750 RETURN
2800 REM -----
2810 REM FIGUR ZIEHT
2820 REM -----
2825 POKE51+6,0:POKE51+5,7:POKE51+1,48
2830 RI=0:IFSF(S,F)=0THEN3000
2831 REM -----
2832 REM RICHTUNG
2833 REM -----
2835 FORZE=1TOQ
2837 P=SF(S,F)
2840 IF(0<PANDP<5)OR(P=90ORP=10)OR(14<PANDP<19)OR(99<PANDP<104)THENRI=2
2850 IF(30<PANDP<35)OR(P=39ORP=40)OR(4<PANDP<9)OR(129<PANDP<134)THENRI=-80
2860 IF(20<PANDP<25)OR(P=29ORP=30)OR(34<PANDP<39)OR(119<PANDP<124)THENRI=-2
2870 IF(10<PANDP<15)OR(P=19ORP=20)OR(24<PANDP<29)OR(109<PANDP<114)THENRI=80
2880 IFKM=1THENKM=0:GOTO2950
2920 POKEPF(S,F),MK(S,F):POKE51+PF(S,F),MF(S,F)
2930 SF(S,F)=SF(S,F)+1:IFSF(S,F)=41THENSF(S,F)=1
2940 IFSF(S,F)=-9+10*STHENSF(S,F)=SF(S,F)+99:KM=1:GOTO2837
2950 PF(S,F)=PF(S,F)+RI
2955 MK(S,F)=PEEK(PF(S,F)):MF(S,F)=PEEK(SI+PF(S,F))
2960 POKEPF(S,F),176+F:POKE51+PF(S,F),FA(S)
2970 POKE51+4,0:POKE51+4,32
2975 IFLD=1THENGOSUB14500:IFII=0THENRETURN
2980 NEXTZE:RETURN
2990 REM -----
2991 REM FIGUR ZIEHT VON STARTFELD
2992 REM -----
3000 POKE5P(S,F),87:SF(S,F)=-9+10*S
3010 IFS=1THENXU=1433
3020 IFS=2THENXU=1125
3030 IFS=3THENXU=1613
3040 IFS=4THENXU=1921
3050 PF(S,F)=XU:MK(S,F)=PEEK(XU):MF(S,F)=PEEK(SI+XU)
3051 POKE51,176+F:POKE51+XU,FA(S)
3055 IFLD=1THENRETURN
3060 POKE51+4,0:POKE51+5,13:POKE51+4,21
3070 POKE51+11,16
3080 FORT=1TO40:FORTH=0TO2
3090 POKE51+1,32+16*TN:POKE51+15,10+5*T/(TN+1):FORW=1TOABS(30-T):NEXT
3100 NEXTTN,T:POKE51+1,0:RETURN
3200 REM -----
3210 REM FIGUR WERFEN

```

programme

```

0320 REM -----
0320 FORQA=1T04:FORQB=1T02
0320 IFQA=STHEN3270
0320 IFSF(S,F)=SF(QA,QB)THEN3300
0320 NEXTQB
0320 NEXTQA:RETURN
0300 POKESI+11,0:POKESI+13,120:POKESI+11,21:POKESI+6,240:POKESI+1,9
0310 FL=0:FORT=150T00STEP-5
0320 FORTH=1T05:POKESI+8,T+TN*2:POKESI+8,200-T-TN*2
0330 FL=FL+1:POKESI+PF(S,F),FLAND15
0340 NEXTTH,T
0345 MK(S,F)=MK(QA,QB):MF(S,F)=MF(QA,QB)
0350 POKESI+PF(S,F),FA(S)
0360 SF(QA,QB)=0:POKESF(QA,QB),176+QB
0370 POKESI+5,10:POKESI+6,0:POKESI+8,0:POKESI+4,33:POKESI+1,0:POKESI+11,0
0380 RETURN
0400 REM -----
0410 REM GEMONNEN ?
0420 REM -----
0430 FORI=1T02:IFSF(S,I)<90+10*STHENRETURN
0440 NEXTI
0450 REM -----
0460 REM GEMONNEN
0470 REM -----
0480 A=WA:PRINTW$(0);
0490 Z$(10)="          | |          | - = |          | |          | II"
0491 Z$(9)="          | |          |>|>"
0492 Z$(8)="          | |          |>|>"
0493 Z$(7)="          | |          |>|>"
0494 Z$(6)="          | |          |>|>"
0495 Z$(5)="          | |          |>|>"
0496 Z$(4)="          | |          |>|>"
0497 Z$(3)="          | |          |>|>"
0498 Z$(2)="          | |          |>|>"
0499 Z$(1)="          | |          |>|>"
0500 E$="TTTTTT":FORX=1T010:Z$(X)=Z$(X)+E$:NEXT
0505 POKESI+4,0:POKESI+5,0:POKESI+6,240:POKESI+4,33
0510 GW=GW+1:FORX=10T0GUSTEP-1
0520 PRINTZ$(X);
0530 FORJ=X*15T010STEP-5
0540 POKESI+1,J:POKESI+1,0:POKESI+1,J/3:POKESI+1,0:POKESI+1,J/5:POKESI+1,0
0550 NEXTJ,X
0570 GW$(GW)=CHR$(FA(S))+CHR$(48+GW)+". "+F$(S):F$(S)=CHR$(100+GW)+F$(S)
0580 IFGW=ANTHEN3627
0590 IFGW=AN-1THENMP=1
0595 IFLD=1THENRETURN
0600 GOSUB1560:RETURN
0610 REM -----
0620 REM ENDE
0625 REM -----
0627 POKE214,11:PRINT PRINTTAB(10)" TASTE DRUECKEN !" WAIT200,60
0628 GOSUB6000
0670 W1=31:W2=64:A=26:M$="HOCH EIN SPIEL ?"
0680 FORS=0T024:POKESI+8,0:NEXT:GOSUB9920:GOSUB10040
0645 PRINT"T"TAB(27);
0650 GETA$:IFA$<>"J"ANDA$<>"N"THEN3650
0660 POKESI+4,0:POKESI+5,12:POKESI+6,0:POKESI+4,33
0670 IFA$="N"THENPRINT"NEIN"CHR$(9)" "
0680 IFA$="J"THENPRINT"JA"
0690 FORT=1T020:GOSUB20000:NEXT
0700 IFA$="J"THENPOKE2024,1:RUN20
0710 POKESI+24,0:SYS49152:POKE198,0:END
0800 REM -----
0810 REM COMPUTER WAHLT FIGUR
0820 REM -----
0830 FORMS=1T04:FORMF=1T02:IFWS=STHENNEXTWS:RETURN
0840 IFSF(W,S)=PTHEFI=FIOR21:I:GOTO4055
0850 NEXTWF,WS
0855 IFP>40ANDSF(S,I)<41THENFJ=FJ+2*I
0856 RETURN
0860 F=INT(RND(1)*Z)+1
0865 IFFJ<0THENFI=0
0870 IF(FIAND(2*F))<0THEN2700
0875 IF(FJAND(2*F))<0THEN2700
0877 IFFI=0ANDFJ=0THENIFUN=0OR(UNANDMO)=MOOR(UNAND(2*F))=0THEN2700
0880 GOTO4060
0900 FORK=1T04:IFK=STHEN4130
0910 FORL=1T02:IFSF(K,L)=PANDFR<>QTHENUN=UN+2*I
0920 NEXTL
0930 NEXTK:RETURN
5000 REM -----
5010 REM ABSPEICHREN DES SPIELSTANDES
5020 REM -----
5030 POKE53280,4:XP=POS(0):YP=PEEK(214)-1:SYS49233
5031 IFZC=STHEN5500
5035 FORI=0T02:FORJ=1433T01453

```



