

Compute mit

39/84

Das unabhängige Magazin für Anwender und Interessenten von Commodore Computern

2,80 DM
24 öS
2,80 sF



NEU!

**Spezialrecorder für die bekanntesten
Micros MC 3810 von BOSTON COMPUTER**

Trotz Floppies und Microdrives ist und bleibt der Kassetten-Recorder Massenspeichereinheit Nr. 1 und Top of the charts in der Beliebtheitsskala der Computer-Anwender. Leider sind die gängigen Geräte vom Hersteller in der Regel für Musikaufnahmen konzipiert, genügen also dem EDV-Mann nicht immer. Hier kommt der neue MC 3810 gerade richtig. Prüfen Sie ihn gleich selbst auf Seite

LOGIK im Basic-Programm (Teil 3)

Für alle, die mit dieser neuen und raffinierten Serie sich selbst in Sachen Programmier-technik auf Vordermann bringen wollen, erscheint heute pünktlich die 3. Folge, in der wir gemeinsam unsere

"Hausaufgabe" vom letzten Mal überprüfen werden. Sie erinnern sich bestimmt an unser kleines Anwender-Programm zum Schreiben einer Rechnung. Gleich auf Seite 53 nachlesen!

TANK BATTLE

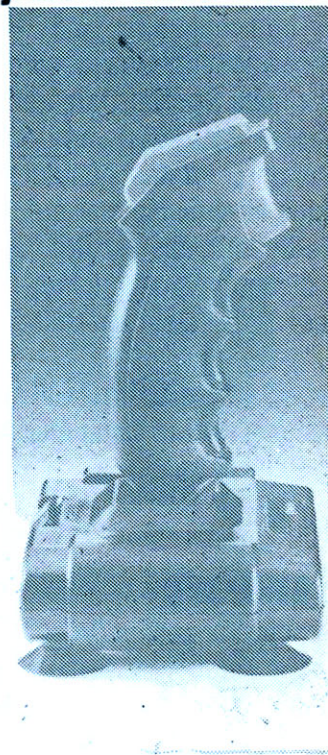
Das Bildschirm-Ereignis, das alle Video-Hits Kopf stehen läßt, erwartet Sie in Ihrer neuesten "Compute mit..."

Ob Konventionell bei Tage und durch Hindernisse getarnt, ob als "Nightbattle" im Finstern (man sieht nicht die Hand vor Augen, geschweige den eigenen oder gegneri-

schen Panzer), ob Mann gegen Mann oder Mensch gegen Maschine (Sonderausführung "Robotbattle") – dieses Programm läßt niemanden kalt, der für Taktik und packende Nahkampf-Situation etwas übrig hat. Also – Helm auf, Knobelbecher an und abmarsch auf Seite 37.

Quickshot-Joysticks jetzt in einer Hand

Nachdem in der Vergangenheit verschiedene Importeure das Recht für sich in Anspruch nahmen, die überaus erfolgreichen Quickshot-Joystick-Modelle SVI-101 und SVI-102 in der Bundesrepublik Deutschland vertreiben zu dürfen, hat es jetzt der Generalimporteur der SVI-Computermodelle SVI-318 und SVI-328 nach eigener Aussage geschafft, das Alleinvertragsrecht für die Bundesrepublik Deutschland auch für die Quickshot-Joysticks zu erhalten. Dieses wurde anlässlich der HiFiVideo 84 in Düsseldorf vom SVI-Generalimporteur der Firma B. Jöllenbeck, während der Präsentation des 2.500.000sten Quickshot-Joysticks SVI-101 mitgeteilt.



— Assembler-Kurs Teil 3 auf 5 1/2 Seiten! — Berichte — News —
Tips & Tricks — 11 Superprogramme für Commodore 64 und VC-20 — Basic-Kurs (Teil 3)
— CM-Lexikon — Rätsel —

Soft-Service 3

Basic-Kurs (Teil 2) 4

Telex

**Zum 20igsten Male:
Jugend forscht 1985**

Ordnung im Computer Zimmer
durch den Heimcomputer-Schrank
von C.P.V. 5

Computer im Koffer: Der SX 64

Mit dem SX 64 stellte Commodore
seinen ersten Portablen
vor. Die rein äußerlichen
Merkmale sind Grundlage
dieses Tests

Spezial-Datenrecorder für Microcomputer: Boston Computer MC 3810

Die meisten der derzeit
angebotenen Cassettenrecorder
für Microcomputer sind
abgewandelte Musikrecorder.
Der neue Spezial-Datenrecorder
System MC 3810 von Boston
Computer ist dagegen durch
seine Funktionen klar auf
die Bedürfnisse der Daten-
speicherung ausgerichtet.
Lesen Sie selbst auf Seite 6

Kampf gegen Software-Piraterie: Bundesverband-Computer Software gegründet 7

Software

Schiffe-Versenken (VC-20)	8
Tropic Trouble (VC-20)	18
Texas (VC-20)	23
Tankbattle (VC-20)	37
ABM (VC-20)	43
Barriere (C-64)	16
Graphik (C-64)	21
Cave of Death (C-64)	31
Backgammon (C-64)	40

Acey Deucey (C-64)	40
ID-Changer (C-64)	46

Testbericht

Interpod:

Ein Interface von OCS
als Schlüssel zu vielen neuen
Möglichkeiten 34

Tips & Tricks für den Commodore 64 und den VC-20, die bestimmt auch Ihnen helfen werden. 35

CM-Lexikon

Ab dieser Ausgabe von
"Compute mit"/39
erklären wir in alphabetischer
Reihenfolge die wichtigsten
Begriffe aus der
Computersprache 47

Software-Reviews KFC-Super

Toolkit und Monitor im
Eprom für den C-64 48

Nachruf auf Olympia

Der C-64 spielt sogar die
Nationalhymnen... 49

C-Club 52

Werkstatt

Logik im Basic-
programm (Teil 3) 53

Assembler-Kurs (Teil 3) 56

Kleinanzeigen 62

Rätselserie 63



Impressum

Compute mit
erscheint wöchentlich im:
Roeske Verlag, Eschwege

Herausgeber:
Ralph Roeske

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

Freie Mitarbeiter:
H. Schaden, M. Kostopoulos, O. Hilgendorf,
A. Gerstle, J. Ernst, J. Battles, R. Pahlmeyer,
J. Stumpf, H. Oldermann, H.-J. Haller, M. Rüdiger

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 Buchhandels),
sowie Österreich und Schweiz:
Verlagsunion
Friedrich-Bergius-Straße 7
6200 Wiesbaden
Tel.: 06121/2660

**Anfragen nicht an Vertrieb oder Druckerel, son-
dern 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 Über-
setzungen, vorbehalten.
Reproduktionen jeder Art (Fotokopie, Microfilm,
Erfassung in Datenverarbeitungsanlagen, usw.) be-
dürfen der schriftlichen Genehmigung des Ver-
lags. Alle veröffentlichte Software wurde von Mit-
arbeitern des Verlages oder von freien Mitar-
beitern erstellt.
Aus ihrer Veröffentlichung kann nicht geschlos-
sen 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, ein-
verstanden sind.

Bei Zusendung von Manuskripten und Software,
erteilt der Autor dem Verlag die Genehmigung zum
Abdruck und Versand der veröffentlichten Pro-
gramme auf Datenträger.
Rücksendung erfolgt nur gegen Erstattung der
Unkosten. Zusendungen von Software zur Veröf-
fentlichung soll bitte folgendes enthalten:
Kopierfähige Kassette oder Diskette mit dem Pro-
gramm (Computer-Bezeichnung), von Drucker er-
stelltes Listing oder Serie von Bildschirmfotos
(keine Schreibmaschinenlistings), evtl. Bild-
schirmfotos von einem Probelauf, ausführliche
Programmbeschreibung, (Erklärung der Vari-
ablenliste, Beschreibung des Bildschirmaufbaues,
Farbe, Grafik usw.)
Für eingesandte Programmunterlagen kann kei-
nerlei Haftung übernommen werden.

DatEx 64

DatEx 64 ist eine universelle Daten- und Textverarbeitung, vorwiegend für den Privatanwender, mit überdurchschnittlichen Leistungen. Sie können mit **DatEx 64** Ihre Videocassetten, Briefmarken, Rezepte, Schallplatten, Bücher etc. verwalten und Texte bearbeiten, mit deutschen Sonderzeichen. Durch einfache Anwendbarkeit ist die Arbeit mit **DatEx 64** auch für den Anfänger schnell zu erlernen. Die wichtigsten Eigenschaften in Stichworten:

Datensätze frei erstellbar (von einem Zeichen bis zu einer DIN-A4-Seite in der selben Datei) – verschiedene Suchoptionen – Zugriff über jeden Eintrag (Datenbankähnliche Funktion) – Erweiterungsmöglichkeiten – Textbreite bis 80 Zeichen (Scrolling) – Übernahme des letzten Wortes (kein Trennen notwendig) – nachträgliches Einfügen – **deutsche Sonderzeichen auf den Druckern: MPS 801, VC 1525, Seikosha GP 100 VC** – Bildschirmfarben jederzeit änderbar – Daten- und Textverarbeitung wird nur einmal geladen – Plausibilitätskontrollen – deutsches Handbuch – nur Diskettenversion.

DM 69,90

Disketten 64

Disketten 64 braucht jeder Anwender, der mehr als fünfzig Programme oder Dateien auf Diskette besitzt. Mit **Disketten 64** hat die Sucherei nach Programmen ein Ende und außerdem sind Sie jederzeit über Ihren Dateien-Programmbestand informiert. Der Vorteil von **Disketten 64** besteht eigentlich darin, daß die Eingaben nicht über die Tastatur erfolgen, sondern von **Disketten 64** selbständig eingelesen werden.

Die wichtigsten Eigenschaften in Stichworten: Erfassung nach max. 40 verschiedenen Gebieten – Sortieren nach Programmnamen (alphabetisch) – Folgende Informationen werden von **Disketten 64** eingelesen und ausgegeben: Länge des Programmes (Blocks), Anzahl der Teile des Programmes, Name der Diskette, fortlaufende Programmnummer – zusätzlich kann ein kurzer Kommentar eingegeben werden – deutsches Programm.

DM 69,90

AdrEx 64

AdrEx 64 ist ein menügesteuertes Adressenverwaltungsprogramm.

AdrEx 64 zeichnet sich durch eine hohe Anwenderfreundlichkeit aus. Es können 450 Adressen pro Datei verwaltet werden. Die wichtigsten Eigenschaften in Stichworten: 450 Adressen pro Datei (Cassettenversion 150) – zusätzlich 3 Textzeilen – Direktzugriff auf alle Kriterien einer Adresse, auch Textzeilen – komfortable Suchmöglichkeiten – löschen, ändern, ausdrucken einer Adresse – Listen und Etiketten drucken – Absicherung gegen Fehlbedienungen – Bildschirmfarben jederzeit änderbar – deutsches Programm.

Cassettenversion (150 Adressen)
Diskettenversion (450 Adressen)

**DM 69,-
DM 79,-**

Demnächst erscheint unser komfortabler "DISK-MONITOR" für den C-64.

BITTE INFO ANFORDERN.

Unser C-64 Katalog liegt für DM 2,- in Briefmarken bereit.

HÄNDLERANFRAGEN ERWÜNSCHT. Alle Programme für den Commodore 64. Computeranwendungen Seucan. Postf. 24 44, 8600 Bamberg 1, Tel.: 0 95 42/83 48

BESTELLCOUPON

Bitte senden Sie mir _____

☐ per Nachnahme (zzgl. DM 5,50) ☐ per Vorauszahlung (zzgl. DM 3,50)

Auslandsbestellungen nur gegen Vorkasse möglich (zzgl. DM 7,-)

Name: _____ Adresse: _____

PLZ: _____ Ort: _____ Unterschrift: _____

Bitte einsenden an: Computeranwendungen Seucan, Postfach 24 44, 8600 Bamberg 1.

soft service

Unser Softwareversand bietet Ihnen zu wirklich einmalig günstigen Preisen jeden Monat 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 Kreissparkasse Eschwege, Bankleitzahl 522 500 30 Kto.-Nr. 45 22 934

senden wir Ihnen die gewünschten Programme schnellstmöglich zu. Geben Sie bitte unbedingt die genaue Bezeichnung der Kassette (z.B. Compute mit 37/

84 C-64) und den Computer an.

Bestellungen Ausland:

Nur Vorauskasse, Schein
Keine Schecks oder Überweisungen!

Commodore 64 K 14,-DM

Cave of Death
ID-Changer
Backgammon und
Acey Deucey
Graphik
Barriere

VC-20

ABM
Schiffe-Versenken
Tankbattle
Tropic Trouble
Texas

K 14,-DM

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

Basic-Kurs

Teil 2

Diese Woche: Einführung in die Variablen-Arithmetik.

Zunächst wollen wir zur Auflösung der Aufgaben der letzten Woche kommen.

Die erste Frage lautete:
Wo liegt der Fehler in der folgenden Zeile:

```
PRINT 'COMPUTE MIT  
COMMODORE'
```

Nun, die Lösung war relativ einfach. Das Zeichen ' war falsch. Korrekt muß es " sein. Die Gründe dafür werden wir in dieser Ausgabe noch genau erläutern.