```

5040 POKE30000+J+I*80,PEEK(J+I*80):POKE30200+J+I*80,PEEK(SI+J+I*80)
5050 NEXT J,I
5060 POKE214,9:PRINT:PRINTTAB(9)"      ■ABSPEICHERN      "
5070 PRINTTAB(12)"DES SPIELSTANDES"
5080 PRINTTAB(9)"
5085 IFGN<1THENPRINTTAB(10)"3 TASTE DRUECKEN ":A1%=203:A2%=63:A3%=0
5090 IFGN=1THENPRINTTAB(10)"PRESS RECORD & PLAY":A1%=1:A2%=39:A3%=39
5100 PRINTTAB(9)"
5110 WAITA1%,A2%,A3%
5120 OPEN1,GN,1:"SPIELSTAND":PRINT#1,Z
5130 FORI=1T04:IFI=STHENF(I)="*"+F(I)
5140 PRINT#1,F(I):IFI=STHENF(I)=MID(F(I),2,20)
5150 FORJ=1T02:PRINT#1,SF(I,J):NEXTJ,I:CLOSE1
5160 POKE214,9:PRINT
5170 FORI=1T05:PRINTTAB(10)"
5180 NEXT
5220 POKE53280,14:FORI=0T02:FORJ=1433T01453
5230 POKEJ+I*80,PEEK(30000+J+I*80):POKESI+J+I*80,PEEK(30200+J+I*80)
5240 NEXTJ,I
5250 POKE214,10:PRINT:PRINTTAB(10)"
5260 POKE214,YP:PRINT:PRINTTAB(XP)"#":GOTO2490
5490 REM -----
5495 REM PLAZIERUNG BEI SPIELABBRUCH
5497 REM -----
5500 FORI=1T04:IFF(I)=" "ORASC(F(I))>100THENGOSUB5550
5510 FORJ=1T02:IFSF(I,J)>40THENSF(I,J)=41:GOTO5550
5520 IFSF(I,J)=0THEN5550
5530 SF(I,J)=SF(I,J)-10*(I-1)
5540 IFSF(I,J)<=0THENSF(I,J)=SF(I,J)+40
5550 NEXTJ
5560 FORJ=1T02:SF(I,0)=SF(I,0)+SF(I,J):NEXT
5570 NEXTI:GOTO5600
5580 IFF(I)<>" "THENSF(I,0)=600+100*(ASC(F(I))-100)
5585 SF(I,0)=-1
5590 RETURN
5600 FORI=1T04:IFASC(F(I))>100THENF(I)=MID(F(I),2,19)
5610 GW(I)=CHR$(F(I))+F(I)
5620 NEXT
5630 FORI=1T03
5640 IFSF(I,0)=SF(I+1,0)THENNEXT GOTO5670
5650 SF=SF(I,0):SF(I,0)=SF(I+1,0):SF(I+1,0)=SF
5660 GW=GW(I):GW(I)=GW(I+1):GW(I+1)=GW:GOTO5630
5670 I=1:FORK=1T04
5680 GW(K)=LEFT$(GW(K),1)+CHR$(48+I)+". "+MID$(GW(K),2,19)
5690 IFK<4THENIFSF(K,0)~SF(K+1,0)THENI=I+1
5700 NEXT
5710 GOTO3620
5800 POKE53280,7:POKE53281,9:PRINT"1000"
5805 PRINT"1000PLAZIERUNGEN 10"
5810 FORI=1T04:POKE646,ASC(GW(I))
5820 PRINT"10"TAB(10),MID$(GW(I),2,19)
5830 NEXT:PRINT"100000":RETURN
5800 REM -----
5810 REM ES WURDE 6 GEMUERFELT
5820 REM -----
5830 FORI=1T02:IFSF(S,I)=-9+10*STHENRETURN
5840 NEXTI
5850 FORI=1T02:IFSF(S,I)=0THENMO=MOOR2+I
5860 NEXTI:RETURN
5900 REM -----
5910 REM BEGRUESSUNG
5920 REM -----
5925 GOSUB9920:POKESI+24,15:POKESI+23,160
5930 FORI=0T02:POKESI+3+I*7,0:POKESI+6+I*7,240:POKESI+4+I*7,65
5940 POKESI+1+I*7,3:POKESI+I*7,I*4:NEXTI
5950 POKE53280,7:POKE53281,9:PRINT"1000"TAB(4);
5955 SYS49233
"
5970 A$=A$+"1000TTTTT1000SUDDER GAMES & CPU PRESENT:"
5980 FORI=1T092:PRINTMID$(A$,I,1):POKESI+1,ASC(MID$(A$,I,1))
5985 IFI<60THENPOKE646,FA(RND(1)*4+1)
5990 FORN=1T040:NEXTN,I
5991 FORI=0T02:POKESI+5+I*7,14:POKESI+6+I*7,0:NEXT
5995 FORK=2T028:POKE646,K
9100 PRINT:FORI=1T027:PRINTTAB(11)">":POKESI+1,I:NEXT:PRINT"1000";
9110 FORI=27T01STEP-1:PRINT"1000":POKESI+1,I:NEXT:PRINT"1000";
9120 NEXTK
9130 PRINT:PRINT"1000"TAB(10);
9140 FORS=0T023:POKESI+S,0:NEXT
9150 POKE214,13:PRINT:PRINTTAB(9)"#-----"
9155 PRINTTAB(9)"1000 BY MARTIN SAUR-BROSCH"
9160 PRINTTAB(9)"1000 KOLPINGSTR.19"
9170 PRINTTAB(9)"1000 6969 HOEPFINGEN"
9180 PRINTTAB(9)"1000"
9190 GOTO10020

```

programme

```

9200 SYS49233:A$="T0000000000001.NORMALER SPIELBEGINN"
9210 A$=A$+CHR$(13)+"00002.SPIELSTAND LADEN VON "
9215 G$="BAND":IFON<>1THENG$="DISK"
9216 A$=A$+G$
9220 A$=A$+"XXXXXXXXXXXXXXXXXXXXBITTE WAEHLEN SIE !"
9230 GOSUB9200:POKE198,0
9240 GETA$:IFA$="1"THENRETURN
9250 IFA$="2"THENLD=1:RETURN
9260 GOTO9240
9300 FORA=1TOLEN(A$):PRINTMID$(A$,A,1):FORM=1TO50:NEXTW,A:RETURN
9301 REM -----
9302 REM   DATEN FUER MASCHINENPROGRAMME
9303 REM -----
9910 DATA169,39,141,22,288,162,250,142,232,7,164,0,232,208,253,200,208,250
9911 DATA238,232,7,173,232,7,208,242,169,200,141,22,208,169,14,141,32,208
9912 DATA169,6,141,33,208,169,14,141,134,2,165,43,164,44,32,8,164,169,115
9913 DATA160,228,32,30,171,165,55,56,229,43,170,165,56,229,44,32,205,189,169
9914 DATA96,160,228,32,30,171,96,238,32,208,232,208,250,200,208,247,238,32
9915 DATA208,238,33,208,232,208,247,200,208,244,238,33,208,232,208,250,200
9916 DATA208,247,96
9920 RESTORE:FORI=1TO20:READUE:NEXT
9930 FORI=49152TO49263:READC:POKEI,C:NEXT:RETURN
10000 REM -----
10010 REM   BEGRUESSUNGSMELODIE
10015 REM -----
10020 POKE214,7:PRINT:PRINT"  TAB(10);
10040 DATA4455,5005,5600,5990,6680,7493,8911
10050 DATA1114,1403,1669,2228
10060 DATA1051,1403,1669,2103
10070 DATA 992,1403,1669,1985
10080 DATA 937,1114,1487,1873
10090 DATA1114,1487,1768,2228
10100 DATA 834,1114,1403,1669
10110 DATA1250,1487,1873,2228
10120 DATA 834,1051,1250,1487
10130 DATA1114,1403,1669,2228
10150 FORI=1TO7:READT(I):NEXT
10160 FORI=1TO9:FORJ=1TO4:READB(I,J):NEXTJ,I
10170 DATA3,5,0,5,0,0,0,0,3,5,0,5,0,0,0,0,3,0,5,7,0,0,6,5,0,4,0,4,0,0,0,0
10180 DATA4,4,0,4,0,0,0,0,5,4,3,0,3,0,0,0,0,3,3,0,2,0,2,3,0,2,0,0,0,0,0,0,0
10190 DATA3,0,2,0,4,3,0,1,0,0,0,0,0,0,0,0,0,0
10200 FORI=1TO80:READM(I):NEXT:POKESI+24,31:POKESI+5,11:POKESI+12,11.
10210 POKESI+23,4:POKESI+22,0
10220 POKESI+17,11:POKESI+20,240:POKESI+18,65:GE=W1:TE=W2:ER=0
10230 J=0:K=0:B=1
10240 FORI=1TO64:IFM(I+ER)=0THENFORM=1TO80:NEXT:GOTO10280
10250 POKESI+4,32:POKESI+11,20
10260 M=M(I+ER):POKESI,T(M)AND255:POKESI+1,T(M)/256
10270 POKESI+7,-20+T(M)AND254:POKESI+8,T(M)/TE
10275 A=A+1:IFA<27THENPRINTMID$(M$,A,1);
10276   IFA=27THENPRINT
10277   IFA<27THENPOKE646,A:PRINTTAB(10)"  M$
10280 K=K+1:IFK=5THENK=1:J=J+1
10290 IFJ=2THENB=B+1:J=0
10300 POKESI+4,33:POKESI+11,GE
10310 BA=B(B,K)*2:POKESI+14,BAND255:POKESI+15,BA/256
10320 IFB=6ANDK=4ANDJ=1ANDGE=21THENER=16:B=B+1
10330 IFB=9ANDJ=1THEN10360
10340 FORM=1TO50:NEXTW,I
10350 TE=64:GE=21:GOTO10230
10360 POKESI+19,12:POKESI+20,0:IFM$="NOCH EIN SPIEL ?"THENRETURN
10365 FORM=1TO1500:NEXT
10370 POKESI+4,0:POKESI+5,0:POKESI+6,240:POKESI+4,17
10380 FORI=0TO24:FORJ=0TO2:POKEI+J*40+55906,1:NEXTJ
10390 FORX=40TO64-I:POKESI+1,X:NEXTX
10400 NEXTI:POKESI+5,14:POKESI+6,0
10410 A=0:POKE198,0
10415 FORI=0TO13:POKE55879+I,A:POKE55878-I,A:NEXT
10420 FORI=0TO4:POKE55905+I*40,A:POKE55931+I*40,A:NEXT
10425 FORI=13TO8STEP-1:POKE56039+I,A:POKE56038-I,A:NEXT
10430 A=A+1:IFA=256THENA=0
10440 IFPEEK(198)=0THEN10415
10460 FORI=0TO23:POKESI+I,0:NEXT
10470 GOSUB9200
10480 REM -----
10490 REM   WIEVIELE SPIELER/FIGUREN ?
10500 REM -----
10510 IFLD=1THEN14000
10515 SYS49233:PRINT"  "
10520 FORI=0TO40:POKE53281,1:POKE53281,2:NEXT:POKESI+6,240:POKESI+4,33
10530 PRINT"  WIEVIELE VON EUCH WOLLEN SICH AN"
10540 PRINT"  DEM SPIEL BETEILIGEN (0-4) ?";
10550 GETA$:AN=VAL(A$):IFAN>4THEN10550
10551 IFA$="0"THENPRINT"  KEINER.":GOTO11000
10552 IFAN=0THEN10550

```

```

10560 AN$=STR$(AN)+" LEUTE":IFAN=1THENAN$=" EINER"
10570 PRINTAN$:GOSUB20000
10610 PRINT"MIT WIEVIELEN SPIELFIGUREN PRO SPIELER"
10620 PRINT"MSOLL GESPIELT WERDEN (2-4)?";
10630 GETZ$:Z=VAL(Z$):IFZ<2ORZ>4THEN10630
10640 PRINTZ"FIGUREN":GOSUB20000:FORW=1TO1000:NEXT:SYS49233
10650 POKE53281,9:PRINT"J"
10660 PRINT"JEDER SPIELER GIBT SEINEN NAMEN UNTER"
10670 PRINT"DER FARBE AN,DIE ER WAHLT."
10671 PRINT"JEDE FARBE,HINTER DER EIN C EINGEGEBEN"
10672 PRINT"WIRD,STEUERT DER COMPUTER."
10680 PRINT"WAHLT KEINER DIESE FARBE,NUR RETURN"
10690 PRINT"DRUECKEN."
10700 PRINT"NGELB":PRINT"KROT":PRINT"KBLAU":PRINT"NGRUEN"
10710 FORI=1TO4:POKE214,8+I*2:PRINT
10720 POKE646,FA(I)
10730 INPUT"DDDD";F$(I)
10740 NEXTI
10750 A=0:FORI=1TO4:IFF$(I)<>"ANDF$(I)<>"C"THENA=A+1
10760 NEXTI
10770 IFA=ANTHEN10820
10780 DI=AN-A:IFDI=1THENFE$="ES FEHLT NOCH EINER."
10790 IFDI=-1THENFE$="ES IST EINER ZUVIEL."
10795 IFDI>1THEN FE$="ES FEHLEN"+STR$(AN-A)+" LEUTE."
10800 IFDI<-1THENFE$="ES SIND"+STR$(AN-A)+" LEUTE ZUVIEL."
10805 POKE214,18:PRINT:PRINT"                                ":REM 39 'S
PACES'
10810 POKE214,18:PRINT:PRINT"MI FE$
10815 FORI=1TO4:F$(I)="" :NEXT:GOTO10710
10820 FORI=1TO4:IFF$(I)="C"THENF$(I)="COMPUTER":AN=AN+1
10830 NEXTI
10835 IFAN=1THENPOKE214,18:PRINT:PRINT"DER COMPUTER MUSS MITSPIELEN."
10836 IFAN=1THEN10710
10840 FORI=1TO4:F$(I)=LEFT$(F$(I),19):NEXT
10845 FORW=1TO2000:NEXT
10850 POKE214,18:PRINT:PRINT"OK,ES GEHT LOS.
10860 FORW=1TO1500:NEXT:RETURN
11000 GOSUB20000:PRINT"ES FOLGT EIN DEMONSTRATIONSSPIEL."
11010 FORI=1TO4:F$(I)="C":NEXT
11020 FORW=1TO2000:NEXT:POKE53281,9:AN=0:Z=4:GOTO10820
10970 REM -----
10980 REM   SPIELSTAND LADEN
10990 REM -----
14000 PRINT:IFGN=1THENINPUT"ABAND AN RICHTIGE STELLE GESPULT";U$:PRINT"
14010 OPEN1,GN,0,"SPIELSTAND"
14020 INPUT#1,Z
14030 FORI=1TO4:INPUT#1,F$(I)
14040 FORJ=1TOZ:INPUT#1,SF(I,J):NEXTJ,I:CLOSE1:POKE53280,4
14050 FORI=1TO4
14055 IFF$(I)=""THEN14090
14060 IFASC(F$(I))<101THEN14080
14070 SF(I,0)=ASC(LEFT$(F$(I),1))-100:F$(I)=MID$(F$(I),2,19)
14080 IFASC(F$(I))=42THENS$=I:F$(I)=MID$(F$(I),2,19)
14085 AN=AN+1
14090 NEXT
14100 GOSUB50:X=AN:GOSUB2030
14105 POKESI+5,7:POKESI+6,0
14110 FORI=1TO4:IFF$(I)="" THEN14300
14115 GOSUB2125
14120 RESTORE:FORJ=1TOI:READAF:NEXT
14130 II=0:IFSF(I,1)<>0THENPOKEAF,87:II=1
14140 IFSF(I,2)<>0THENPOKEAF+2,87:II=II+2
14150 IFSF(I,3)<>0ANDZ>2THENPOKEAF+80,87:II=II+3
14160 IFSF(I,4)<>0ANDZ=4THENPOKEAF+82,87:II=II+4
14170 L=0:FORK=1TO4:L=L+SF(I,K):NEXT
14180 IFL=0THEN14300
14200 S=I:F=5:GOSUB3010
14280 Q=44:SF(I,5)=-9+10*I:GOSUB14500:IFII=0THEN14300
14290 GOSUB2835:POKEPF(I,5),176+LL
14291 IFSF(I,0)=0THEN14300
14292 GW=SF(I,0)-1
14294 PRINT"MI";:GOSUB3480:POKESI+5,7:POKESI+6,0
14295 GZ=GZ+1
14300 NEXTI:GW=GZ
14320 GOSUB2125:IFS<>SSTHEN14320
14330 RETURN
14500 FORLL=1TOZ:IFSF(S,LL)<>SF(S,F)ORF=LLTHENNEXT:RETURN
14510 PF(S,LL)=PF(S,F)
14520 MK(S,LL)=MK(S,F):MF(S,LL)=MF(S,F)
14530 MK(S,F)=176+LL:MF(S,F)=FA(S)
14540 II=II-LL:RETURN
20000 U=16:GOSUB20100:U=24:GOSUB20100:U=32:GOSUB20100:U=40:GOSUB20100
20010 U=48:GOSUB20100:POKESI+1,0:RETURN
20100 POKESI+1,U:FORW=1TO40:NEXT:RETURN

```


PROGRAMM-

VERWALTUNG

für den C-64

Das Programm belegt ca. 9,5 kB und ist überwiegend in Basic geschrieben. Es wurde lediglich eine kurze Maschinencode-Routine zum schnellen sortieren von Strings eingebaut.

Von einem Hauptmenue aus können alle Funktionen aufgerufen werden, die teilweise noch ein Untermenue enthalten.

Nach dem Starten des Programmes fragt der Computer, ob man eine bereits auf Diskette gespeicherte Datei laden möchte. Gibt man nur "RETURN" ein, so gelangt man in das Hauptmenue. Andernfalls wird die Datei deren Namen man eingegeben hat, geladen. Existiert die Kartei noch gar nicht, so wird gefragt, ob eine Datei mit dem eingegebenen Namen eröffnet werden soll. (Nur "J" oder "N" eintippen).

Die Pfeil nach links-Taste bzw. das Eingeben von LEERstrings (nur RETURN-Taste drücken) haben eine ESCAPE-Funktion, d. h. die laufende Funktion wird unterbrochen oder aus einem Untermenue in das Hauptmenue gesprungen.

Nach der Bearbeitung der Programm-punkte 1, 2 oder 3 werden die Daten auf Wunsch sortiert und es wird ein Zeigerfeld zum schnellen Zugriff auf die Programmnamen aufgebaut.

ACHTUNG: Der Menüpunkt 4 kann nur ausgewählt werden, wenn die Daten vorher sortiert werden.

Die sechs Funktionen:

DATEN EINGEBEN:

Nach dem Anwählen des Menüpunktes "DATEN EINGEBEN" kann man unter drei Formen der Eingabe auswählen:

a) Eingabe per Hand: Alle Daten müssen manuell eingegeben werden.

b) Direkt von der Diskette: Man legt die Diskette, deren Daten man in die Datei aufnehmen möchte ein und drückt "RETURN". Der Computer zeigt nur die Namen und die Längen der Programme nacheinander an, und man kann auswählen ob man das Programm übernehmen möchte oder nicht. Will man einen Namen übernehmen, so fordert der Computer zur Eingabe der restlichen Daten auf (z.B. Programm-Art).

c) Datei dazuladen: Eine auf der Diskette gespeicherte Datei wird zu der im Speicher stehenden dazugeladen.

2) PROGRAMME LÖSCHEN

Man wird zu Anfang nach dem Namen des zu löschenden Programmes gefragt. Gibt man jetzt "alle" ein, dann werden alle vorhandenen Daten gelöscht.

Bei Eingabe eines anderen Namens werden alle Programme mit diesem Namen gelöscht.

Um unabsichtlichen Löschen vorzubeugen, wurde eine zusätzliche Sicherheitsabfrage eingebaut.

3) DOPPELTE PROGRAMME LÖSCHEN

In diesem Programmteil können doppelt vorkommende Programmnamen gelöscht werden.

Hat der Computer zwei gleiche Programmnamen gefunden, so zeigt er die beiden Einträge vollständig an und fragt, welchen der beiden Namen er löschen soll. Drückt man jetzt die Pfeil nach links-Taste, so überspringt der Computer diese Einträge, d.h. es bleiben beide erhalten.

4) PROGRAMME LISTEN

Dies ist der längste und komplexeste Programmteil. Hat man ihn angewählt, wird man gefragt, ob die Ausgabe auf Bildschirm oder Drucker erfolgen soll. Wählt man die Ausgabe auf dem Bildschirm, wird zusätzlich abgefragt, ob die Bemerkungen oder die Programmdaten ausgegeben werden sollen, da der Bildschirm bei Ausgabe von Bemerkungen und Programm-Daten zu voll und zu unübersichtlich wird. Nun gelangt man in das nächste Untermenue, in dem man das Suchkriterium eingeben muß. Hierbei gibt es vier verschiedene Möglichkeiten:

1. Kein Kriterium, d.h. alle Einträge werden ausgegeben.
2. Suchen nach Programmnamen.
3. Ausgabe bestimmter Programmtypen (z.B. Spiele, Basicerweiterungen, usw.).
4. Inhalte bestimmter Disketten.

5) PROGRAMMLISTE SAVEN

Alle Einträge werden auf Diskette gespeichert.

VORSICHT: Gibt man einen Dateinamen ein, der bereits existiert, so wird die alte Datei überschrieben!!

6) PROGRAMM BEENDEN

Es wird abgefragt, ob man die Daten abspeichern möchte. Gibt man "J" ein, so gilt das unter 5. Gesagte.

Andernfalls wird das Programm beendet.

Das Programm und die Variablen sind noch nicht gelöscht. Möchte man noch weiterarbeiten, so muß GOTO 260 eingegeben werden.