Die zweite Frage:
Kann der folgende Befehl funktionieren:

?"Werner was here"

hatte den erstaunlichen Effekt, daß das Geschriebene tatsächlich auf dem Bildschirm erschien. Das liegt daran, daß der Commodore für fast jeden BASIC-Befehl eine Abkürzung kennt. Beim PRINT-Befehl ist es das Fra-

gezeichen. Am Ende der Lektion steht eine Tabelle, in der Sie die Abkürzungen für die anderen Befehle nachlesen können.

Doch nun zum eigentlichen Basic-Kurs.

Wir wollen diese Woche die Variablen behandeln. Eine Variable ist grundsätzlich erst einmal ein veränderlicher Wert – im Gegensatz zur Konstanten, die immer ihren festen Wert behält. Bei der Programmiersprache Basic werden solche Variablen auch als 'STRINGS' bezeichnet, und ein String stellt eine alphanumerische Zeichenkette dar, deren Länge von Null bis 255 Zeichen zu variieren ist. Innerhalb des Basic-Programmes wird festgelegt, wieviel Stellen eine Variable haben kann und darf, sodaß der Speicherplatz einigermaßen vernünftig ausgefüllt wird. Es gibt jedoch auch noch Zahl-Variablen mit einfacher und doppelter Genauigkeit. Eine solche Variable haben

wir in der vorigen Lektion schon einmal, ohne es zu wissen, abgefragt und zwar, als wir den Computer überreden wollten HALLO zu schreiben und nur eine 0 herauskam, da der Variablen noch kein Wert zugewiesen war. Hätten wir vorher eingegeben:

```
HALLO=8888  
so hätte der Computer:  
8888
```

```
READY  
geschrieben.
```

HALLO hätte in dem Falle den Wert 8888 angenommen.

Ganz klar wird das einem, wenn man es so ausspricht, wie man es geschrieben hat: HALLO ist gleich 8888

Das Gleichheitszeichen dient also dazu, einen Buchstaben (eine Intervariable) einen Wert zuzuweisen.

Mit dem PRINT-Befehl kann man sich den Inhalt der Variablen sichtbar machen.

Variablen können, da sie numerische Werte halten, auch wie Zahlen verwaltet werden.

Geben Sie nun ein:

```
A=3
```

```
B=9
```

```
PRINTA+B
```

Nach der letzten Anweisung wird der Computer

```
12
```

```
READY
```

auf dem Bildschirm schreiben.

Sollten wir jetzt

```
PRINT A oder PRINT 8
```

eingeben, so würde jeweils der Wert erscheinen, den wir der entsprechenden Variablen angewiesen haben.

Aufgabe zur nächsten Lektion:

Vervollständigen Sie das nächste Programm so, daß der Computer zum Schluß in der Variablen Y den Wert 55 hält:

```
A=66
```

```
B=3
```

```
X=B+C
```

```
Y=A-X
```

```
PRINT Y
```

(tm)

Damit ist die Einführung in die Strukturen der wichtigsten Grundregeln schon so gut wie abgeschlossen. Der erste Teil dieses sehr wichtigen Bereiches der Programmierung mag vielleicht ein wenig langatmig gewesen sein.

Sie werden aber bemerken, daß es ohne die ersten beiden Lektionen nicht geht, zumal das Thema im Handbuch nur am Rande behandelt wird.

Im nächsten Heft werden wir mit dem ersten Erstellen eines BASIC-Algorithmus beginnen und unser erstes richtiges Programm schreiben.

Also versäumen Sie nicht den dritten Teil unserer Serie, denn hier beginnen die Übungen, die einen BASIC-Programmierer aus Ihnen machen.

Start für die 20. Runde JUGEND FORSCHT 1985

Sei kein Frosch, mach' mit

Daß der Frosch kein Wetterprophet ist, hat die Forschung inzwischen bewiesen. Und so hüpfert er denn davon, unser Plakattfrosch. Aber in den Naturwissenschaften sind noch viele Fragen offen, auf die eine Antwort erst gefunden werden muß. Täglich kommen neue Fragestellungen hinzu.

Sein kein Frosch, mach' mit heißt darum auch das Motto beim 20. Wettbewerb JUGEND FORSCHT. Wer Lust hat, naturwissenschaftliche Zusammenhänge zu erkennen, seine Umwelt genauer unter die Lupe zu nehmen, wer nicht blind glaubt, was man ihm erzählt, wer selbst experimentieren und eigene Ergebnisse erhalten möchte, der ist bei JUGEND FORSCHT an der richtigen Adresse. Gefragt sind Einfallsreichtum und Originalität.

Bei JUGEND FORSCHT können Mädchen und Jungen bis zum Alter von 22 Jahren mitmachen, ganz gleich, ob Schülerin oder Student (bis zum 1. Semester), Auszubil-

dender oder junge Angestellte, Bundeswehrangehöriger oder Zivildienstleistender. Wer noch keine 16 Jahre alt ist, startet in der Wettbewerbssparte SCHÜLER EXPERIMENTIEREN. Geforscht werden kann allein oder in einer Gruppe bis zu drei Teilnehmern.

Beim Wettbewerb JUGEND FORSCHT darf das Thema frei gewählt werden, es muß nur in die Fachgebiete Biologie, Chemie, Geo- und Raumwissenschaften, Mathematik/Informatik, Physik, Technik oder zum Sonderpreisthema Arbeitswelt passen. Zusätzlich ausgezeichnet werden Arbeiten, die im Rahmen dieser Fachgebiete einen Beitrag zur Energieeinsparung und zur Verbesserung unsrer Lebensbedingungen leisten; zum Beispiel in den Bereichen Arbeitswelt und Umweltschutz.

Anmeldeschluß für die 20. Wettbewerbsrunde ist der 30. November 1984. Teilnahmebedingungen und Informationsmaterial gibt es kostenlos bei: Stiftung JUGEND



FORSCHT e.V., Notkestraße 31, 2000 Hamburg 52. Zu gewinnen sind Geldpreise, Studienaufenthalte und -reisen im Werte von DM 100.000,-.

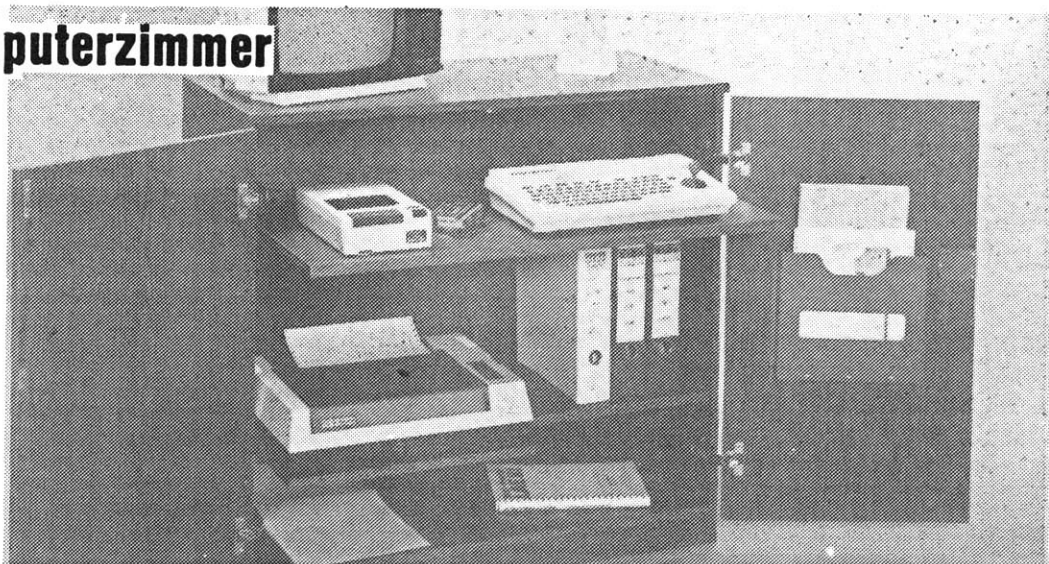
Mitmachen und Spaß haben, gleichgesinnte Freunde finden, Diskussionen mit Experten und Anerkennung bekommen – das ist auch schon ein Gewinn.

Ordnung im Computerzimmer

Wer kennt das nicht? Überall dort wo Heimcomputer aufgebaut sind, wirrt es von Kabelsträngen und Zubehörteilen.

Der Homecomputer-Schrank von C.P.V. macht Schluß mit der Unordnung, in ihm findet sogar sämtliche Peripherie Platz.

Durch bequeme Auszugböden für Tastatur, Drucker, Floppy und Kassettenrecorder problemlos zu bedienen, Kabelstränge verschwinden ebenfalls.



Mit einer Größe von 90X87X45 cm und ansprechendem

Design läßt sich in jedem Zimmer ein Plätzchen für den

Home-Computer-Schrank finden.

Direkt darüber befindet sich das sehr praktische Disketten-ablagefach. Das große Manko des SX64 ist leider

Zu erwähnen ist noch der eingebaute Lautsprecher (Durchmesser 8cm), der bei Spielen und Musikprogrammen die akustische Untermalung



In seiner Grundversion kann der SX64 nicht netzunabhängig betrieben werden, da kein Akku vorhanden ist.

6

der Rückseite sowie ein aus der professionellen Datenverarbeitung bekanntes technisches Design.

Über die jeweiligen Interfacekabel ist der Datenrecorder System MC 3810 universell an Microcomputer von Commodore, Atari, Spectravideo, Acorn, Triumph Adler, Sinclair, Sanyo etc. anschließbar. Beim Wechsel des Computersystems kann der Anwender damit den Datenrecorder MC 3810 weiter einsetzen.

Hier die technischen Daten:
Der Datenrecorder MC 3810 ist über Interfacekabel mit

folgenden Microcomputertypen kompatibel:
Commodore 2000/3000/4000
Commodore VC-20/VC-64
Atari 600
Atari 800
Sinclair Spectrum
Sinclair ZX-81
Acorn B
Acorn Electron
Anyo Laser
Oric
Dragon 32
Dragon 64

Technische Features:
Präzisionslaufwerk für optimalen Gleichlauf, Motor Control (Computersteuerung), Pegelautomatik, Endabschalt-

automatik, Leuchtfelder für Funktionsanzeige READY, SAVE, LOAD, CONTROL sowie LED Anzeige für Datentransfer.
Sämtliche Anschlüsse für Interfacekabel sowie Stromversorgung an der Rückseite des Gerätes,
Monitorbuchse auf Bedienungspanel, ölgedämpfter Datenkassettenauswurf, Bandzählwerk.

Datenübertragungsrate: 1200 Baud
Bandgleichlauf: +/- 0,3%
Bandgeschwindigkeit: 4,75 cm/sek.
Stromversorgung: Externes

Netzteil zur Wärmeableitung 220 Volt (nicht notwendig bei Commodore VC-20/64 und Acorn B)

Gehäuse: Farbe Greige im Proflook

Masse: 25 x 14 x 12 cm

Gewicht: 1,2 kg.

Garantie: 6 Monate

Angeboten wird der Datenrecorder MC 3810 zum empfohlenen Endverbraucherpreis von DM 98,-. Die Preise für die jeweiligen Interfacekabel liegen zwischen DM 8,- und DM 38,-. Nachweis über Boston Computer Handelsges.m.b.H., in München.

Kampf gegen Software-Piraterie

Bundesverband Computer Software gegründet

Rechtzeitig vor Beginn der HiFiVideo 1984 konstituierte sich am 21.08. in Düsseldorf der Bundesverband Computersoftware, die Vereinigung der Computersoftware-Hersteller Deutschlands.

Vordringlichste Aufgabe des Verbandes wird die Aufnahme der organisierten Bekämpfung der Computer-Programmpiraterie sein. Bekanntlich haben Raubkopierer der noch jungen Industrie in den vergangenen Jahren weltweit Schaden in Milliardenhöhe zugefügt.

Die Grundlage für die Bekämpfung wird der Bundesverband Computer Software durch Anregung für eine entsprechende Rechtsgrundlage schaffen. Dazu gehört die Klärung der noch immer offenen Rechtsfragen bezüglich des Computer-Programm-Urheberrechtsschutzes.

Neben der Wahrung und Sicherung der in der Satzung verankerten gemeinsamen In-

teressen, sieht der Verband eine wichtige Aufgabe in der Entmystifizierung des Computers und wird entsprechende Öffentlichkeitsarbeit leisten, um den Stellenwert des Computers in der Gesellschaft zu verdeutlichen. Die Förderung und Entwicklung neuer Software-Technologien ist weiteres Anliegen der Software-Hersteller.

Die zwölf Gründungsmitglieder (AriolaSoft, Atari Elektronik, Commodore Büromaschinen GmbH, DataMedia, Hansesoft, Markt + Technik, Roeske Verlag, Pecher Software, Thorn EMI, VideoMagic, Strecker Elektronik sowie der Westermann Schulbuchverlag) wählten Harald Speyer (44), Vorsitzender der Ge-

schäftsführung der Commodore Büromaschinen GmbH und Vice President der Commodore International Ltd., vertreten durch Heinz-Joachim Kröber (36) zum 1. Vorsitzenden des Verbandes. Peter Knippel (33), designerter Geschäftsführer der AriolaSoft GmbH München wurde zum 2. Vorsitzenden gewählt.