```

840 PRINT "J"; CHR$(142)
850 E$="":PRINT "SUCHEN NACH PROGRAMMENAMEN"; INPUT E$
860 IFE$=" " THEN 260
870 IFLEN(E$)>16 THEN GOSUB 90:GOTO 830
880 G$="":PRINT "SUCHEN NACH DISKETTEN"; INPUT G$: IFE$=" " THEN 260
890 IFLEN(G$)=0 OR LEN(G$)>3 THEN GOSUB 90:ONFL+1 GOTO 830,2220
900 G$=RIGHT$( " " + G$, 3)
910 B$="":PRINT "SUCHEN NACH SEITE 1/2"; INPUT B$: IFE$=" " THEN 260
920 IFLEN(B$)>20 OR (VAL(B$)=0 AND B$<>"00") OR B$=" " THEN GOSUB 90:ONFL+1 GOTO 830,2220
930 B$=RIGHT$( "00" + B$, 2)
940 C$="":PRINT "SUCHEN NACH SPRACHE 1/2"; INPUT C$: IFE$=" " THEN 260
950 IFC$<>"1" AND C$<>"2" THEN GOSUB 90:ONFL+1 GOTO 830,2220
960 PRINT "SUCHEN NACH SPRACHE (1/SP IEL, 2/DISKPRG., 3/BASICERW., 4/USER
970 PRINT " 5/SPRACHE, 6/MUSIK, 7/GRAFIK, 8/HILFS)
980 D$="":PRINT "SUCHEN NACH ART"; INPUT D$: IFE$=" " THEN 260
990 IFVAL(D$)<10 OR VAL(D$)>8 THEN GOSUB 90:ONFL+1 GOTO 830,2220
1000 E$=LEFT$(E$+" ", 19)+B$+"/"+C$+" " + G$+" " + D$
1010 F$="":PRINT "SUCHEN NACH BEMERKUNGEN"; INPUT F$
1020 PRINT "SUCHEN NACH BEMERKUNGEN"; INPUT F$
1030 INPUT F$: IFLEN(F$)>20 THEN GOSUB 90:GOTO 830
1040 IFF$=" " THEN 260
1050 PRINT "ALLES O.K. ? (J/N)";
1060 GOSUB 50: IFA$="J" THEN 1100
1070 IFA$<>"N" THEN 1060
1080 PRINT " "
1090 GOTO 850
1100 E$=E$+F$
1110 A$(PO)=E$:PO=PO+1:A$(PO)=" "
1120 IFPO=507 THEN PRINT "ATEI VOLL !!!":CLOSE 1:GOTO 250
1130 ONFL+1 GOTO 830,2250
1140 GOTO 860
1150 IFPO<2 THEN 230
1160 PRINT "SUCHEN NACH AUSGABE AUF DRUCKER ODER TILDSCHIRM ?"; DR=1
1170 PRINT "SUCHEN NACH AUSGABEGERAET : ";
1180 GOSUB 50: IFA$<>"B" AND A$<>"D" THEN 1160
1190 A1=0: IFA$="D" THEN PRINT "RUCKER":GOTO 1250
1200 IFA$="B" THEN DR=0:PRINT "TILDSCHIRM"
1210 PRINT "VOLL ICH 1 - BEMERKUNGEN"
1220 PRINT "ODER 2 - ATEN AUSGEBEN ?"
1230 GOSUB 50: IFA$<"1" OR A$>"2" THEN 1230
1240 A1=2: IFA$="1" THEN A1=1
1250 PRINT "LEBEN SIE DAS KUCHKRITERIUM EIN : ";
1260 PRINT "1) KEIN KRITERIUM"
1270 PRINT "2) PROGRAMM-/AMEN"
1280 PRINT "3) PROGRAMM-ART"
1290 PRINT "4) ISK-/UMMER"
1300 GOSUB 50: IFA$=" " THEN 260
1310 IFA$<"1" OR A$>"4" THEN 1310
1320 PRINT "J"; CHR$(142):AR=VAL(A$):ONARGOTO 1330,1470,1570,1520.
1330 IFDR=1 THEN OPEN 1,4:CMD1:PRINT R1$;R2$;R3$;R4$:PRINT R1$;:GOTO 1370
1340 PRINT "J";R2$;
1350 IFA1=1 THEN PRINT R4$:GOTO 1370
1360 IFA1=2 THEN PRINT R3$
1370 PRINT "H=0:FORA=1 TO PO-1:IFDR=0 THEN 1390
1380 PRINT LEFT$(A$(A),30);C$(VAL(MID$(A$(A),31,1)));SPC(10);MID$(A$(A),32):GOTO 1
410
1390 IFA1=2 THEN PRINT LEFT$(A$(A),30);C$(VAL(MID$(A$(A),31,1)));GOTO 1410
1400 IFA1=1 THEN PRINT LEFT$(A$(A),18);TAB(19);MID$(A$(A),32):PRINT
1410 H=H+1:IFH<10 OR DR=1 THEN NEXT:IFDR=1 THEN PRINT #1:CLOSE 1:GOTO 260
1420 IF (DR=0 AND H=10) OR A$=POT THEN PRINT "TASTE DRUECKEN ! ";GOSUB 5
0:H=0
1430 IFA=POT THEN 260
1440 PRINT "J";R2$: IFA1=1 THEN PRINT R4$
1450 IFA1=2 THEN PRINT R3$
1460 NEXT:GOTO 1420
1470 PRINT "SUCHEN NACH PROGRAMMENAMEN";
1480 PRINT "NAME"; INPUT S$: IFE$=" " THEN 260
1490 S$="":INPUTS$: IFE$=" " THEN 260
1500 IFS$=" " OR LEN(S$)>16 THEN 1470
1510 GOTO 1690
1520 PRINT "SUCHEN NACH DISKETTEN";
1530 PRINT "DISKETTENNR. "; INPUT S$: IFE$=" " THEN 260
1540 S$="":INPUTS$: IFE$=" " THEN 260
1550 IFS$=" " OR LEN(S$)>20 OR VAL(S$)=0 THEN 1520
1560 S$=RIGHT$( "00" + S$, 2):GOTO 1690
1570 PRINT "J"; CHR$(142):LEBEN SIE DAS KUCHKRITERIUM EIN : ";
1580 PRINT "1) P I E L E"
1590 PRINT "2) ISK-PROGRAMME"
1600 PRINT "3) BASICERWEITERUNGEN"
1610 PRINT "4) SER-PROGRAMME"
1620 PRINT "5) PROGRAMMIERSPRACHEN"
1630 PRINT "6) MUSIK-PROGRAMME"
1640 PRINT "7) GRAFIK-PROGRAMME"
1650 PRINT "8) HILFSPROGRAMME"
1660 GOSUB 50: IFA$=" " THEN 260
1670 IFA$<"1" OR A$>"8" THEN 1570

```



```

1620 S$=A$
1630 IFDR=1THENOPEN1,4:CMD1:PRINTR1$;R2$;R3$;R4$:PRINTR1$;GOTO1730
1700 PRINT"U32";CHR$(142);R2$;
1710 IFA1=1THENPRINTR4$
1720 IFA1=2THENPRINTR3$
1730 PRINT"U":H=0:FU=1:F0=PO-1
1740 IFAR=2THENFU=B%(ASC(S$)-1):F0=B%(ASC(S$))
1750 FORA=FUTOFO
1760 ONAR-1GOTO1770,1790,1810
1770 IFS$(>)LEFT$(A$(A),LEN(S$))THEN1930
1780 GOTO1820
1790 IFS$(>)MID$(A$(A),31,1)THEN1930
1800 GOTO1820
1810 IFS$(>)MID$(A$(A),20,2)THEN1930
1820 IFDR=0THEN1840
1830 PRINTLEFT$(A$(A),30);C$(VAL(MID$(A$(A),31,1)));SPC(19);MID$(A$(A),32)
1840 IFA1=2THENPRINTLEFT$(A$(A),30);C$(VAL(MID$(A$(A),31,1)));GOTO1860
1850 IFA1=1THENPRINTLEFT$(A$(A),18);TAB(19);MID$(A$(A),32):PRINT
1860 IFPEEK(197)=57THENA=F0
1870 H=H+1:IFH<10ORDR=1THENNEXT:IFDR=1THENPRINT#1:GOTO260
1880 IF(DR=0ANDH=10)ORA=F0THENPRINT"U" TASTE DRUECKEN I "U":GOSUB
50:H=0
1890 IFA=F0THEN260
1900 PRINT"U32";R2$;
1910 IFA1=1THENPRINTR4$
1920 IFA1=2THENPRINTR3$
1930 IFAR=2ANDAF0THENA=F0:GOTO1880
1940 NEXT:GOTO1830
1950 PRINT"U";CHR$(142);"U 1) EINGABE PER HAND"
1960 PRINT"U" ODER"
1970 PRINT"U 2) DIREKT VON DISK"
1980 PRINT"U" ODER"
1990 PRINT"U 3) DATEI DAZULADEN?"
2000 GOSUB50:IFA$="+":THENQQ$="6":GOTO260
2010 IFA$<"1"ORA$>"3"THEN2000
2020 IFA$="3"THEN2370
2030 IFA$="1"THENFL=0:GOTO830
2040 GOSUB130:FL=1
2050 Q=0:OPEN1,8,0,"$0":GET#1,A$,B$
2060 GET#1,A$,B$,A$,B$
2070 BL=0:IFA$<"":THENBL=ASC(A$)
2080 IFB$<"":THENBL=ASC(B$)*256+BL
2090 GET#1,B$:IF ST<0THENGOTO2260
2100 GET#1,B$:IFST<0THEN2260
2110 IF B$<CHR$(34)THEN2100
2120 GET#1,B$:C$="":IFB$=CHR$(32)THEN2120
2130 C$=C$+B$:GET#1,B$:IF B$<CHR$(34)THEN2130
2140 GET#1,B$:IFB$<"":THEN2140
2150 E$=LEFT$(C$+" ",18):IFQ=0THENQ=1:GOTO2250
2160 PRINT"U";CHR$(142);"UUEBERNEHMEN (J/N)?"
2170 PRINT"U"NAME : "E$
2180 PRINT"U"BLOCK : "BL
2190 GOSUB50:IFA$="N"THEN2250
2200 IFA$="+":THENCLOSE1:GOTO260
2210 IFA$<"J"THEN2190
2220 PRINT"U"NAME "U-----"U"?"E$
2230 PRINT"U"BLOCK "U---"U"?:BL:PRINT"UUUU"
2240 GOTO880
2250 IF ST=0THEN2060
2260 PRINT"U";CHR$(142);"U -IRECTORY - "NDE"
2270 FORA=1TO1000:NEXT:CLOSE1:PRINT#15,"1":GOTO260
2280 PRINT"U";CHR$(142);"WELCHES PROGRAMM SOLL GELOESCHT WERDEN?"
2290 N$="":INPUTNAME "N$:IFN$="ORLEN(N$)>16THEN2280
2300 PRINT"U"WIRKLICH (J/N)?"
2310 GOSUB50:IFA$<"J"ANDAF$<"N"THEN2310
2320 IFA$="N"THENQQ$="4":GOTO260
2330 IFN$="ALLE"THENFORA=1TO507:A$(A)="":NEXT:GOTO260
2340 PRINT"UUUU":GOSUB2860:PRINT"UUU";
2350 N$=LEFT$(N$+" ",16):FU=B%(ASC(S$)-1):F0=B%(ASC(S$))
2360 FORN=FUTOFO-1:IFLEFT$(A$(N),16)<N$THENNEXTN:GOTO2360
2370 FORT=NTFO-1:A$(T)=A$(T+1):NEXT:PO=FO-1:GOTO260
2380 PRINT:PRINT:PRINT "N$ NICHT GESPEICHERT!"
2390 FORA=0TO1000:NEXT:QQ$="4":GOTO260
2400 FORA=0TO79:R1$=R1$+"":NEXT:R2$="NAME"
2410 R3$="DISK BLK ART "SO=1
2420 R4$="BEMERKUNGEN "QQ$="4"
2430 C$(1)="SPIELE"
2440 C$(2)="DISK"
2450 C$(3)="BASICERW."
2460 C$(4)="USER"
2470 C$(5)="SPRACHEN"
2480 C$(6)="MUSIK"
2490 C$(7)="GRAFIK"
2500 C$(8)="HILFSPR."
2510 A$(0)=CHR$(1)
2520 PO=1

```