Neben den Firmen, die spontan die Bereitschaft zur Gründung des Verbandes erklärten, waren alle namhaften Anbieter in Düsseldorf vertreten. Die Häuser Apple, Epson, Hewlett Packard, Acorn Computer GmbH, Digital Equipment, Microsoft und der Otto Maier Verlag Ravensburg, sowie die Vielzahl junger mittelständischer Anbieter werden dem Verband

voraussichtlich nach entsprechender Prüfung durch die Geschäftsleitungen ebenfalls beitreten.

Sitz des Verbandes i. Gr. ist Hamburg. Die im Bundesverband Computersoftware vertretenen Firmen beschlossen, im Anschluß an die HiFiVideo 1984 Gespräche mit dem Bundesverband Video zur Koordinierung der Aktivitäten gegen Tendenzen einer nicht notwendigen Verschärfung des Jugendschutzes aufzunehmen, in der besonders die Hersteller und Importeure von Game-Software eine Gefahr sehen. Eine enge Zusammenarbeit mit der IFBI, die bei der Gründungsversammlung als Gast zugegen war, ist darüber hinaus vorgesehen.



Schiffe-Versenken

für den VC-20 + 24K

Das Spiel Schiffe-Versenken besteht aus drei Programmen, die sich selbst nachladen. Das erste Programm begrenzt den für Basic verfügbaren Speicherplatz, um neue Zeichen aufnehmen zu können. Das Zweite kopiert zuerst den Zeichensatz vom ROM ins freigewordene RAM und definiert dann die neuen Zeichen. Dann wird das Hauptprogramm geladen. Für das VC-20 Spiel gelten die bekannten Spielregeln. Schiffe dürfen nur horizontal oder vertikal gesetzt werden und sich nicht berühren. Die Anzahl der Schiffe kann man selbst festlegen:

- 1-3 Kreuzer (4 Kästchen)
- 1-5 Zerstörer (3 Kästchen)
- 1-5 U-Boote (2 Kästchen)

Beim Setzen der Schiffe fragt der Computer zuerst ab, ob das Schiff senkrecht oder waagrecht liegen soll und in welcher Spalte (A-T), bzw. Reihe (01-10). Danach wird der Anfang des Schiffes eingegeben. Sollte die Position des Schiffes nicht den Spielregeln entsprechen, so teilt der Computer es Ihnen mit. Gespielt wird auf dem 10x20 Kästchen großen Feld.

Der Schwierigkeitsgrad bestimmt, wieviel % der Computer-Versuche zufällig sind, und bei wieviel % systematisch ein

Schiff gesucht wird.

Sie beginnen das Spiel dann mit Eingabe Ihrer Zielkoordinaten, z.B.: A-05. Sollte sich dort ein Schiff befinden, so zeigt der Computer Treffer an und markiert die Stelle durch ein ausgefülltes Kästchen. Dann haben Sie so lange einen weiteren Versuch, bis ein Schuß daneben geht. Sollte dies der Fall sein, so wird es durch einen Punkt im Koordinatensystem dargestellt, und der Computer ist an der Reihe. Er zeigt seinen Versuch in der unteren Bildschirmhälfte in einem eigenen Koordinatensystem an, in dem Ihre Schiffe zu sehen sind. Hat er Ihr Schiff getroffen, so fragt er systematisch weiter, bis es versenkt ist.

Sieger ist, wer zuerst alle gegnerischen Schiffe versenkt hat.

Mit f1 können Sie das Spiel abbrechen, wenn Sie an der Reihe sind. Wenn Sie das Spiel verloren haben, oder mit f1 abbrechen, zeigt der Computer Ihnen seine Schiffe. Danach sagt er Ihnen, wieviel Einheiten Sie und er getroffen haben und den prozentualen Anteil an der Gesamtschiffszahl.

Bei Schwierigkeitsgrad 0 fragt der Computer zwar nach einem bestimmten System aber auch, wie Sie, rein zufällig nach Schiffen ab. Bei Stufe 9 wird jedes Schiff mit ein oder zwei Schüssen getroffen.

Variablenbelegung:

T1-T4: Tongeneratoren

LA: Lautstärke

BS: Positionspoke

FA: Farbpoke

K1,Z1,U1: Anzahl der Schiffe

P%(x): Positionen der Spielerschiffe

C%(x): Positionen der Computerschiffe

Programmaufbau:

1000-2000: Variablenbelegung

2000-3000: Titelbild

3000-4000: Spielanleitung

4000-5000: Eingabe der Schiffszahlen

5000-6000: Eingabe der Kreuzer

6000-7000: Eingabe der Zerstörer

7000-8000: Eingabe der U-Boote

8000-9000: Computer-Schiffe

9000-10000: Spielfeldaufbau

10000-11000: Spielereingabe und Auswertung

11000-12000: Computerzug und Auswertung

12000-13000: Spielabbruch bzw. Spielende

13000-14000: Spieler-Sieg

14000-15000: Computer-Sieg

30000-31000: Sound und andere Routinen

```
10 PRINT "PF44,28:PF7168,0:NEW":POKE36879,25
20 POKE631,19:POKE632,13:POKE633,131:POKE198,3
30 PRINT "Schiffe-Versenken - LÖSUNGSLADUNG"
40 END
READY.
```



```

10 PRINT "BITTE LASSEN SIE DIE 'PLAY' TASTE GEDRUECKT":POKE36879,8
20 FOR I=1 TO 3E3:NEXT
30 FOR I=0 TO 255:POKE4096+I,I:I:NEXT:POKE36869,205
40 FOR I=0 TO 2047:POKE5120+I,PEEK(32768+I):NEXT
100 FOR I=5848 TO 6039:READ A%:POKE I,A%:B=B+A%:NEXT
110 IF B<>26213 THEN PRINT "Fehler":END
120 PRINT " ":POKE631,131:POKE198,1:END
1000 DATA 255,127,63,31,15,7,3,1
1010 DATA 255,255,255,255,255,255,243,192
1020 DATA 255,255,255,255,255,255,255,199
1030 DATA 255,255,255,255,255,255,255,255
1040 DATA 255,255,255,255,255,255,224,27
1050 DATA 0,0,0,0,0,0,0,0
1055 DATA 224,224,240,240,240,240,224,192
1060 DATA 7,7,7,7,255,255,255,255
1070 DATA 255,255,51,255,255,255,255,255
1080 DATA 128,128,128,128,128,128,255,255,255
1100 DATA 7,15,63,96,64,128,192,192
1110 DATA 126,126,126,126,126,126,126,126
1120 DATA 0,255,255,255,255,255,255,0
1130 DATA 126,126,126,126,126,126,60,60,24
1140 DATA 24,60,60,126,126,126,126,126
1150 DATA 0,243,254,255,255,254,248,0
1160 DATA 0,31,127,255,255,127,31,0
1170 DATA 128,112,120,124,62,31,14,4
1180 DATA 133,69,52,64,148,34,72,136
1190 DATA 160,80,168,120,170,85,170,85
1200 DATA 0,64,224,64,64,64,64,64
1210 DATA 0,16,16,16,16,255,126,60
1220 DATA 0,0,0,16,16,0,0,0
1230 DATA 255,128,128,128,136,128,128,128
READY,

1000 REM DEFINIEREN
1010 T1=36874:T2=36875:T3=36876:T4=36877:LA=36878
1020 BS=4096:FA=37888
1030 S$="          "
1040 L$="          "
1050 TR=108:E$="          "
1060 S1$="1":S2$="1":S3$="1":S4$="1":S5$="1":S6$="1"
1070 POKE LA,15:POKE LA+1,8:POKET1=5,205:POKE650,0:POKE657,128
1080 POKE36865,38:POKE36864,12
2000 REM TITEL
2010 PRINT "SCHIFFE VERSENKEN"
2020 PRINT " "
2030 FOR I=16 TO 1 STEP -1:PRINT TAB(I)S$;" ";
2035 FORK=1 TO 40:NEXT
2040 FORK=128 TO 138 STEP .7:POKET1,K:NEXT:FORK=138 TO 128 STEP -.7:POKET1,K:NEXT
2050 PRINT L$;" ":IF I=16 THEN PRINT " "
2060 NEXT I
2070 PRINT S$:FOR I=15 TO 0 STEP -1:POKE LA,I:NEXT
2080 POCKET1,0:POKE LA,15
2100 FOR I=9 TO 1 STEP -1:POKEBS+22*(I+1)+I,TR:POKEBS+22*(I+1+1)+I+1,32
2110 POCKET4,250-(9-I)
2120 FORK=1 TO 10:NEXT K,I
2150 PRINT " "
2200 PRINTE$
2210 POCKET4,128:FOR I=15 TO 4 STEP -.01:POKE LA,I:NEXT:POCKET4,0:POKE LA,15
2220 PRINT " "
2230 FORK=10 TO 12:FOR I=0 TO 5:POKEBS+22*K+I,32:NEXT I:PRINT:NEXT K
2300 IF E THEN PRINT TAB(9)" " :END
2500 PRINT "DABEI BY:" :PRINT "OLIVER HILGENDORF"
2600 FOR I=1 TO 3E3:NEXT
3000 REM ANLEITUNG
3001 POKE LA+1,25:PRINT "SCHWIERIGKEITSGRAD 1-9?"
3002 POKE198,0:WAIT 198,1:GETA$
3003 IF A$="1" OR A$="9" THEN GOSUB 30000:GOTO 30002
3004 SW=VAL(A$)/10-.1:PRINT "SPIELANLEITUNG J/N"
3005 POKE198,0:WAIT 198,1:GETA$
3006 IF A$("<)J"&AND A$("<)N" THEN GOSUB 30000:A$="":GOTO 3005
3007 IF A$="N" THEN 4000
3008 PRINT "ES IST IHRE AUFGABE, ALLE SCHIFFE DES GEGNERS ZU VERNICHTEN."
3010 PRINT "DOCH GLEICHZEITIG VERSUCHT ER DASSELBE MIT IHREN SCHIFFEN."
3015 GOSUB 30100
3020 PRINT "SIE BESITZEN 2 ANZEIGEN. DIE ERSTE ZEIGT IHNEN, AN WELCHEN"
3030 PRINT "STELLEN SIE SCHON NACH DEM GEGNER GEORDET HABEN, UND OB SIE"
3040 PRINT "DABEI SEINE SCHIFFE GETROFFEN ODER VERNICHTET HABEN."
3050 GOSUB 30100
3060 PRINT "DIE ZWEITE ANZEIGE ZEIGT IHNEN, AN"
3070 PRINT "WELCHEN STELLEN DER GEGNER GEORDET HAT, UND OB DABEI IHRE"
3080 PRINT "SCHIFFE GETROFFEN WORDEN SIND."
3090 GOSUB 30100
3100 PRINT "AM BEGINN DES SPIELS SETZTEN SIE IHRE SCHIFFE."
3110 PRINT "DIE SCHIFFE DUERFEN NUR HORIZONTAL ODER VERTIKAL ANGEORDNET"