```

2530 POKE53281,12:POKE53280,11
2540 POKE650,128
2550 S=0:FORN=828T0977:READX:POKEN,X:S=S+X:NEXT:IFS=17663THEN2570
2560 PRINT"DATA ERROR !!!":STOP
2570 DATA160,0,177,47,48,13,200,177,47,16,8,160
2580 DATA4,177,47,201,1,240,1,96,24,165,47,105
2590 DATA7,133,110,165,48,105,0,133,111,160,0,177
2600 DATA110,240,236,133,34,200,177,110,153,34,0,192
2610 DATA2,208,246,165,110,133,113,165,111,133,114,24
2620 DATA165,113,105,3,133,113,144,2,230,114,160,0
2630 DATA177,113,240,61,133,77,197,34,144,2,165,34
2640 DATA133,85,200,177,113,153,77,0,132,2,208,246
2650 DATA160,0,177,35,209,78,240,4,176,11,144,207
2660 DATA200,196,85,208,241,196,34,176,198,160,2,177
2670 DATA110,170,177,113,145,110,153,34,0,138,145,113
2680 DATA136,16,240,48,178,24,165,110,105,3,133,110
2690 DATA144,143,230,111,208,139
2700 GOTO510
2710 PRINT" "
2720 GOSUB2960
2730 FORN=1TOPO-1:IFLEFT$(A$(N),16)=LEFT$(A$(N+1),16)AND A$(N)<>" "THEN GOTO2750
2740 NEXTN:GOTO260
2750 PRINT" "CHR$(142);" WELCHER EINTRAG SOLL GELOESCHT WERDEN ? "
2760 PRINT" 1  : "
2770 PRINTLEFT$(A$(N),30);C$(VAL(MID$(A$(N),31,1))):PRINTMID$(A$(N),32)
2780 PRINT" 2  : "
2790 PRINTLEFT$(A$(N+1),30);C$(VAL(MID$(A$(N+1),31,1))):PRINTMID$(A$(N+1),32)
2800 PRINT"  : BITTE '1' ODER '2' DRUECKEN !!! "
2810 GOSUB50:IFA$(N)<"1"AND A$(N)<"2"AND A$(N)<" "THEN2810
2820 IFA$="1"THEN2740
2830 IFA$="2"THENN=N+1
2840 FORM=NTOP0-1:A$(M)=A$(M+1):NEXT:PO=PO-1:A$(PO)=" "
2850 N=N-1:GOTO2740
2860 PRINT" "CHR$(142);" BITTE WARTEN !!!":RETURN
2870 PRINT" "CHR$(142);" WELCHE DATEI SOLL GELADEN WERDEN ? "
2880 W$=" ":INPUT"NAME "W$:IFLEN(W$)>16THEN2880
2890 IFW$=" "THEN260
2900 GOSUB180
2910 CLOSE2:QQ=0:OPEN2,S,2,W$+" ",S,R"
2920 PL=0:INPUT#2,QQ:IFA$(PO)=" "THENPL=1
2930 IFQQ=0THENIR=PO:GOTO2980
2940 IFPO=1THENPO=2
2950 FORIR=PO-1+PLTOP0+QQ-2
2960 INPUT#2,A$(IR)
2970 NEXTIR
2980 CLOSE2:A$(IR)=" ":PO=IR:GOTO260
2990 SYS828
3000 IFA$(PO-1)=" "THENPO=PO-1:GOTO3100
3010 FORA=0TO256:B%(A)=0:NEXT
3020 FORA=1TOPO-1
3030 B%(ASC(A$(A)))=B%(ASC(A$(A)))+1
3040 NEXT
3050 Q=0:Q1=0:FORA=0TO255
3060 Q=B%(A+1)+Q
3070 B%(A)=Q1:Q1=Q
3080 NEXT
3090 PRINT"
3100 IFA$(PO-1)=" "THENPO=PO-1:GOTO3100
3110 RETURN
11111 PRINTPEEK(197):GOTO11111

```

**Speicherplatz geklaut?
oder
was passiert eigentlich mit dem 64KRAM des C-64?**

Viele Käufer eines Commodore 64 wundern sich nach dem ersten Anschließen des Gerätes. Der C-64 meldet sich mit 38K freiem Speicherplatz. Da der C-64 jedoch 64KRAM hat, bleibt häufig die Frage offen: Wohin gehen die restlichen 26 KByte?

Eine weitverbreitete Meinung besagt, daß der restliche Speicherplatz für die hochauflösende Grafik benötigt wird. Dies kann aber so nicht gelten.

Denn ca. 20K haben im normalen Basic-Mode über-

haupt keine Funktion.

Wie das?

Dazu müssen wir uns den Mikroprozessor 6510, auf dessen Basis der C-64 arbeitet, etwas genauer ansehen.

Wie die Maschinensprach-

programme, die Befehle LDA, JMP, JSR usw. nur bis zur Adresse 65535 in Assembler zu benutzen.

Der Commodore 64 hat einen Festwertspeicher (ROM) von 20KByte, davon werden 16K für das Basic belegt (Inter-

preter).

Um dieses Problem zu lösen, benutzt die Maschine ROM/RAM-Overlays. Das bedeutet, das einige Speicheradressen im ROM und RAM überlappen. Wenn das ROM angesprochen wird, überlappt

es einen Teil des oberen RAM-Bereichs. Dieser RAM-Block ist versteckt und kann nicht adressiert werden. Leider werden im Basic-Modus 16K ROM belegt, sodaß auch 16K von RAM blockiert sind. Die restlichen 4K werden vom Basic-Block getrennt und können nicht in Basic adressiert werden. Untenstehende Tabelle zeigt die Adressen, wo ROM und RAM überlappen.

HEX	DEZ
FFFF	65535
F000	61440
E000	57344
D000	53248
C000	49152
B000	45056
A000	40960
9000	36864
8000	32768
7000	28672
6000	24576
5000	20480
4000	16384
3000	12288
2000	08192
1000	04096
0800	02048
0000	00000

E000=16K RAM
 0000-8000=32K Basic
 8000-A000=8K RAM
 A000-C000=8K Basic-ROM
 C000-D000=4K RAM
 (versteckt)
 D000-E000=4K
 Ein/Ausgabe-RAM
 E000-FFFF=8K Kernal ROM

Von der Adresse \$0800 bis \$9FFF liegt der Speicherbereich für Basicprogramme. Von \$A000 bis \$BFFF liegt in der Regel das Basic-ROM. Dies kann jedoch beim Programmieren in Maschinencode ausgeschaltet werden, sodaß der dadurch freigewordene Bereich vom Maschinencode mitbenutzt werden kann.

In Basic kann es nicht ausgeschaltet werden, da dieser Teil des Speichers dem Mikroprozessor mitteilt, was die Basic-Befehle bewirken. Von \$C000 bis \$CFFF liegen 4K RAM. Dieser Bereich kann

nur im Maschinencode angesprochen werden, da dieser vom Basic-Hauptblock (\$0800-\$9FFF) getrennt ist. Von \$D000 bis \$DFFF liegt der Ein/Ausgabespeicher, welcher das Farbram und die SID- und VIC-Chip-Register beinhaltet. Das Zeichensatz-ROM liegt ebenfalls in diesem Teil, er ist aber normalerweise ausgeschaltet. Er wird meistens angesprochen, wenn selbstdefinierte Grafikzeichen (UDG's) benutzt werden.

Zum Schluß, von Adresse \$E000 bis \$FFF liegt das Kernal-ROM.

Es ist unpraktisch, das Kernal-ROM auszuschalten, da die meisten Maschinecodeprogramme auch das Kernal-Rom benutzen. Um dieses ein wenig praktisch zu veranschaulichen, poken wir:

POKE 40960 (Start Basic-ROM), 0-255

und anschließend

Print PEEK (40960)

Wir stellen fest, daß als Ergebnis immer 148 erscheint. Es kann also in das ROM kein Wert eingepoket werden. Die Adresse 0001 entscheidet, welcher Speicherbereich ROM oder RAM ist. Dabei werden nur die ersten 3 Bits für die Erkennung benötigt.

Adresse 0001

Bit	Name	Block	Speicher (wenn Bitgesetzt)
0	LORAM	\$A000-\$BFFF	Basic-ROM
1	CHAREN	\$D000-\$DFFF	Ein/Ausgabe RAM
2	HIRAM	\$E000-\$FFFF	Kernal ROM

Wenn wir jetzt **Poke 1, PEEK (1) AND 254** eingeben, wird Bit 0 auf 0 gesetzt und das Basic-ROM ausgeschaltet, der RAM-Bereich wird frei. Damit werden aber auch die Basic-Befehle PEEK und



Wir haben fast alles für Ihren VC-20/64

Software:
 Flugsimulatoren, Biorhyt., Psycho, Lottoberechnung, Krankheitsdiagnose, Textverarbeiter, Dateiprogramm, Maschinensprachemonitor, Buchhalter 64, Flight II (Sublogic), Quickcopy, Teleterm 64, Schachprogramm, Diskmanager und viele, viele Spiele.....

Zubehör:
 Staubschutzhauben, Resettaster, Stecker, Bauteile, Bücher, Tastaturmaske, Disketten + Boxen, Hifi-Kabel.

Hardware:
 5x schnellere Floppy, Schnell-Save-Modul, 10er Tastatur, Speichererweiterungen 8K-64K RAM, Moduladapter 2/3/5-fach, Telefonmodems, 40/80 Zeichenkarten, Eprommer + Karten, Toolkitmodul, Joysticks, PIO In/Out-Module, Interface, RS 232, IEEE 488, und...und...und...und...

Prüfen Sie unser Angebot 

Schnell den Gesamtkatalog 4/84 anfordern, für 2,50 DM (Briefmarken) 24-Std. Katalogversand.

Händleranfragen erwünscht.



Rotdornweg 15
 1000 Berlin 45
 ☎ 030-817 38 57
 341 45 73

POKE, die es nur im Basic gibt, ausgeschaltet und der Computer "stürzt" ab. Will man den "versteckten" RAM-Bereich unter dem Basic-ROM benutzen, so muß in Maschinencode programmiert werden. Nachfolgend ein kurzes Programm, daß eine kurze MC-Routine enthält.