```

```

3120 PRINT "WERDEN. ZWISCHEN ZWEI SCHIFFEN MUSS IMMER EIN KASTCHEN FREI"
3130 PRINT "GEIN."
3140 GOSUB 30100
3150 PRINT "FUEER DEN GEGNER GILT NATUERLICH DASSELBE."
3160 PRINT "DIE ZIELKOORDINATEN WERDEN IMMER 3-STELLIG EINGEGEBEN. ZB: A5=A05"
"
3170 PRINT "ZUERST KOMMT DER BUCHSTABE DANN DIE ZAHL."
3180 PRINT "MIT F1 KOENNEN SIE DAS SPIEL ABRECHEN."
3200 GOSUB 30100
4000 REM ABFRAGE
4010 PRINT "GEBEN SIE DIE ANZAHL DER SCHIFFE EIN:"
4020 PRINT "DIE LAENGE STEHT IN ()"
4030 INPUT "KREUZER (4) 1-3"; K$
4040 IF VAL(K$) > 3 OR VAL(K$) < 1 THEN GOSUB 30000: K$ = "": GOTO 4030
4050 INPUT "ZERSTOERER (3) 1-5"; Z$
4060 IF VAL(Z$) > 5 OR VAL(Z$) < 1 THEN GOSUB 30000: Z$ = "": GOTO 4050
4070 INPUT "U-BOOTE (2) 1-5"; U$
4080 IF VAL(U$) < 1 OR VAL(U$) > 5 THEN GOSUB 30000: U$ = "": GOTO 4070
4500 DIM K$(VAL(K$), 4, 2), Z$(VAL(Z$), 3, 2), U$(VAL(U$), 2, 2)
4520 DIM P%(221), C%(221)
5000 REM SPIELER-SCHIFFE
5010 PRINT "GEBEN SIE EIN:"
5020 PRINT "1. SOLL DAS SCHIFF HORIZONTAL ODER VERTIKAL LIEGEN?"
5030 PRINT "2. IN WELCHER REIHE BZW. SPALTE SOLL ES LIEGEN?"
5040 PRINT "3. WO SOLL ES BEGINNEN?"
5050 GOSUB 30100
5100 PRINT "KREUZER"; FOR I = 65 TO 64: PRINT CHR$(I);: NEXT
5110 FOR I = 1 TO 9: PRINT I;: NEXT: PRINT "10"
5120 FOR I = 2 TO 21: FOR K = 4 TO 13
5130 POKEBS + 22 * K + I, 79: POKEFA + 22 * K + I, 6
5140 NEXT K, I
5200 K1 = VAL(K$): IF K1 = 0 THEN 6000
5210 K2 = K2 + 1: PRINT "2. K2 11. KREUZER (4)"
5220 PRINT "HORIZONTAL ODER VERTIKAL": FOR K = 1 TO 200: NEXT K
5225 PRINT "HORIZONTAL ODER VERTIKAL": FOR K = 1 TO 200: NEXT K
5230 GETP$: IF P$(K) = "H" AND P$(K) = "V" THEN 5220
5240 GOSUB 30050: IF P$ = "V" THEN 5600
5250 PRINT "HORIZONTAL": PRINT "REIHE (1-10) ? 2 11 12 13 14 15 16 17 18 19 20";
5255 FOR L = 1 TO 2
5260 POKE 198, 0: WAIT 198, 1: GETP$
5270 IF L = 1 THEN IF P$(K) = "0" AND P$(K) = "1" THEN GOSUB 30000: GOTO 5260
5280 IF L = 2 AND P$(K) = "1" THEN IF P$(K) = "0" THEN GOSUB 30000: GOTO 5260
5285 IF L = 2 AND P$(K) = "0" THEN IF P$(K) = "1" OR P$(K) = "3" THEN GOSUB 30000: GOTO 5260
5290 GOSUB 30050: PRINT "3. WO SOLL ES BEGINNEN? R1 R2 R3 R4 R5 R6 R7 R8 R9 R10";
5300 PRINT "1. SPALTE (A-Q) ? 2 11 12 13 14 15 16 17 18 19 20";
5310 POKE 198, 0: WAIT 198, 1: GETP$
5320 IF P$(K) = "A" OR P$(K) = "Q" THEN GOSUB 30000: GOTO 5310
5330 PRINT "3. WO SOLL ES BEGINNEN? R1 R2 R3 R4 R5 R6 R7 R8 R9 R10";
5340 FOR I = 1 TO 4: K(K2, I, 1) = VAL(R1$): K(K2, I, 2) = ASC(S$) - 65 + 1: NEXT
5350 IF K2 < 2 THEN 5400
5360 FOR K = 1 TO 4: FOR I = 1 TO K2 - 1: FOR L = 1 TO 4
5370 IF K(K2, K, 1) < K(K2, I, 1) OR K(K2, K, 2) < K(K2, I, 2) THEN NEXT L, I, K: GOTO 5400
5380 GOSUB 30300
5390 K2 = K2 - 1: R1$ = "": GOTO 5210
5400 A = K(K2, 1, 2) - 1: IF A = 0 THEN A = 1
5410 E = K(K2, 4, 2) + 1: IF E = 21 THEN E = 20
5420 O = K(K2, 1, 1) - 1: IF O = 0 THEN O = 1
5430 U = K(K2, 1, 1) + 1: IF U = 11 THEN U = 10
5440 FOR I = 0 TO U: FOR K = A TO E
5450 IF P%(20 * (I - 1) + K) = 0 THEN NEXT K, I: GOTO 5500
5460 GOSUB 30400
5470 K2 = K2 - 1: R1$ = "": GOTO 5210
5500 R1$ = " "
5510 FOR I = 1 TO 4: P%(20 * (K(K2, 1, 1) - 1) + K(K2, 1, 2)) = K2: NEXT
5530 FOR I = 1 TO 4: POKEBS + 22 * (K(K2, 1, 1) + 3) + K(K2, 1, 2) + 1, 224: NEXT
5540 IF K2 < 1 THEN GOSUB 30500: GOTO 5210
5550 GOTO 6000
5600 PRINT "VERTIKAL";
5610 PRINT "3. SPALTE (A-T) ? 2 11 12 13 14 15 16 17 18 19 20";
5620 POKE 198, 0: WAIT 198, 1: GETP$
5630 IF P$(K) = "A" OR P$(K) = "T" THEN GOSUB 30000: GOTO 5620
5640 GOSUB 30050: PRINT "3. WO SOLL ES BEGINNEN? R1 R2 R3 R4 R5 R6 R7 R8 R9 R10";
5650 POKE 198, 0: WAIT 198, 1: GETP$
5660 IF P$(K) = "1" OR P$(K) = "7" THEN GOSUB 30000: GOTO 5650
5665 PRINT "3. WO SOLL ES BEGINNEN? R1 R2 R3 R4 R5 R6 R7 R8 R9 R10";
5670 FOR I = 1 TO 4: K(K2, I, 2) = ASC(S$) - 64: K(K2, I, 1) = VAL(R$) + 1 - 1: NEXT
5680 IF K2 < 2 THEN 5800
5690 O = K(K2, 1, 1) - 1: IF O = 0 THEN O = 1
5700 U = K(K2, 4, 1) + 1: IF U = 11 THEN U = 10
5710 A = K(K2, 1, 2) - 1: IF A = 0 THEN A = 1
5720 E = K(K2, 1, 2) + 1: IF E = 21 THEN E = 20
5730 FOR I = 0 TO U: FOR K = A TO E
5740 IF P%(20 * (I - 1) + K) = 0 THEN NEXT K, I: GOTO 5800
5750 GOSUB 30400
5760 K2 = K2 - 1: R1$ = "": GOTO 5210
5800 R1$ = " "
5810 FOR I = 1 TO 4: P%(20 * (K(K2, 1, 1) - 1) + K(K2, 1, 2)) = K2: NEXT

```



```

5820 FOR I=1 TO 4:POKEBS+22*(K(K2,I,1)+3)+K(K2,I,2)+1,224:NEXT
5830 IF K2<K1 THEN GOSUB 30500:GOTO 5210
6000 GOSUB 30500
6010 Z1=VAL(Z$):IF Z1=0 THEN 7000
6020 Z2=Z2+1:PRINT "Z2" II. ZERSTOERER (3)"
6030 PRINT "XXXXXXXXXXXXXXXXXXXX? HORIZONTAL, VERTICAL":FOR K=1 TO 200:NEXT K
6040 PRINT "XXXXXXXXXXXXXXXXXXXX? V":FOR K=1 TO 200:NEXT K
6050 GETP$:IF P$<>"H" AND P$<>"V" THEN 6030
6060 GOSUB 30050:IF P$="V" THEN 6500
6070 PRINT "XXXXXXXXXXXXXXXXXXXX?":PRINT "REIHE (1-10) ? 1 2 3 4 5 6 7 8 9 10";
6080 FOR L=1 TO 2
6090 POKE 198,0:WAIT 198,1:GETR$
6100 IFL=1 THEN IF R$<>"0" AND R$<>"1" THEN GOSUB 30000:GOTO 6090
6110 IFL=2 AND R1$="1" THEN IF R$<>"0" THEN GOSUB 30000:GOTO 6090
6120 IFL=2 AND R1$="0" THEN IF R$<>"1" OR R$<>"9" THEN GOSUB 30000:GOTO 6090
6130 GOSUB 30050:PRINT "R1$";R1$=R1$+R$:NEXT L:PRINT
6140 PRINT "1. SPALTE (A-R) ? 1 2 3 4 5 6 7 8 9 10";
6150 POKE 198,0:WAIT 198,1:GETS$
6160 IF S$<>"A" OR S$<>"R" THEN GOSUB 30000:GOTO 6150
6170 PRINT "2. I1";S$:GOSUB 30050
6180 FOR I=1 TO 3:Z(Z2,I,1)=VAL(R1$):Z(Z2,I,2)=ASC(S$)-65+I:NEXT
6190 IF Z2<2 THEN 6300
6200 FOR K=1 TO 3:FOR I=1 TO Z2-1:FOR L=1 TO 3
6210 IF Z(Z2,K,1)<>Z(Z2,I,L,1) OR Z(Z2,K,2)<>Z(Z2,I,L,2) THEN NEXT L, I, K:GOTO 6300
6220 GOSUB 30300
6230 Z2=Z2-1:R1$="":GOTO 6020
6300 A=Z(Z2,1,2)-1:IFA=0 THEN A=1
6310 E=Z(Z2,3,2)+1:IFE=21 THEN E=20
6320 O=Z(Z2,1,1)-1:IFO=0 THEN O=1
6330 U=Z(Z2,1,1)+1:IFU=11 THEN U=10
6340 FOR I=0 TO U:FOR K=ATOG
6350 IF P$(20*(I-1)+K)=0 THEN NEXT K, I:GOTO 6400
6360 GOSUB 30400
6370 Z2=Z2-1:R1$="":GOTO 6020
6400 R1$=""
6410 FOR I=1 TO 3:P$(20*(Z(Z2,1,1)-1)+Z(Z2,1,2))=Z2+10:NEXT
6430 FOR I=1 TO 3:POKEBS+22*(Z(Z2,1,1)+3)+Z(Z2,1,2)+1,224:NEXT
6440 IF Z2<2 THEN GOSUB 30500:GOTO 6020
6450 GOTO 7000
6500 PRINT "XXXXXXXXXXXXXXXXXXXX? VERTICAL";
6510 PRINT "3. SPALTE (A-T) ? 1 2 3 4 5 6 7 8 9 10";
6520 POKE 198,0:WAIT 198,1:GETS$
6530 IF S$<>"A" OR S$<>"T" THEN GOSUB 30000:GOTO 6520
6540 GOSUB 30050:PRINT "1. I1";S$:PRINT "1. REIHE (1-8) ? 1 2 3 4 5 6 7 8";
6550 POKE 198,0:WAIT 198,1:GETR$
6560 IF R$<>"1" OR R$<>"8" THEN GOSUB 30000:GOTO 6550
6565 PRINT "2. I1";R$:GOSUB 30050
6570 FOR I=1 TO 3:Z(Z2,I,2)=ASC(S$)-64:Z(Z2,1,1)=VAL(R$)+I-1:NEXT
6590 O=Z(Z2,1,1)-1:IFO=0 THEN O=1
6600 U=Z(Z2,3,1)+1:IFU=11 THEN U=10
6610 A=Z(Z2,1,2)-1:IFA=0 THEN A=1
6620 E=Z(Z2,1,2)+1:IFE=21 THEN E=20
6630 FOR I=0 TO U:FOR K=ATOE
6640 IF P$(20*(I-1)+K)=0 THEN NEXT K, I:GOTO 6700
6650 GOSUB 30400
6660 Z2=Z2-1:R1$="":GOTO 6020
6700 R1$=""
6710 FOR I=1 TO 3:P$(20*(Z(Z2,1,1)-1)+Z(Z2,1,2))=Z2+10:NEXT
6720 FOR I=1 TO 3:POKEBS+22*(Z(Z2,1,1)+3)+Z(Z2,1,2)+1,224:NEXT
6730 IF Z2<2 THEN GOSUB 30500:GOTO 6020
7000 GOSUB 30500
7010 U1=VAL(U$)
7020 U2=U2+1:PRINT "3. U2" II. U-BOOT (2)"
7030 PRINT "XXXXXXXXXXXXXXXXXXXX? HORIZONTAL, VERTICAL":FOR K=1 TO 200:NEXT K
7040 PRINT "XXXXXXXXXXXXXXXXXXXX? V":FOR K=1 TO 200:NEXT K
7050 GETP$:IF P$<>"H" AND P$<>"V" THEN 7030
7060 GOSUB 30050:IF P$="V" THEN 7500
7070 PRINT "XXXXXXXXXXXXXXXXXXXX?":PRINT "REIHE (1-10) ? 1 2 3 4 5 6 7 8 9 10";
7080 FOR L=1 TO 2
7090 POKE 198,0:WAIT 198,1:GETR$
7100 IFL=1 THEN IF R$<>"0" AND R$<>"1" THEN GOSUB 30000:GOTO 7090
7110 IFL=2 AND R1$="1" THEN IF R$<>"0" THEN GOSUB 30000:GOTO 7090
7120 IFL=2 AND R1$="0" THEN IF R$<>"1" OR R$<>"9" THEN GOSUB 30000:GOTO 7090
7130 GOSUB 30050:PRINT "R1$";R1$=R1$+R$:NEXT L:PRINT
7140 PRINT "1. SPALTE (A-S) ? 1 2 3 4 5 6 7 8 9 10";
7150 POKE 198,0:WAIT 198,1:GETS$
7160 IF S$<>"A" OR S$<>"S" THEN GOSUB 30000:GOTO 7150
7170 PRINT "2. I1";S$:GOSUB 30050
7180 FOR I=1 TO 2:U(U2,I,1)=VAL(R1$):U(U2,I,2)=ASC(S$)-65+I:NEXT
7190 IF U2<2 THEN 7300
7200 FOR K=1 TO 2:FOR I=1 TO U2-1:FOR L=1 TO 2
7210 IF U(U2,K,1)<>U(U2,I,L,1) OR U(U2,K,2)<>U(U2,I,L,2) THEN NEXT L, I, K:GOTO 7300
7220 GOSUB 30300
7230 U2=U2-1:R1$="":GOTO 7020
7300 A=U(U2,1,2)-1:IFA=0 THEN A=1
7310 E=U(U2,2,2)+1:IFE=21 THEN E=20
7320 O=U(U2,1,1)-1:IFO=0 THEN O=1

```

programme

```

7330 U=U(2,1,1)+1:IFU=11THENU=10
7340 FORI=0TOU:FORK=ATOE
7350 IFP%(20*(I-1)+K)=0THENNEXTK,I:GOTO7400
7360 GOSUB30400
7370 U2=U2-1:R1$="":GOTO7020
7400 R1$=""
7410 FORI=1TO2:P%(20*(U(2,1,1)-1)+U(2,1,2))=U2+20:NEXT
7430 FORI=1TO2:POKEBS+22*(U(2,1,1)+3)+U(2,1,2)+1,224:NEXT
7440 IFU2<U1THENGOSUB30500:GOTO7020
7450 GOTO8000
7500 PRINT"#####VERTIKAL";
7510 PRINT"SPALTE (A-T) ? ☐ ☒ ☐";
7520 POKE198,0:WAIT198,1:GETS$
7530 IF$<"A"OR$>"T"THENGOSUB30000:GOTO7520
7540 GOSUB30050:PRINT"☒ ☐";S$:PRINT"☒1.REIHE (1-9) ? ☐ ☒ ☐";
7550 POKE198,0:WAIT198,1:GETR$
7560 IFR$<"1"ORR$>"9"THENGOSUB30000:GOTO7550
7565 PRINTR$:GOSUB30050
7570 FORI=1TO2:U(2,1,2)=ASC(S$)-64:U(2,1,1)=VAL(R$)+1-1:NEXT
7590 O=U(2,1,1)-1:IFO=0THENO=1
7600 U=U(2,2,1)+1:IFU=11THENU=10
7610 A=U(2,1,2)-1:IFA=0THENA=1
7620 E=U(2,1,2)+1:IFE=21THENE=20
7630 FORI=0TOU:FORK=ATOE
7640 IFP%(20*(I-1)+K)=0THENNEXTK,I:GOTO7700
7650 GOSUB30400
7660 U2=U2-1:R1$="":GOTO7020
7700 R1$=""
7710 FORI=1TO2:P%(20*(U(2,1,1)-1)+U(2,1,2))=U2+20:NEXT
7720 FORI=1TO2:POKEBS+22*(U(2,1,1)+3)+U(2,1,2)+1,224:NEXT
7730 IFU2<U1THENGOSUB30500:GOTO7010
8000 REM COMPUTER-SCHIFFE
8010 FORI=38TO160STEP.5
8020 POKE3685,I:NEXT
8030 PRINT"#####INEN AUGENBLICK BITTEICH SETZE MEINE SCHIFFE."
8040 POKE3685,38
8100 IFK1=0THEN8400
8110 FORK2=1TOK1
8120 IFRND(1)>.5THEN8200
8130 P1=INT(RND(1)*10)+1:P2=INT(RND(1)*16)+1
8140 A=P2-1:IFA=0THENA=1
8150 E=P2+5:IFE>20THENE=20
8160 O=P1-1:IFO=0THENO=1
8170 U=P1+1:IFU=11THENU=10
8180 FORI=0TOU:FORK=ATOE
8190 IFC%(20*(I-1)+K)=0THENNEXTK,I:GOTO8194
8192 GOTO8130
8194 FORI=1TO4:C%(20*(P1-1)+P2+1)=K2:NEXT
8196 PRINT"#####K2### KREUZER":NEXTK2:GOTO8400
8200 P1=INT(RND(1)*7)+1:P2=INT(RND(1)*20)+1
8210 A=P2-1:IFA<1THENA=1
8220 E=P2+1:IFE>20THENE=20
8230 O=P1-1:IFO<1THENO=1
8240 U=P1+5:IFU>10THENU=10
8250 FORI=0TOU:FORK=ATOE
8260 IFC%(20*(I-1)+K)=0THENNEXTK,I:GOTO8300
8270 GOTO8200
8300 FORI=1TO4:C%(20*(P1+1-2)+P2)=K2:NEXTI
8310 GOTO8196
8400 IFZ1=0THEN8700
8410 FORZ2=1TOZ1
8420 IFRND(1)>.5THEN8600
8430 P1=INT(RND(1)*10)+1:P2=INT(RND(1)*17)+1
8440 A=P2-1:IFA=0THENA=1
8450 E=P2+4:IFE>20THENE=20
8460 O=P1-1:IFO=0THENO=1
8470 U=P1+1:IFU=11THENU=10
8480 FORI=0TOU:FORK=ATOE
8490 IFC%(20*(I-1)+K)=0THENNEXTK,I:GOTO8510
8500 GOTO8430
8510 FORI=1TO3:C%(20*(P1-1)+P2+1)=Z2+10:NEXT
8520 PRINT"#####Z2### ZERSTOERER":NEXTZ2:GOTO8700
8600 P1=INT(RND(1)*3)+1:P2=INT(RND(1)*20)+1
8610 A=P2-1:IFA<1THENA=1
8620 E=P2+1:IFE>20THENE=20
8630 O=P1-1:IFO<1THENO=1
8640 U=P1+4:IFU>10THENU=10
8650 FORI=0TOU:FORK=ATOE
8660 IFC%(20*(I-1)+K)=0THENNEXTK,I:GOTO8680
8670 GOTO8600
8680 FORI=1TO3:C%(20*(P1+1-2)+P2)=Z2+10:NEXTI
8690 GOTO8520
8700 :
8710 FORU2=1TOU1
8720 IFRND(1)>.5THEN8900
8730 P1=INT(RND(1)*10)+1:P2=INT(RND(1)*18)+1
8740 A=P2-1:IFA=0THENA=1
8750 E=P2+3:IFE>20THENE=20

```



```

8760 O=P1-1:IF0=0THENO=1
8770 U=P1+1:IFU=11THENU=10
8780 FORI=0TOU:FORK=ATOE
8790 IFC%(20*(1-1)+K)=0THENNEXTK,1:GOTO8810
8800 GOTO8730
8810 FORI=1TO2:C%(20*(P1-1)+P2+1)=U2+20:NEXT
8820 PRINT"599"U2"11. U-BOOT":NEXTU2:GOTO9000
8900 P1=INT(RND(1)*9)+1:P2=INT(RND(1)*20)+1
8910 A=P2-1:IFA<1THENA=1
8920 E=P2+1:IFE>20THENE=20
8930 O=P1-1:IF0<1THENO=1
8940 U=P1+3:IFU>10THENU=10
8950 FORI=0TOU:FORK=ATOE
8960 IFC%(20*(1-1)+K)=0THENNEXTK,1:GOTO8930
8970 GOTO8900
8980 FORI=1TO2:C%(20*(P1+1-2)+P2)=U2+20:NEXTI
8990 GOTO8820
9000 REM SPIELFELDAUFBAU
9010 PRINT"1000":FORI=65TO84:PRINTCHR$(I):NEXT
9020 FORI=1TO9:PRINTI:NEXT:PRINT"10"
9030 FORI=2TO21:FORK=2TO11
9040 POKEBS+22*K+1,79:POKEFA+22*K+1,6
9050 NEXTK,I
9060 PRINT:FORI=1TO9:PRINTI:NEXT:PRINT"10";
9070 FORI=2TO21:FORK=13TO22
9080 POKEBS+22*K+1,79:POKEFA+22*K+1,6
9090 NEXTK,I
9100 REM SCHIFFE
9110 FORK=0TO9:FORI=1TO20:P=K*20+I
9120 Z=224:A1=1:A2=-1:A3=20:A4=-20
9130 IFP%(P)=0THEN3700
9135 IFK=0ORK=9OR1=10R1=20THENGOSUB9600
9140 IFP%(P+A2)=0ANDP%(P+A4)=0ANDP%(P+A3)=0THENZ=107:GOTO9700
9150 IFP%(P+A1)=0ANDP%(P+A4)=0ANDP%(P+A3)=0THENZ=106:GOTO9700
9160 IFP%(P+A1)<0ANDP%(P+A2)<0ANDP%(P+A3)<0THENZ=103:GOTO9700
9170 IFP%(P+A4)=0ANDP%(P+A2)=0ANDP%(P+A1)=0THENZ=105:GOTO9700
9180 IFP%(P+A3)=0ANDP%(P+A2)=0ANDP%(P+A1)=0THENZ=104:GOTO9700
9190 IFP%(P+A3)<0ANDP%(P+A4)<0ANDP%(P+A2)<0THENZ=102:GOTO9700
9195 STOP
9200 IFK=0THENA4=-P
9210 IFK=9THENA3=-P
9220 IF1=1THENA2=-P
9230 IF1=20THENA1=-P
9240 RETURN
9250 POKEBS+22*(K+13)+I+1,2
9260 IFZ<>224THENPOKEFA+22*(K+13)+I+1,0
9270 NEXTI,K
9280 FORI=0TO21:FORK=0TO9
9290 IFPEEK(BS+22*(K+13)+1)=224THENPOKEBS+22*(K+13)+1,32
9300 NEXTK,I
9310 FORI=0TO21:POKEBS+22*12+1,64:POKEFA+22*12+1,6:NEXT
10000 REM SPIELEEREINGABE
10010 PRINT"331E BEGINNEN":FORI=1TO3E3:NEXT:PRINT"3
10020 PRINT"31HRE EINGABE BITTE:":FORI=1TO3E3:NEXT:PRINT"3
10100 PRINT"31HR ZIEL: 3 3-3 3";
10110 POKE198,0:WAIT198,1:GETS$:IFS$="3"THEN12000
10120 IFS$="A"ORS$="T"THENGOSUB30000:GOTO10110
10130 PRINT"3333333333"3$3";
10140 R1$="":FORL=1TO2
10150 POKE198,0:WAIT198,1:GETR$:IFR$="3"THEN12000
10160 IFL=1ANDR$<>"0"ANDR$<>"1"THENGOSUB30000:GOTO10150
10170 IFL=2ANDR1$="1"ANDR$<>"0"THENGOSUB30000:GOTO10150
10180 IFL=2ANDR1$="0"ANDR$<>"1"ORR$<>"3"THENGOSUB30000:GOTO10150
10190 R1$=R1$+R$:PRINTR$:NEXTL
10200 P1=VAL(R1$):P2=ASC(3$)-64
10210 PP=20*(P1-1)+P2:IFC%(PP)=0THEN10300
10220 ZZ=C%(PP):C%(PP)=99:IFZZ=99THEN10800
10230 FORI=1TO200:IFC%(I)<>ZZTHENNEXT:GOTO10400
10240 PRINT"33"TAB(15)"TREFFER3"
10250 POKEBS+22*(P1+1)+P2+1,224:POKEFA+22*(P1+1)+P2+1,4:GOSUB30300
10270 GOSUB31000:GOTO10100
10400 PRINT"33 3 VERSENKT 3";
10410 POKEBS+22*(P1+1)+P2+1,224:POKEFA+22*(P1+1)+P2+1,4:GOSUB30300:GOSUB30900:GO
SUB31000
10430 X=0:FORI=1TO200:X=X+C%(I):NEXT:IFX/99=INT(X/99)+1E-6THEN10100
10440 GOTO13000
10800 PRINT"33WAR SCHON GETROFFEN!3":GOSUB30700
10810 GOTO11000
10900 PRINT"3"TAB(15)"WASSER"
10310 POKEBS+22*(P1+1)+P2+1,114:GOSUB30600
11000 REM COMPUTERZUG
11010 IFXTHENIFX/99=INT(X/99)THEN13000
11015 GOSUB31000
11020 PRINT"3333333333"3-3 3 3";
11021 IFTETHEN11500
11022 IFRND(1)>SWTHEN11030
11023 FORI=1TO200:IFP%(I)=0ORP%(I)>40THENNEXT:STOP
11024 P=1:P1=INT((P-1)/20):P2=INT((P-1)*20):GOTO11070