```
10 Data 169,254,45,1,0,141,1,0,173,0,160,141,110,195,169,1
20 Data 13,1,0,141,1,0,96
30 FORA = 50000 to 50022:REM MC in Speicher
40 READD:POKEA,D:NEXT
42 SYS50000:REM M/C Aufruf
45 N=PEEK(50030):REM
50 PRINT"
```

Es schaltet das Basic-ROM aus und sucht die Adresse, die wir versucht haben, einzupoken.

Die Routine legt den Wert in Adresse 50030 ab, welche vom Basic-Bereich aus angesprochen werden kann. Sie schaltet das ROM bevor es wieder in den Basic-Bereich zurückspringen kann. Nach dem Starten des Programmes finden wir den Inhalt der RAM-Adresse 40960. Versuchen Sie doch

einmal irgendeine Zahl nach 40960 zu poken. Wenn Sie jetzt **PEEK 40960** eingeben, wird immer noch der Wert 148 aus dem ROM-Bereich darin stehen. Starten Sie jedoch das Programm, so bekommen Sie den Wert, der von Ihnen in den "versteckten" RAM-Bereich gepoket wurde.

Für MC-Benutzer folgt ein kurzes Disassembler-Listing:

```
C350 LDA# $FE
C352 AND $0001
C355 STA $0001
C358 LDA $A000
C35B STA $C36E
C35E LDA $01
C361 ORA $0001
C364 STA $0001
C367 RTS
```

Dieser POKE schrieb tatsächlich etwas in das "versteckte" RAM, wir konnten es nur nicht "peeken", da die Adresse 40960 die unveränderbare Nummer des ROM-Bereichs hielt.

CM-LEXIKON

Es folgt Teil 4 unseres "Compute mit"-Wörterbuches. Mit jeder Ausgabe erhöht sich Ihr Sprachschatz, das Computeralphabet wird immer durchsichtiger!
Noch immer ist genügend Zeit einzusteigen, falls Sie nicht von Anfang an dabei sind.
Denn "Compute mit"-Leser können mitreden!

Datenträger:

Jedes System, das in der Lage ist, Computerdaten aufzunehmen und zu speichern. Dazu gehören unter anderem Kassettenrecorder, Disketten, Festplatten und Laser-Bildplatten. Der wohl älteste Datenträger ist der Lochstreifen.

Debugger:

Ist ein Hilfsprogramm, welches Fehler im Programmablauf findet und anzeigt.

Digital/Analog-Wandler:

Ist eine elektronische Schaltung, die aus digitalen Zahlen in die für Menschen verständliche Signale verwandelt. So können z.B. vom Kassettenrecorder Töne in den Computer überspielt werden, die dieser in Form von Binärzahlen (0-1) aufnimmt. Der D/A-Wandler bewirkt nun, daß diese digitalen Töne für den Benutzer in Ton oder Schrift wiedergegeben werden.

Directory:

Als Directory wird das Inhaltsverzeichnis eines Datenträgers bezeichnet. Man bekommt einen besseren Überblick über vorhandene Programme oder Dateien. Im allgemeinen stehen neben dem Programmnamen die Länge des jeweiligen Programmes und der freie Speicherplatz im Directory.

Direkter Speicherzugriff:

Ohne daß die CPU (Zentraleinheit) angesprochen wird, kann der Benutzer Infor-

mationen aus einem bestimmten Speicher holen und lesen.

Disassembler:

Ein Programm, das ein Computer-Anwendungsprogramm in Maschinensprache wieder in die Assemblersprache zurückverwandelt.

DOS:

Abkürzung für Disketten-Operation-System. Das DOS ist verantwortlich für den Datenverkehr mit den angeschlossenen Peripheriegeräten. Disketten-Betriebssysteme sind u.a. CP/M, Unix, PC-DOS. Alle werden auf Diskette geliefert und um Speicherplatz zu sparen, werden Unterprogramme des DOS nur dann geladen, wenn sie auch wirklich gebraucht werden (z.B. Kopie von Daten).

Datenbank:

Gesamtheit aller Dateien in einem Dateiverwaltungssystem. Datenbanken kann man mit Lexika vergleichen. Ein Sammelwerk von Daten, die auf Abruf gelesen, zugeordnet und gespeichert werden. Verschiedene Datenbanken können heute mit Hilfe eines Akustikkopplers angerufen werden; Entfernungen spielen keine Rolle mehr.

Datenbus:

nennt man die Leitungen, auf denen die Daten der verschiedenen Computerbaugruppen transportiert werden. Über den Datenbus werden z.B. Mikroprozessor- und Speicherbausteine bzw. Ein/Ausgabeelemente miteinander verbun-

den. Die Wortbreite (8bit, 16bit usw.) ist maßgeblich für die Anzahl der Leitungen. Zu beachten ist, daß dabei die Daten in beide Richtungen fließen.

Datenflußplan:

Ist die schrittweise Darstellung eines Programmablaufs. Es werden genormte Symbole verwandt. Ein Datenflußplan beinhaltet im wesentlichen die Art des Datenträgers (Diskette, Kassette), die Art der Bearbeitung (drucken, Display) und die Wege des Datenträgers. Der Flußplan zeigt z.B., wie Werte eingetastet werden, wie sie auf dem Bildschirm zu sehen sind, in den Speicher gelangen und von dort auf den Datenträger ausgegeben werden.

DIAGNOSEPROGRAMM:

Überprüft alle Systemeinheiten des Computers auf einwandfreie Funktion. Bei größeren Rechnern ist solch ein Programm im ROM untergebracht, aber auch auf Heimcomputern gibt es per Software Diagnoseprogramme.

DISPLAY:

andere Bezeichnung für optische Anzeige. Ein Display kann eine LCD-Anzeige, ein Bildschirm, Leuchtdioden-anzeige oder ein Plasma-Display sein. Jede optische Darstellungsart von Computerdaten heißt Display.

DISTRIBUTOR:

Ein Händler von Computer-Systemen oder Computer-Software in einem bestimmten Vertriebsgebiet.

Hardcopy des Bildschirms bei Text und Blockgrafikmode bei C-64

Tips & Tricks

```

11000 OPEN4,4
11005 FORV=1TO25
11010 QC$=""
11020 FORW=1TO40
11030 QC=PEEK(1024+(V-1)*40+W-1)
11033 IFQC<128THENGOSUB11100:QC$=QC$+CHR$(QC)
11035 IFQC>127THENGOSUB12000
11040 NEXTW
11050 PRINT#4,QC$
11060 NEXTV
11070 CLOSE4
11080 RETURN
11100 REM SET1 -> ASCII
11131 IFQC<32THENQC=QC+64:GOTO11134
11132 IFQC>63ANDQC<96THENQC=QC+32:GOTO11134
11133 IFQC>95ANDQC<128THENQC=QC+64
11134 RETURN
12000 REM REVERSE
12010 QC$=QC$+CHR$(27)
12020 QC$=QC$+" "
12030 QC=QC-128
12040 GOSUB11100
12045 QC$=QC$+CHR$(QC)
12050 QC$=QC$+CHR$(27)
12060 QC$=QC$+" "
12070 RETURN
    