```

```

11025 IFTETHEN11500
11030 P=INT(RND(1)*200+1)
11040 IFP%(P)=99ORP%(P)=98THEN11030
11050 P1=INT((P-1)/20):P2=INT(P-P1*20)
11060 IFP1/2=INT(P1/2)ANDP2/2<>INT(P2/2)THEN11030
11065 IFP1/2<>INT(P1/2)ANDP2/2=INT(P2/2)THEN11030
11070 IFP%(P)=0THENP%(P)=98:GOTO11300
11100 Z=P%(P):P%(P)=99:FORI=1TO200
11110 IFP%(1)<>ZTHENNEXT:GOTO11200
11126 R$=CHR$(P2+64)+"-":IFP1=9THENR$=R$+"10":GOTO11130
11127 R$=R$+"0"+RIGHT$(STR$(P1+1),1)
11130 POKEFA+22*(P1+13)+P2+1,4
11135 PRINT"§"TAB(10)R$ " §TREFFER§":PL=0:LO=LO+1:T(LO)=P
11140 GOSUB30800:GOSUB31000
11150 IFTE=0THEENTE=P:TM=TE:GOTO11020
11160 GH=1:GOTO11020
11200 P1=INT((P-1)/20):P2=INT(P-P1*20)
11210 R$=CHR$(P2+64)+"-":IFP1=9THENR$=R$+"10":GOTO11230
11220 R$=R$+"0"+RIGHT$(STR$(P1+1),1)
11225 PRINT"§"TAB(10)R$:FORK=1TO100:NEXT
11230 PRINT"§§ § VERSENKT §§":LO=LO+1:T(LO)=P
11240 POKEFA+22*(P1+13)+P2+1,4:PL=0:GH=0:TE=0:PO=0:TM=0
11250 FORI=1TOLO:IFT(1)<=180THENIFP%(T(1)+20)=0THENP%(T(1)+20)=98
11260 IFT(1)>20THENIFP%(T(1)-20)=0THENP%(T(1)-20)=98
11270 IF(T(1)-INT(T(1)/20)*20)<>1ANDP%(T(1)-1)=0THENP%(T(1)-1)=98
11280 IF(T(1)-INT(T(1)/20)*20)<>0ANDP%(T(1)+1)=0THENP%(T(1)+1)=98
11290 NEXTI:LO=0:GOSUB30800:GOSUB30900:GOSUB31000
11295 X1=X1+1:IFX1<>K1+21+U1THEN11020
11296 GOTO14000
11300 :
11305 R$=CHR$(P2+64)+"-":IFP1=9THENR$=R$+"10":GOTO11310
11306 R$=R$+"0"+RIGHT$(STR$(P1+1),1)
11310 PRINT"§"TAB(10)R$ " WASSER":
11320 POKEBS+22*(P1+13)+P2+1,113:PL=1
11330 GOSUB30600:GOSUB31000:GOTO11900
11500 IFGH=1THEN11700
11510 IFGH=0THENPO=PO+1:GOTO11520
11515 PL=1:GOTO11700
11520 IFTE<=20ANDPO=1THEN11510
11530 IFTE>180ANDPO=3THEN11510
11540 IF(TE-INT(TE/20)*20)=1ANDPO=4THENSTOP
11545 IF(TE-INT(TE/20)*20)=0ANDPO=2THEN11510
11550 IFGH=1THEN11600
11560 IFPO=1THCNP=TE-20
11570 IFPO=2THENP=TE+1
11580 IFPO=3THENP=TE+20
11590 IFPO=4THENP=TE-1
11595 IFGH=1THENGOTO11520
11600 IFP%(P)=99ORP%(P)=98ANDGH=0THEN11510
11605 IFP%(P)=99ORP%(P)=98THENPL=1:GOTO11700
11610 P1=INT((P-1)/20):P2=INT(P-P1*20):GOTO11070
11700 IFPL=0THEENTE=P:GOTO11560
11710 IFPO=1ORPO=2THENPO=PO+2:GOTO11730
11720 IFPO=3ORPO=4THENPO=PO-2
11730 TE=TM:GH=0
11800 PL=0:GOTO11560
11900 GOTO10100
12000 REM AEBRUCH
12005 PRINT"§§ ***** SCHADE *****§":FORI=240TO140STEP-1:FORK=1TO3
12010 POKET1,I+K:NEXTK:FORK=3TO1STEP-1:POKET1,I+K:NEXTK,1
12020 POKET1,0:PRINT"§ "
12030 PRINT"§§MEINE SCHIFFE:"
12040 FORI=0TO9:FORK=1TO20:P1=I*20+K:P2=22*(I+2)+K+1
12060 IFPEEK(BS+P2)=224THENPOKEBS+P2,113
12070 IFCK(P1)<>0THENPOKEBS+P2,224:POKEFA+P2,4
12100 NEXTK,1:GOSUB30100
12110 S1=0:C1=0:FORI=1TO200
12120 IFCK(1)=99THENS1=S1+1
12130 IFP%(1)=99THENC1=C1+1
12140 NEXT:G=K1*4+21*3+U1*2
12150 S2=INT(S1/G*100):C2=INT(C1/G*100)
12200 PRINT"§SIE HABEN VERSENKT:§"
12210 PRINTS1TAB(3)"EINHEIT" = "S2"§§":IFS1<>1THENPRINT"Q"TAB(10)"EN"
12220 PRINT"§§ICH HABE VERSENKT:§"
12230 PRINTC1"EINHEIT" = "C2"§§":IFC1<>1THENPRINT"Q"TAB(10)"EN"
12240 PRINT"§§":IFC1=S1THEN12300
12250 PRINT"§§DAMIT HABEN SIE GE-§§§§JONNEN.":PRINT"§§§§§§§§GRATULIERE!§"
12260 FORI=1TO10:FORK=200TO240:POKET2,K:NEXTK:FORK=240TO200STEP-1
12270 POKET2,K:NEXTK,1:POKET2,0:GOTO12500
12300 IFC1=S1THEN12400
12310 PRINT"DAMIT HABE ICH GE-§§JONNEN."
12320 FORI=1TO5:FORL=254TO128STEP-1:POKET2,L:NEXTL,1:POKET2,0
12350 GOTO12500
12400 PRINT"§§§§SIE ICHSTAND"
12410 FORI=254TO128STEP-1:POKET1,1:POKET2,1:NEXT
12500 GOSUB30100
12510 PRINT"§NOCH§§EIN§§SPIEL§§?"

```



```

12520 POKE198,0:WAIT198,1:GETA$
12530 IFA$<>"J"ANDA$<>"N"THENGOSUB30000:GOTO12520
12540 IFA$="J"THENRUN
12550 ED=1:GOTO1000
13000 REM SPIELER-SIEG
13010 FOR1=1TO200
13020 IFC%(1)=99THENS1=S1+1
13030 IFP%(1)=99THENC1=C1+1
13040 NEXT
13050 G=K1*4+Z1*3+U1*2
13060 S2=INT(S1/G*100):C2=INT(C1/G*100)
13070 GOTO12200
14000 REM COMPUTER-SIEG
14010 FOR1=1TO200
14020 IFC%(1)=99THENS1=S1+1
14030 IFP%(1)=99THENC1=C1+1
14040 NEXT
14050 G=K1*4+Z1*3+U1*2
14060 S2=INT(S1/G*100):C2=INT(C1/G*100)
14070 GOTO12030
30000 POKET1,128:POKE198,0
30010 FORZZ=1TO200:NEXT
30020 POKET1,0:RETURN
30050 POKET3,240
30060 FORZZ=1TO50:NEXTZZ
30070 POKET3,0:RETURN
30100 POKE198,0:POKEFA+505,0
30110 GETA$:IFA$<>"N"THENRETURN

30120 POKEBS+505,127:FORK=1TO200:NEXTK:POKEBS+505,255:FORK=1TO200:NEXTK
30130 GOTO30110
30200 S1=0:RETURN
30300 GOTO30400
30400 PRINT"53 DIESE POSITION IST NICHT MOEGLICH." :GOSUB30000
30410 FORZZ=1TO2E3:NEXT:GOSUB30500:RETURN
30500 FORZZ=330TO480:POKEBS+ZZ,32:NEXT
30510 FORZZ=0TO44:POKEBS+ZZ,32:NEXT
30520 RETURN
30600 POKET2,200:POKET3,199:POKET4,254
30610 FORI=15TO0STEP-.2:POKELA,I:NEXT:POKELA,15
30620 POKET2,0:POKET3,0:POKET4,0:RETURN
30700 FORI=200TO130STEP-1:FORK=1TO3
30710 POKET3,I+K:POKET2,I+K
30720 NEXTK:FORK=3TO1STEP-1
30730 POKET3,I+K:POKET2,I+K:NEXTK,I
30740 POKET3,0:POKET2,0:RETURN
30800 POKET4,128:FORI=15TO0STEP-.07:POKELA,I:NEXT
30810 POKET4,0:POKELA,15:RETURN
30900 FORI=1TO3:FORK=180TO240STEP.6
30910 POKET2,K:NEXTK:POKET2,0:FORK=1TO2E2:NEXTK:NEXTI:RETURN
31000 FORJ=1TO2E3:NEXT:PRINT"
";:RETURN
READY.

```

TASTATUR-MEISTER® ist da!

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

BASIC Funktionen

STEUER-TASTEN

Schneller Überblick



ASCII Tabelle zum Aufhängen
(Nicht abgebildet)

TASTATUR-MEISTER®

- Befehle vollständig vorhanden, mit voll ausgeschriebener 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. 089/53 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

BARRIERE

für den Commodore 64

Zunächst ist der Rechner einmalig mit dem Programm "Barrieredata" zu laden. Dieses erzeugt auf der Diskette das Maschinenspracheprogramm "mpg".

Nun ist der Rechner mit dem eigentlichen Spielprogramm "Barriere" zu laden. Es handelt sich hier um eine Version des Spieles "Break out". Nach dem Starten fragt das Programm nach der gewünschten Spielversion. Diese ist durch Rechts- und Linksbewegen des Joysticks 1 einzugeben:

Version 1: Mauer sichtbar, normale Geschwindigkeit, ein Spieler

Version 2: = Version 1, jedoch für zwei Spieler

Version 3: Mauer unsichtbar, normale Geschwindigkeit, ein Spieler

Version 4: = Version 3, jedoch für zwei Spieler

Version 5: Mauer sichtbar, schneller Spielmodus, ein Spieler

Version 6: = Version 5, jedoch für zwei Spieler

Version 7: Mauer unsichtbar, schneller Spielmodus, ein Spieler

Version 8: = Version 7, jedoch für zwei Spieler

Version 9: Anfängerkurs (Mauer sichtbar, langsamer Spielmodus, nur ein Spieler)

Nach der Auswahl wird das Spiel durch Drücken des Feuerknopfes von Joystick 1 initialisiert.

Nach erneutem Druck des Feuerknopfes

erscheint der Ball. Dieser ist nun mit dem Schläger zu treffen. Prallt der Ball gegen die Mauer, so wird dort ein Stein herausgelöst, wofür es je nach Farbe Punkte gibt (2,5,8).

Zu Bemerken ist noch, daß der Ball in gewisser Weise zu beeinflussen ist. Trifft man diesen nämlich mit der Schlägerkante, so wird er in die gleiche Richtung geschlagen, aus der er gekommen ist.

Wem das Spiel zu schnell oder zu langsam erscheint kann durch Ändern der Befehle POKE 49673,x bzw. POKE 49801,x im Basicteil, die Geschwindigkeit beeinflussen. x sollte jedoch nicht kleiner als drei gewählt werden, da sonst Fehler im Spielablauf auftreten können.

```

100 OPEN1,8,1,"MPG":PRINT#1,CHR$(0)CHR$(192):A=0
110 FORI=49152TO50313:READQ:PRINT#1,CHR$(Q):A=A+Q:NEXT:CLOSE1
120 IFAC>145326THENPRINT"ACHTUNG FEHLER IN DATAZEILEN"
121 DATA 173,16,208,41,1,208,16,238,0,208,208,8,173,16,208,9
122 DATA 1,141,16,208,76,79,192,173,0,208,201,66,240,49,238,0
123 DATA 208,76,79,192,173,16,208,41,4,208,14,238,4,208,208,8
124 DATA 173,16,208,9,4,141,16,208,96,173,4,208,201,66,240,26
125 DATA 238,4,208,96,173,1,220,201,247,240,181,201,251,240,12,173
126 DATA 0,220,201,119,240,206,201,123,240,38,96,173,16,208,41,1
127 DATA 208,13,173,0,208,201,10,240,241,206,0,208,76,79,192,206
128 DATA 0,208,16,219,56,173,16,208,233,1,141,16,208,76,79,192
129 DATA 173,16,208,41,4,208,11,173,4,208,201,10,240,204,206,4
130 DATA 208,96,206,4,208,16,195,56,173,16,208,233,4,141,16,208
131 DATA 96,238,2,208,208,8,173,16,208,9,2,141,16,208,108,58
132 DATA 199,173,2,208,208,9,56,173,16,208,233,2,141,16,208,206
133 DATA 2,208,108,58,199,238,3,208,96,206,3,208,96,108,56,199
134 DATA 32,68,192,32,205,192,32,68,192,173,3,208,201,250,208,3
135 DATA 76,97,194,201,74,208,14,169,197,141,58,199,169,1,133,146
136 DATA 133,165,32,86,196,173,16,208,41,2,208,15,173,2,208,201
137 DATA 53,176,23,169,161,141,56,199,76,23,193,173,2,208,201,32
138 DATA 144,8,169,177,141,56,199,32,86,196,120,173,30,208,240,72
139 DATA 169,0,133,165,32,112,196,169,201,141,58,199,173,16,208,41
140 DATA 2,240,11,173,16,208,41,4,208,4,24,76,63,193,56,173
141 DATA 2,208,237,4,208,174,56,199,224,177,240,12,201,0,16,17
142 DATA 169,177,141,56,199,76,97,193,201,8,48,5,169,161,141,56
143 DATA 199,165,166,240,3,32,181,195,169,40,133,167,56,173,2,208
144 DATA 233,12,176,8,74,74,74,133,178,76,143,193,74,74,74,133
145 DATA 178,173,16,208,41,2,240,7,24,165,178,105,32,133,178,56
146 DATA 173,3,208,233,48,74,74,74,133,179,133,171,169,0,133,2
147 DATA 162,8,70,167,144,3,24,101,179,106,102,2,202,208,243,133
148 DATA 179,24,165,2,101,178,133,178,165,179,105,4,133,179,160,0
149 DATA 177,178,201,160,208,65,173,58,199,201,197,240,12,165,165,208
150 DATA 54,169,197,141,58,199,76,226,193,165,165,240,42,169,201,141

```

19. September '84

TROPIC TROUBLE

für den VC-20 ohne Erweiterung

Ziel des Spieles ist es, Eingeborene einer tropischen Insel zu retten, auf welcher vor kurzem ein Vulkan ausgebrochen ist. Der Spieler steuert einen Hubschrauber und muß versuchen, auf der Insel zu landen. Danach sollte man aufpassen, daß nicht mehr als fünf Eingeborene in den Hubschrauber einsteigen, da ein überladener Hubschrauber nicht in der Lage wäre, über den Düsen-Jet zu fliegen.

Hat man trotzdem zu viele Eingeborene im Hubschrauber, so kann man sie über Bord werfen.

Fliegt man dann über dem Düsen-Jet, so muß man versuchen, die Eingeborenen auf das Flugzeug zu werfen.

Gesteuert wird mit den Tasten "P", "L", ";", und "." und abgeworfen wird mit der Taste Z. Viel Erfolg beim Retten der Eingeborenen.

```

5 REMBY JOHN BETTELS & RAINER PAHMEYER /84
10 POKE52,29:POKE56,29:POKE55,29:POKE36879,30
20 FORI=7464TO7679:READA:POKEI,A:NEXTI
75 DATA63,32,16,0,64,4,32,0
90 DATA4,41,1,64,8,0,2,0
85 DATA160,9,82,224,205,32,130,73
90 DATA68,64,18,169,43,5,72,130
95 DATA64,66,0,16,128,128,34,84
100 DATA0,68,0,0,4,37,145,8
104 DATA129,66,60,129,129,60,66,129
105 DATA128,66,4,1,194,36,149,86
110 DATA28,190,255,255,255,255,255,255
115 DATA255,255,255,255,255,255,255,255
120 DATA63,128,192,255,3,1,0,15
125 DATA126,128,248,228,226,252,137,254
130 DATA0,0,15,19,34,255,127,0
135 DATA3,7,255,255,7,255,254,0
140 DATA8,8,8,28,8,28,28,20
145 DATA60,127,207,1,28,62,115,65
150 DATA0,0,0,152,188,254,227,193
155 DATA192,224,96,112,48,48,56,24
160 DATA24,24,16,60,48,112,148,72
165 DATA20,190,255,255,255,255,255,255
180 DATA126,1,31,39,71,63,145,127,0,1,3,255,192,128,0,240
181 DATA128,135,255,252,190,252,255,199,1,0,31,63,237,255,127,31
182 DATA0,0,0,0,0,24,54,193
183 DATA0,1,0,3,1,3,1,0
184 DATA0,0,1,3,15,31,127,255
200 POKE36869,255
215 PRINT"███ 545██ TROPIC TROUBLE 545"
216 PRINT"████ 562=====███"
220 PRINT"███ YOU MUST SAVE THE NATIVES 57██ WITH YOUR HELICOPTER!███
:██"
225 PRINT"██ THE SUBMARINE ██;██ IS SHOOTING ROCKETS 53██ TO YOU."
230 PRINT"██ THE NATIVES MUST BE PUT IN THE AIRCRAFT.██████12"
235 PRINT"██+-----"
237 PRINT"██████ PRESS RETURN"
240 IFPEEK(197)<>15THEN240
245 POKE36869,240:POKE36879,110
250 PRINT"██████LOAD██"
READY.
```



Du hättest zumindest den
Akustikkoppler mit in's
Rettungsboot nehmen können, Erna!

```

2 PRINT "S";:FOR I=1 TO 101:PRINT " ";:NEXT:GOTO 400
5 WW=2:R=48:R1=7544:R2=63:R3=126:F=22
6 POKE 8185,46:POKE 38905,7
10 POKE 36869,255:X=7724:Z=X
11 PRINT "SPC(14)"
12 PRINT ".....OOC===== ";
13 PRINT "3"
20 PRINT "SAVED: 0 DEAD: 0MEN IN+++++X";
25 PRINT "+++++"
100 POKER1,0:POKER1+8,R3:POKE 36877,128:K=(X-(22*(INT((X-7680)/22)))-7680:IFY>KT
HENY=Y-1
101 Y=Y-(Y+1<K)+(Y>1)
103 IFY=INT(K)ORWW=1THEN170
104 IFY<>ETHENPOKE 8165+E,160:POKE 8164+E,160:POKE 8164+Y,60:POKE 8165+Y,59:E=Y
105 B=B+3:C=0:IFB>22THENX=X+22:B=0:C=1
106 A=PEEK(197):X=X-(A<37ANDC=0)*22+(A=13)*22
107 IFA=21THENR=58:R1=7624:R2=126:R3=252:U=U-.17:IFU<-1THENU=-1
108 IFA=22THENR=48:R1=7544:R2=63:R3=126:U=U+.17:IFU>1THENU=1
109 X=X+U:IF INT(X)<>2 THEN140
110 POKER1+8,0:POKER1,R2:POKE 36877,0:IFPEEK(7788+F)<>160THEN300
112 PRINT "SPC(F)"":F=F-1:IFF=1THENF=22:GOSUB 320:PRINT " "
115 IFPEEK(197)=33ANDM>0ORP=-1THEN121
117 GOTO 100
121 IFF=0THENQ=X:P=-1:B=0
123 Q=Q+22:IFPEEK(Q)<>160THENP=0:M=M-1:PRINT "SPC(17)M:" ":GOTO 250
125 POKEQ,55:IFX+22<>QTHENPOKEQ-22,160
130 GOTO 100
140 POKEZ,160:POKEZ+1,160
141 IFPEEK(X)<>160ORPEEK(X+1)<>160THEN150
142 IFM>5ANDX<7811THENX=Z
143 POKE X,R-1:POKE X+1,R:Z=X:GOTO 110
150 IFX>8135ANDX<8141THENX=Z:GOTO 200
151 IFPEEK(X)=55ORPEEK(X+1)=55THEN143
155 A=0:B=0:POKE 36877,0:POKE 36878,15:FORI=233TO128STEP-1
156 A=A+1:IFA=20THENB=B+1:ONBGOSUB 160,161,162
157 POKE 36877,I:POKE 36875,I+7:NEXTI
158 POKE 36875,0:POKE 36877,0:RUN350
160 POKE 30721+X,2:POKE X+1,39:A=18:RETURN
161 POKE 30720+X,2:POKE X,40:A=17:RETURN
162 POKE 30698+X,2:POKE 30699+X,2:POKE 30742+X,2:POKE 30743+X,2
163 POKE X-22,42:POKE X-21,41:POKE X+22,38:POKE X+23,37:A=0:RETURN
170 IFWW=2THENC=8098+Y:L=8098+Y:WW=1:N=Y
174 IF INT(X)=CCORCC+1=INT(X)THENPOKE L,160:GOTO 155
175 POKE CC,51:POKE L,160:L=CC:CC=CC-22
176 DD=DD+1:IFDD=2THENDD=0:GOTO 179
177 G=(NKK):N=N-G:CC=CC-G:G=(N>K):N=N+G:CC=CC+G
179 IFPEEK(CC)=43THENWW=2:POKE L,160:POKE 36874,0
180 GOTO 104
200 D=8114:POKE X,R-1:POKE X+1,R
201 IFA=37THEN155
202 IFPEEK(197)=13THENFORT=8114TO8119:POKE 30720+T,3:NEXTT:GOTO 100
205 FORI=8114TOX-1:IFS=0THENPOKER1,R2:POKER1+8,0:POKE 36877,0
206 FORT=1TO40:NEXTT
210 S=S+1:IFS=2THENPOKER1,0:POKER1+8,R3:POKE 36877,128:S=0:GOTO 220
211 IFX>8115THENPOKE 30720+I,4:POKE I,55:D=I:NEXT:FORT=1TO40:NEXTT:POKE I
-1,160
213 M=M+1:PRINT "SPC(17)M"
214 IFM>10THEN155
215 GOTO 200
220 PRINT "SPC(F)"":F=F-1:IFF=1THENF=22:GOSUB 320:PRINT " "
221 GOTO 211
250 H=PEEK(Q)
255 IFH=61THENPOKEQ-22,160:POKEQ,44:FORT=0TO90:NEXT:POKE 36877,210:POKEQ,61
260 IFH=49ORH=50THENV=V+1:POKE 36875,200:PRINT "SPC(17)V:" ":POKEQ-22,160
261 IFH=49ORH=50THENFORT=1TO50:NEXTT:POKE 36875,0:GOTO 100
265 POKEQ-22,160:J=J+1:PRINT "SPC(17):J:IFJ>5THEN155
266 GOTO 100
300 IFPEEK(7788+F)=55THEN112
305 IFPEEK(7788+F)=51THEN100
310 GOTO 155
320 IFPEEK(7833)=57ORPEEK(7833)=58THEN155
325 RETURN
350 IFPEEK(197)=64THEN350
360 PRINT "S";:FOR I=1 TO 101:PRINT " ";:NEXT
400 POKE 36879,8:POKE 36878,15
401 POKE 8185,160:POKE 36869,255:FORI=0TO14:POKE 7544,63:POKE 7552,0:FORT=0TO70

```

```

402 NEXTT:POKE36877,0:READA:POKE38622+I,RND(1)*7+1:POKE7902+I,A:POKE7903+I,47
403 POKE7904+I,48:POKE7544,0:POKE7552,126:FORT=0TO100:NEXTT:POKE36877,128
404 IFPEEK(197)<>64THEN490
405 NEXTI:POKE36877,0
406 POKE198,0:PRINT"SOFTWARE BY JOHN BETTELS    RAINER PAHMEYER    64
50198"
407 PRINT"CONTROL: 1. RUN 2. STOP 3. CLEAR 4. HELP 5. QUIT 6. THROW OUT: 7
Z"
408 FORI=7924TO7939:POKE30720+I,4:POKE30721+I,4:POKEI,60:POKEI+1,59
409 POKE30719+I,6:POKEI-1,61:FORT=1TO100:NEXTT
435 GETA$:IFA$(">")THEN490
437 NEXTI
438 GETA$:IFA$("=")THEN498
490 PRINT"33";:FORI=1TO101:PRINT"  ";:NEXT:RUN5
500 DATA148,146,143,144,137,131,160,148,146,143,149,130,140,133,160
READY.

```

FUNKTIONIERT OHNE FRUST: KOMFORTABLE SOFTWARE VON DYNAMICS.

Weil es nicht nur Computer-Freaks gibt, bietet Dynamics anwender-freundliche Software – komfortabel, menue-gesteuert über den Bildschirm.

Wer also auf seinem Commodore C64, dem TI99/4A oder den Atari 400/800- oder XL-Modellen nicht lange programmieren will oder null Bock auf Listings hat, der geht ins Fachgeschäft oder Kaufhaus.

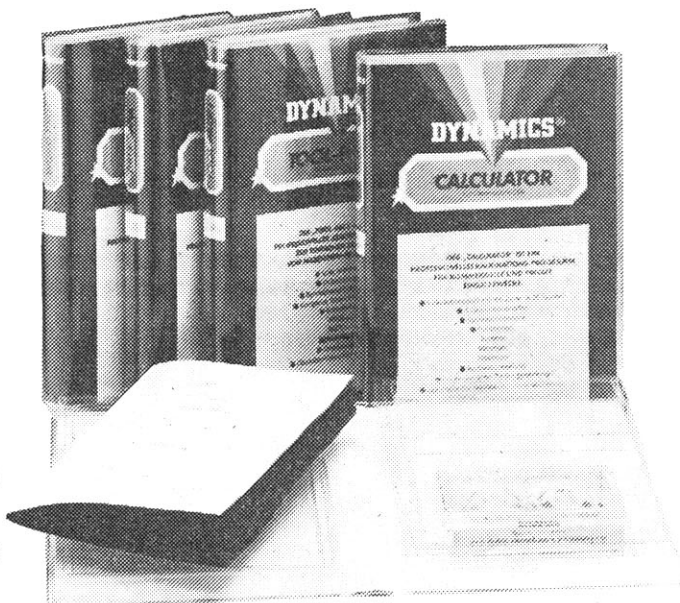
Dort gibt es das Dynamics „Adressen-Archiv“, das „Video-“, „Bücher-“ und „Schallplatten-“

Archiv“. Programme, die Ordnung halten – einfach und flink. Da gibt es neu den „Vokabel-Trainer“, der Lernen zum Vergnügen macht. Fragen Sie danach!

Wer mehr will, sollte sich die Dynamics Arbeits-Programme zeigen lassen. Aber auch „Word-Proc“, „Tool Pack I und II“, „Calculator“ oder „Data-Bank“ bleiben immer anwender-freundlich.

Gut verständliche Bedienungsanleitungen (keine Wälzer!), logischer Aufbau und einfache Anwendung garantieren, daß computern mit Dynamics Spaß macht!

Also: auf ins Fachgeschäft, hin zum Kaufhaus. Weil Dynamics Software einfach komfortabel ist. Ausprobieren!



DYNAMICS®

COMPUTER-SOFTWARE UND
COMPUTER-ZUBEHÖR.

FÜR ATARI 400/800 UND
600 XL/800 XL,
TEXAS INSTRUMENTS TI 99/4A
UND COMMODORE C 64.