```

Listing

Für den VC-20 ändert man
Zeile 11020 in
For W = 1 to 20

Poke's für den VC-20

Poke650,255: Dauerfunktion für alle Tasten
Poke650,127: Ausschalten der Cursor-Dauerfunktionen
Poke788,194: Macht Stop unwirksam
Poke36881,x(x=0-127): Bildstand von oben nach unten
Poke657,128: Blockiert Shift-Taste
Poke646,x(x=0-255): Verändert eingebauten Zeichensatz
Poke36869,242: Umschaltung von Groß- in Kleinmodus
Poke36869,240(192): Normal
Poke36869,255: Ausgangsbasis für selbstdefinierte Zeichen
Poke37877,0: Zeitlupe für's Listen
Poke775,200: Listschutz ein
Poke775,167(199): Listschutz aus
Poke650,64: Dauerfunktion für alle Tasten aus
Poke37154,117: Speert alle Tasten außer ungerade Zahlen
1 Print Peek(197):Run: Schreibt, welche Taste gedrückt wurde

SYS64802: Setzt in den Einschaltzustand
OREM'Shift 1': Schützt Zeile

Print Peek(57): Zeigt, in welcher Zeile gestoppt wurde

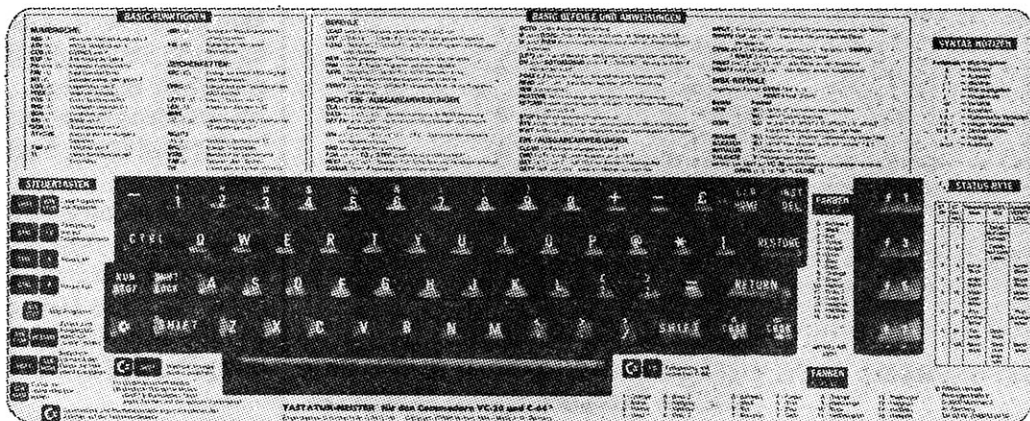
\$: String
A\$: Chr\$(14)
b\$: "Name"
save a\$+b\$: Bewirken das Abspeichern im Klein-Modus
 Christian Klose

TASTATUR-MEISTER® ist da!

Gibt die schnelle Referenz für Befehle, Optionen und Formate direkt an den Arbeitsplatz -

BASIC
Funktionen

STEUER-
TASTEN
Schneller
Überblick



SYNTAX-
NOTIZEN

STATUS-
BYTE
Referenz

FARBEN
Tabellen

ASCII Tabelle zum Aufhängen
(Nicht abgebildet)

TASTATUR-MEISTER®

- Befehle vollständig vorhanden, mit voll ausgeschriebenem Befehlsbeschreibung
- Alle Steuertasten übersichtlich

SCHABLONE aus: ● Widerstandsfähigem Kunststoff
● Dauerhaftem Druck
● Eingeteilt nach Nutzungsbereichen

Händeranfragen erwünscht. Hersteller:

FÜR COMMODORE VC 20 und C-64

- BASIC x á DM 29,90
- SIMON'S BASIC x á DM 29,90
- MULTIPLAN x á DM 29,90
- BLANKO x á DM 27,00

(für eigene Anwendungen)
Schablonen auch für IBM-PC erhältlich.

Das 1.1/2.0, BASIC 1.1/2.0 WORDSTAR, MULTIPLAN
LOTUS 1,2,3, BLANKO

FRIWA-Vertrieb · Reisingerstr. 6 · 8 München 2 · Tel. 0 89/53 04 50 · Telex 5 213 775

Senden Sie bitte TASTATUR-MEISTER:

- ☐ per Nachnahme (zzgl. NN-Gebühr und 5,- DM Versandgebühr)
☐ per Scheck (zzgl. 5,- DM Versandgebühr) (liegt bei)

Name

Straße

PLZ Ort

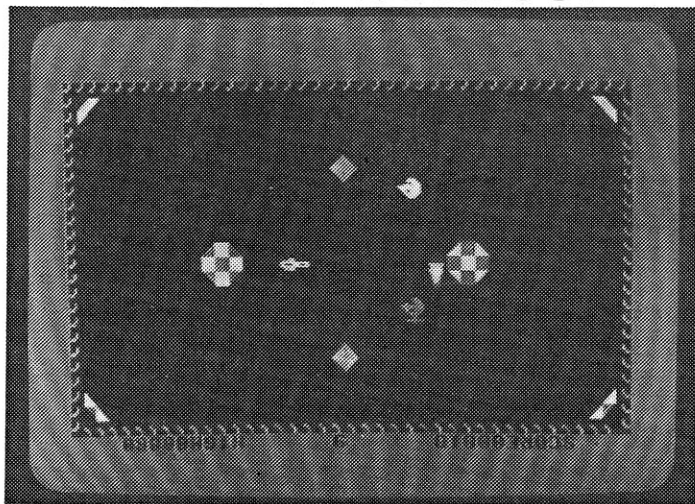
NEU FÜR DEN COMMODORE C-64



**Die revolutionäre,
neue Lernmethode.**

SM SOFTWARE AG / MÜNCHEN

"Cosmic Bouncer"



Zwischen zwei Planeten pendelt eine Raumfähre, die von feindlichen Gestalten angegriffen wird. Mit Ihrem Raumschiff sichern Sie den Weg der Weltraumfähre. Dazu steht Ihnen ein Energieball zur Verfügung, der aber immer wieder aufgefangen werden muß, bevor ein Angreifer damit unschädlich gemacht werden kann. Sie haben insgesamt neun Schiffe für Ihre Mission.

Manchmal müssen Sie ein Schiff opfern, um die Zerstörung der Raumfähre zu meiden.

☆☆☆

boote und Abwehrjäger machen Ihnen das Leben zur Hölle.

Da hilft Ihnen nur Höhe zu gewinnen und alles abzuhängen, was hinter Ihnen her ist. Die Steuerung des Hubschraubers ist einzigartig realistisch.

Im Grundzug kann man das Spiel als "Ballerspiel der 3. Generation" bezeichnen.

Leider ist es noch nicht auf dem deutschen Markt erhältlich, aber wir sind sicher, daß sich ziemlich schnell ein Anbieter für diese tolle Simulation findet und Sie sich zuhause den Gefahren der Bungeling Bay stellen können.

☆☆☆

(tm)

"Megahawk"

In diesem unglaublich schnellen Weltraumspiel in dreidimensionaler Grafik müssen Sie den Angreifern, die sich Ihnen in den Weg stellen, ausweichen oder eliminieren. In immer neuen Bildern tauchen Gefahren von allen Seiten auf, die es zu meistern gilt.

Drücken Sie den Feuerknopf und schon beginnt das Spiel.

Raid on Bungeling Bay



Neues von der amerikanischen Firma Broderbund.

Der Filmhit 'Das fliegende Auge' war Grundlage für dieses neue Spiel. Sie selbst steuern einen der modernsten und gefährlichsten Kampfhubschrauber der Welt.

Ihre Aufgabe ist es, nach dem Start von einem Flugzeugträger, feindliche Stellungen zu finden und zu vernichten.

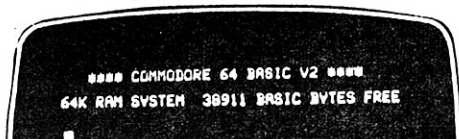
Der Hintergrund des Spiels ist der, daß Ihr Auftrag so geheim ist, daß niemand von Ihrer eigenen Armee informiert werden durfte.

Folge davon ist, daß die Geschützstellungen im eigenen Land auf Sie schießen

und Sie nach Möglichkeit so wenig wie möglich Flaks der eigenen Abwehr vernichten. Dies ist allerdings sehr schwer, da Sie in dem wirklich hektischen Geschehen sehr schnell den Überblick verlieren können.

Nach dem Start müssen Sie erst einmal über ausgedehnte Hafenanlagen fliegen und das offene Meer erreichen. Während dieses Überflugs feuern allerdings ohne Unterbrechung Flaks und Panzer auf Sie.

Haben Sie es trotzdem geschafft, das Meer zu erreichen, so bleibt Ihnen allerdings auch hier keine Zeit zum Ausruhen, die feindlichen so wie die eigenen Schnell-



Basic Kurs

Teil 5

Die in der letzten Folge gestellte Aufgabe war recht einfach zu lösen, wenn man die Funktionsweise der FOR...NEXT Schleife verstanden hatte. Die fehlende Befehlszeile mußte natürlich die Nummer 111-119 tragen. Die korrekte Lösung würde also z.B.:

```
115 NEXT T
```

heißen. Ansonsten hatte sich ja an dem Programm nicht viel verändert, außer den drei Zeilen 90,100,110. In diesen Zeilen wurden die Variablen auf den Anfangswert zurückgesetzt.

Mit dem Erlernen dieser Schleife sind wir unserem Endziel, nämlich richtige Programme zu schreiben, wieder ein ganzes Stück näher gerückt, auch wenn man die vollständigen Anwendungsmöglichkeiten dieses Befehls noch nicht erkennen kann. Wir werden in dieser Folge jedoch weiter mit der FOR... NEXT Schleife arbeiten und noch einen weiteren Befehl erlernen.

Die Sprunganweisung GOTO

Dieser Befehl gilt in der Struktogrammprogrammierung als verpönt, da man bei häufiger Anwendung sehr schnell den Überblick über sein Listing verliert. Er gehört jedoch auch zu den Befehlen, die BASIC so komfortabel machen und soll deswegen nicht ausgelassen werden. GOTO kann man direkt mit 'GEHE NACH' übersetzen.

Die Eingabe dieses Befehls erfolgt immer in der Form:

```
10 GOTO 100
```

Die obige Zeile bewirkt nichts anderes als, da der Interpreter, wenn er in Zeile 10 kommt zur Zeile 100 springen soll und alle dazwischen liegenden Zeilen nicht zu beachten.

Man braucht diesen Befehl häufig am Ende von Routinen, die so oft wiederholt werden wollen, bis irgendeine Situation gegeben ist. Dies kann der Wert einer Variable sein oder ein Tastendruck. Ab-

gefragt werden diese Werte mit einer bedingten Sprunganweisung, welche wir auch noch in dieser Folge erklären werden.

Es handelt sich um:

Die bedingte Sprunganweisung IF...THEN GOTO

Übersetzt heißt das soviel wie 'WENN...-DANN GEHE NACH'.

Mit IF wird also irgendeine Gegebenheit abgefragt oder verglichen.

Vergleichsoperatoren können dabei =(ist gleich), < > (ist ungleich), < (ist kleiner als), > (ist größer als) oder die Booleschen Logikwerte 'wahr' und 'unwahr' sein.

Zumeist werden Zustände von Variablen abgefragt. Dies geschieht in der Form:

```
10 IF A=3 THEN GOTO 100
```

Die obige Zeile hieße im Klartext:

Wenn die Variable A den Wert 3 angenommen hat, dann gehe nach Zeilennummer 100, andernfalls nimm die nächste Zeile nach 10. Man kann also unter Zuhilfenahme dieser Anweisung seinen Interpreter dazu veranlassen, Entscheidungen zu treffen.

Aber man kann an die Befehlsfolge IF...THEN nicht nur Sprunganweisungen anhängen, sondern auch jeden anderen Basic-Befehl.

Man könnte also auch schreiben:

```
10 IF A=3 INPUT "NEUER WERT FUER A";A
```

oder

```
10 IF A=3 THEN B=4
```

oder

```
10 IF A=3 THEN PRINT "A HAT JETZT GERADE DEN WERT 3"
```

Sinnvolle Anwendungen für diese Anweisung gibt es so viele, daß wir sie in einer Lektion gar nicht alle bringen können, darum werden wir auch diesmal nur zeigen, wie man mit dieser Befehlsgruppe frühzeitig eine FOR...NEXT Schleife verläßt.

Wir geben ein:

```
10 FOR T=1 TO 30
20 INPUT B$
30 PRINT B$
40 IF B$ = "XXX" THEN GOTO 60
50 NEXT T
60 PRINT "SIE HABEN SOEBEN UNSERE
SCHOENE EINGABESCHLEIFE VERLASSEN!!"
70 INPUT "WOLLEN SIE NOCH EINMAL ANFANGEN (J/N)";
A$
80 IF A$="J" THEN GOTO 10
90 IF A$="N" THEN PRINT "NA GUT,
DANN HALT NICHT."
```

Sie können, nachdem Sie mit RUN gestartet haben, irgendwelche Worte oder Texte eingeben und der Computer wird diese nach Drücken von RETURN auf den Bildschirm schreiben.

Sie verlassen die Schleife automatisch nach Eingabe des 30. Wortes oder vorher, indem Sie XXX eingeben und abschicken.

Danach fragt Sie das Programm, ob Sie noch einmal beginnen wollen.

Hier liegt jedoch auch die Aufgabe zur nächsten Lektion. Unser Programm hat nämlich noch einen kleinen Fehler. Wenn Sie in der letzten Frage etwas anderes als 'J' oder 'N' eingeben, springt der Computer zum Ende des Programms.

Finden Sie also zum nächsten Heft die richtige Zeile, die dies verhindert.

Viel Spaß!

(tm)

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

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

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

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

 -Datentechnik

Telemannstraße 18
7250 Leonberg (Höfingen)
☎ 0 71 52 / 63 05



Liebe Rätsel Freunde,
Ihre "Compute mit" wird nicht müde sich Woche für Woche eine neue Tüftelei auszudenken. Heute haben Sie einen sogenannten "Rösselsprung" vor sich, den Sie bestimmt schon aus dem einem oder anderen Rätselmagazin oder der Wochenendbeilage Ihrer Tageszeitung kennen.

Seinen Namen verdankt er der besonderen Art des Springens (geradeaus oder quer bzw. umgekehrt), die dem Springerzug beim Schach entspricht.

Wir beginnen bei dem durch (*) gekennzeichneten Feld und notieren die Buchstaben derjenigen Felder, die auf unserem Weg angesprungen werden. Das letzte Feld ist wieder markiert (o).

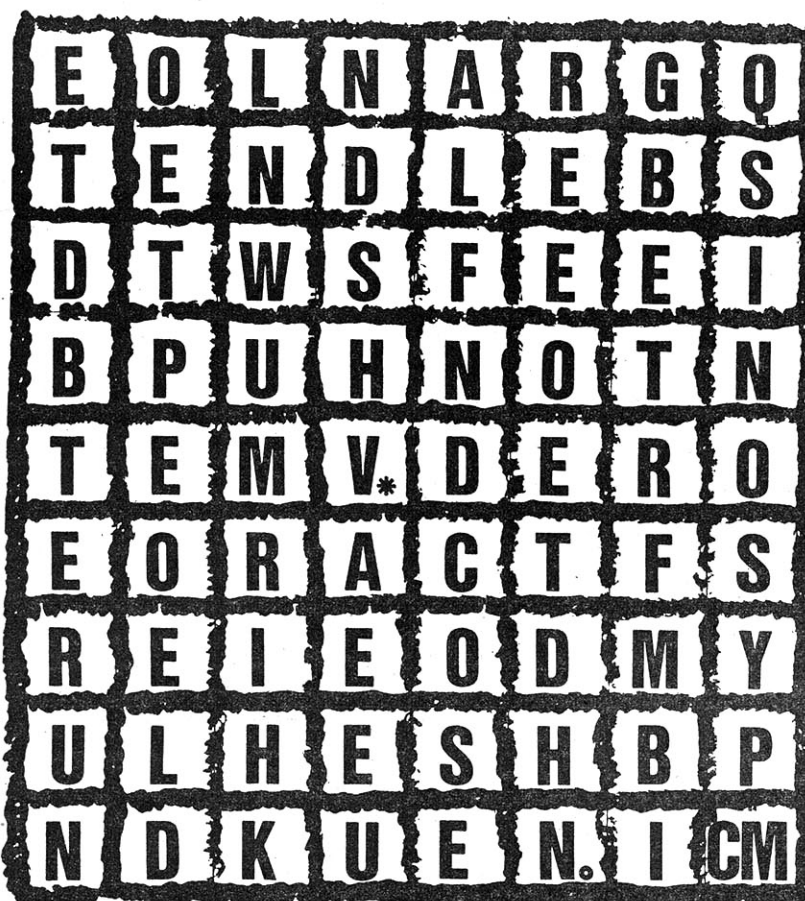
Wenn Ihr Pferd richtig über die Hecken geht, die die Felder abgrenzen, so ergibt sich ein Rat, der Ihnen mehr Freude an Ihrem Programm bringt.

Ein Hinweis ist angebracht:

Wie beim Schachspiel kann vor und zurückgesprungen werden. Felder (und damit auch Buchstaben) können doppelt, ja unter Umständen dreifach angesprungen werden. Umgekehrt ist durchaus möglich, daß das eine oder andere Feld links liegen gelassen wird. Vorsicht ist in jeder Hinsicht geboten.

Den Lösungssatz schreiben Sie wieder auf eine Postkarte, frankieren ordentlich und schicken das Ganze an den

Roeske-Verlag
"Compute mit"
Suchrätsel
Postfach 620
D-3440 Eschwege



Die Gewinner aus Heft 38/84

1. Preis

H. Lang, 6057 Dietzenbach

2. Preis

G. Hirsemann, 4709 Bergkamen

3. Preis

J. Kuthleick, 2400 Lübeck

Hier die Gewinner für je 1 Gutschein:

L. Becker, Göttingen
M. Timmerman, Moers
B. Endress, Stuttgart
J. Stegemann, Brunsbüttel
D. Speer, Münster
E. Kraiser, Stuttgart
B. Eigenmann, Sulzfeld
P. Werner, Berlin
T. Wagner, Dortmund
M. Emmel, Nürnberg
A. Gerbern, München
W. Torget, Stuttgart
P. Bernhard, Gladbach
T. Tiehle, Bergkamen
K. Reinermann, Stade
M. Hirnes, Paderborn
P. Langemann, Essen
B. Selbald, Köln
S. Schmider, Fulda
O. Wernicke, Hannover
A. Brodel, Göttingen
St. Rudolph, Emden

Wie immer, so kann auch diesmal wieder jeder mitmachen, mit Ausnahme der Mitarbeiter des Roeske-Verlages und deren Angehörige. Der Rechtsweg ist ausgeschlossen. Einsendeschluß ist Samstag, der 20. Oktober 1984 (Datum des Poststempels).

Toi, toi, toi!

Zu gewinnen gibt es wie immer Gutscheine aus unserem Software-service.

SOFTWARE

**Super-Angebote
zum Zugreifen!**

WICOSOFT

VC-20

VC1000	Gridder Superarcade (o. Erw.)	DM 29.00
VC1023	Gridtrap Labyrinth (o. Erw.)	DM 35.00
VC1024	Rescue Weltraumaction (o. Erw.)	DM 29.00
VC1001	Space Attack Arcade (o. Erw.)	DM 29.00
VC1005	Martian Raider Arcade (o. Erw.)	DM 29.00
VC1007	Moons of Jupiter (+8K)	DM 29.00
VC1018	Line up 4/Reversi (o. Erw.)	DM 29.00
VC1020	Get lost Labyrinth (o. Erw.)	DM 29.00
VC1025	Penny Slot (o. Erw.)	DM 25.00
VC1028	Power Blaster Arcade (o. Erw.)	DM 29.00
VC1040	Invaders Arcade (o. Erw.)	DM 24.00
VC1003	Fluch des Pharao Adv. (+ 16K)	DM 19.50
VC1029	Pedes & Mutants Action (o. Erw.)	DM 29.00
VC1041	Log Run Action (o. Erw.)	DM 24.00
VC1008	Shark Attack Action (o. Erw.)	DM 32.00
VC1011	Atom Smasher Arcade (o. Erw.)	DM 32.00
VC1034	Insector Arcade (o. Erw.)	DM 32.00

Bücher

BC9017	Commodore 64 Exposed	DM 35.00
BV9016	VIC 20 Exposed	DM 35.00
BV9008	Spiele für Ihren VC-20	DM 14.80

Commodore 64

CB2004	Hungry Horace Arcade u. Spaß	DM 29.00
CB2010	Krazy Kong Arcade Spiel	DM 29.00
CB2012	Star Trek Arcade Spiel	DM 35.00
CB2015	Zappy Zooks Arcade Spiel	DM 29.00
CB2019	Lander Arcade Spiel	DM 35.00
CB2022	Pakacuda Arcade	DM 32.00
CB2029	Stellar Dodger Arcade Spiel	DM 29.00
CB2030	Jammin Arcade u. Musik	DM 29.00
CB2031	Jammin Diskette	DM 35.00
CB2032	Pipeline Arcade u. Spaß	DM 29.00
CB2033	Pipeline Diskette	DM 35.00
CB2003	Gridder Superarcade Spiel	DM 29.00
CB2005	Dickys Diamonds Arcade u. Spaß	DM 35.00
CB2007	Panic Arcade Spiel	DM 29.00
CB2011	Frogger Arcade Spiel	DM 29.00
CB2014	Exterminator Arcade Spiel	DM 29.00
CB2020	Galaxy Weltraum-Arcade	DM 35.00
CB2013	Multisound Synthesizer	DM 65.00
CB2021	Stix Weltraum-Arcade	DM 45.00
CB2024	Purple Turtles Arcade u. Spaß	DM 35.00

Abtrennen und absenden an:

**WICOSOFT, Christian Widuch,
Nordstraße 22, 3443 Herleshausen**

Bitte liefern Sie

- ☐ per Nachnahme (zzgl. Gebühren)
☐ per Vorkasse bzw. Scheck

Anzahl	Artikelbezeichnung	Preis

Name:

Adresse:

STOP +++ Stark reduzierte Preise