Dynamics-Software
gibt es auf
Computer-Cassette
oder Diskette.

Dynamics Marketing GmbH,
Große Bäckerstraße 11,
2000 Hamburg 1

GRAPHIK

für C-64

$$\sin(x)/x + \sin(1/y)/y$$

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

Auch dieses Programm muß absolut geladen werden.

Für den Anwender besteht nun die Möglichkeit mittels:

SYS 49152 den Rechner in den Graphik-Modus zu schalten

SYS 49163 den Rechner in den LOW-Resolution Modus zurückzuschalten

SYS 49174, f die Farbe im High-Resolution Modus zu setzen, wobei f der Farbparameter ist, der wie folgt zu berechnen ist:

$F = \text{Hintergrundfarbe} + 16 \times \text{Vordergrundfarbe}$

SYS 49202 die High-Resolution Bitmap zu löschen

SYS 49224, 1, x, y einen Punkt in die Graphik zu setzen mit x kleiner 320, y kleiner 200

SYS 49224, 0, x, z einen Punkt in der Graphik zu löschen

SYS 49360, 1, x1, y1, x2, y2 eine Linie von P(x1, y1) nach Q(x2, y2) zu ziehen (LINE-Befehl).

SYS 49360, 0, x1, y1, x2, y2 eine Linie zu löschen

SYS 49848 eine Hardcopy der Graphik auf dem GP - 100 VC zu erstellen

SYS 50134, A\$, 8 bzw. 1 eine Graphik von Diskette oder Kassette einzuladen

SYS 50157, A\$, 8 bzw. 1 die Graphik unter A\$ abspeichern

SYS 49729, x, z, xl, yl Rasterpunkte in die Graphik zu setzen mit x, y als Zentrum, xl

gibt den horizontalen, yl den vertikalen Punkteabstand an

SYS 50192 den LIST Befehl als Statement zu benutzen

Als Programmbeispiel ist ein Kurvenplot Programm abgedruckt, welches die Graphikhilfen benutzt. Damit ist es möglich, jede beliebige, stetige und unstetige Funktion graphisch abzubilden. Die Funktion steht in Zeile 100.

Das Programm erklärt sich nach dem Starten mittels des Menüs von selber.

Sein Vorteil besteht darin, daß jede Kurve in einem durchgehenden Kurvenzug gezeichnet wird, zwischen den einzelnen errechneten Punkten also keine Lücke entsteht.

Wir nehmen an, der LINE BEFEHL sowie die Hardcopy Funktion werden jeden Programmierer eine große Hilfe sein.

```
50 IFPEEK(49152)<>169THENLOAD"GMFG",8,1
60 POKE53280,6:POKE53281,6:PRINTCHR$(14)"  GRAFISCHE DARSTELLUNG DER FUNKTION
70 SYS50192:LIST100:PRINT"TI"          "00":PRINT" 1 : PARTIELLISCHE DARSTELLUNG"
80 PRINT"02 : 3D-DARSTELLUNG":PRINT"03 : GRAFIK LADEN":INPUT"0 AUSWAHL: 10000";
P
90 IFP=3THENINPUT"0 NAME":A$:SYS49174,22:SYS49152:SYS50157,A$,8:GOTO300
100 DEFNFX(X)=COS(X)*4
110 POKE768,57:POKE769,194:IFP=2THEN350
120 PRINT:INPUT"UNKTIONEN--INKREMENT = 20000":L:SYS49202
130 INPUT"FAKTOR X--CHSE = 10000":S:RZ=19*S:R=RZ
140 INPUT"FAKTOR Y--CHSE = 10000":T:WZ=15*T:W=WZ
150 INPUT"0 SPRUNG HORIZONTAL -130/130 00000":C:F=160+C
160 INPUT"0 SPRUNG VERTIKAL -90/90 00000":D:E=100+D
```

```

170 INPUT "MASTER (Y/N) Y/N"; A$: IFA$="Y" THEN SYS49729, F, E, R, W
180 SYS49152: SYS49174, 22: SYS49360, 1, F, 0, F, 201: SYS49360, 1, 0, E, 319, E
190 FOR I=FT0319STEP R: SYS49360, 1, I, E-2, I, E+2: NEXT
200 FOR I=FT00STEP -R: SYS49360, 1, I, E-2, I, E+2: NEXT
210 FOR I=ET0199STEP W: SYS49360, 1, F-1, I, F+1, I: NEXT
220 FOR I=ET00STEP -W: SYS49360, 1, F-1, I, F+1, I: NEXT
230 A=0: B=E-(FNR(-F/R)*W): IF B>199 THEN B=199
240 IF B<0 THEN B=0
250 FOR G=1T0319STEP L: Y=E-(FNR((G-F)/R)*W): IF Y<0 THEN B=0: GOTO 290
260 IF Y>199 THEN B=199: GOTO 290
270 IF Y=0 THEN IF B=199 OR B=0 THEN B=290
280 SYS49360, 1, A, B, G, Y: B=Y
290 A=G: NEXT
300 POKE 198, 0: WAIT 198, 255: GET A$: SYS49163
310 INPUT "HARD COPY (Y/N) Y/N"; A$: IFA$<"Y" THEN 330
320 SYS49848: OPEN 4, 4: PRINT #4, CHR$(15): CMD4: LIST 100: PRINT #4: CLOSE 4
330 INPUT "SPEICHERN (Y/N) Y/N"; A$: IFA$="Y" THEN INPUT "NAME": A$: SYS50134, A$,
8
340 GOTO 60
350 PRINT "VERTIKAL-ASSTAB": INPUT " -40/40 20 "; C: A=144: B=2.25
360 PRINT "VERTIKALE-LAGE": INPUT " -50/150 90 "; F: D=.0327: E=160
370 G=199: SYS49202: SYS49152: SYS49174, 22
380 FOR H=-144T0144STEP B: O=INT(.5+SQR(A*H-H*H)): FOR K=-0T00: X=INT(.85*(K+H/B+E))
390 Y=FNR(SQR(K*K+H*H)*D)*C-H/B+F: Y1=INT(.9*(G-Y)): IF Y<0 OR Y>199 THEN NEXT: NEXT: GOT
0300
400 SYS49224, 1, X, Y1: SYS49360, 0, X, Y1+1, X, 199: NEXT: NEXT: GOTO 300

```

```

100 OPEN 1, 8, 1, "GMPG": PRINT #1, CHR$(0)CHR$(192): A=0
110 FOR I=49152T050280: READ Q: PRINT #1, CHR$(Q): A=A+Q: NEXT: CLOSE 1
120 IFA<164740 THEN PRINT "ACHTUNG FEHLER IN DATAZEILEN"
130 DATA 169, 59, 141, 17, 208, 169, 24, 141, 24, 208, 96, 169, 27, 141, 17, 208
140 DATA 169, 23, 141, 24, 208, 96, 32, 253, 174, 32, 158, 183, 160, 4, 132, 254
150 DATA 160, 0, 132, 253, 138, 162, 4, 145, 253, 200, 208, 251, 230, 254, 202, 208
160 DATA 246, 96, 169, 32, 133, 254, 169, 0, 133, 253, 168, 162, 32, 145, 253, 200
170 DATA 208, 251, 230, 254, 202, 208, 246, 96, 32, 253, 174, 32, 158, 183, 134, 150
180 DATA 32, 253, 174, 32, 235, 183, 165, 20, 133, 250, 165, 21, 133, 251, 138, 41
190 DATA 248, 133, 254, 133, 252, 169, 0, 133, 253, 6, 252, 38, 253, 6, 252, 38
200 DATA 253, 24, 165, 252, 101, 254, 133, 252, 165, 253, 105, 0, 133, 253, 6, 252
210 DATA 38, 253, 6, 252, 38, 253, 6, 252, 38, 253, 138, 41, 7, 24, 101, 252
220 DATA 133, 252, 165, 253, 105, 0, 133, 253, 24, 165, 250, 41, 248, 101, 252, 133
230 DATA 252, 165, 251, 101, 253, 133, 253, 24, 169, 32, 101, 253, 133, 253, 165, 250
240 DATA 41, 7, 73, 7, 170, 169, 1, 202, 48, 3, 10, 208, 250, 160, 0, 166
250 DATA 150, 240, 5, 17, 252, 145, 252, 96, 73, 255, 49, 252, 145, 252, 96, 120
260 DATA 32, 253, 174, 32, 158, 183, 134, 150, 32, 253, 174, 32, 235, 183, 134, 167
270 DATA 165, 20, 133, 250, 165, 21, 133, 251, 32, 253, 174, 32, 235, 183, 134, 168
280 DATA 165, 20, 133, 178, 165, 21, 133, 179, 197, 251, 240, 29, 176, 33, 166, 179
290 DATA 165, 251, 133, 179, 134, 251, 166, 178, 165, 250, 133, 178, 134, 250, 166, 167
300 DATA 165, 168, 133, 167, 134, 168, 76, 31, 193, 165, 178, 197, 250, 144, 223, 169
310 DATA 0, 133, 166, 165, 168, 197, 167, 208, 26, 166, 167, 32, 94, 192, 230, 250
320 DATA 208, 2, 230, 251, 165, 179, 197, 251, 208, 239, 165, 178, 197, 250, 208, 233
330 DATA 76, 50, 194, 176, 12, 230, 166, 56, 165, 167, 229, 168, 133, 169, 76, 90
340 DATA 193, 56, 229, 167, 133, 169, 201, 1, 240, 207, 56, 165, 178, 229, 250, 133
350 DATA 170, 165, 179, 229, 251, 133, 171, 208, 106, 165, 170, 208, 2, 230, 170, 197
360 DATA 169, 176, 96, 160, 0, 166, 169, 200, 56, 138, 229, 170, 170, 197, 170, 176
370 DATA 246, 132, 178, 132, 176, 134, 168, 134, 171, 165, 170, 133, 179, 166, 167, 32
380 DATA 94, 192, 165, 166, 240, 5, 198, 167, 76, 157, 193, 230, 167, 198, 178, 208
390 DATA 236, 230, 250, 208, 2, 230, 251, 165, 176, 133, 178, 24, 165, 168, 101, 171
400 DATA 133, 168, 197, 170, 144, 9, 56, 165, 168, 229, 170, 133, 168, 230, 178, 198
410 DATA 179, 208, 202, 56, 165, 250, 233, 1, 133, 250, 165, 251, 233, 0, 133, 251
420 DATA 76, 50, 194, 160, 0, 166, 170, 200, 56, 138, 229, 169, 170, 176, 5, 198
430 DATA 171, 208, 9, 138, 197, 169, 176, 239, 165, 171, 208, 235, 132, 178, 132, 170
440 DATA 134, 168, 134, 171, 165, 169, 133, 179, 166, 167, 32, 94, 192, 230, 250, 208
450 DATA 2, 230, 251, 198, 178, 208, 241, 165, 166, 240, 5, 198, 167, 76, 18, 194
460 DATA 230, 167, 165, 170, 133, 178, 24, 165, 168, 101, 171, 133, 168, 176, 4, 197
470 DATA 169, 144, 9, 56, 165, 168, 229, 169, 133, 168, 230, 178, 198, 179, 208, 200
480 DATA 198, 167, 166, 167, 32, 94, 192, 88, 96, 224, 20, 208, 1, 96, 76, 139
490 DATA 227, 32, 253, 174, 32, 235, 183, 134, 167, 165, 20, 133, 168, 165, 21, 133
500 DATA 169, 32, 253, 174, 32, 158, 183, 134, 176, 32, 253, 174, 32, 158, 183, 134
510 DATA 177, 169, 1, 133, 150, 56, 165, 167, 229, 177, 144, 5, 133, 167, 76, 101
520 DATA 194, 56, 165, 168, 229, 176, 144, 11, 133, 168, 165, 169, 233, 0, 133, 169
530 DATA 76, 113, 194, 166, 169, 208, 241, 165, 167, 133, 178, 165, 168, 133, 250, 165
540 DATA 169, 133, 251, 166, 178, 32, 94, 192, 165, 250, 101, 176, 133, 250, 165, 251
550 DATA 105, 0, 133, 251, 240, 237, 165, 250, 201, 64, 144, 231, 24, 165, 178, 101
560 DATA 177, 133, 178, 201, 200, 144, 212, 96, 169, 4, 133, 186, 169, 126, 133, 184
570 DATA 169, 0, 133, 250, 133, 176, 133, 170, 133, 178, 133, 183, 133, 185, 32, 192
580 DATA 255, 166, 184, 32, 201, 255, 169, 8, 32, 210, 255, 169, 32, 133, 251, 133
590 DATA 177, 169, 196, 133, 179, 169, 28, 133, 159, 165, 159, 201, 20, 208, 11, 169
600 DATA 0, 133, 178, 169, 196, 133, 179, 76, 2, 195, 201, 12, 240, 241, 201, 4
610 DATA 240, 237, 169, 128, 133, 2, 169, 64, 133, 167, 169, 1, 133, 168, 169, 0
620 DATA 133, 171, 168, 169, 1, 133, 150, 162, 7, 228, 170, 208, 13, 24, 165, 250
630 DATA 105, 56, 133, 250, 165, 251, 105, 1, 133, 251, 177, 250, 37, 2, 240, 6
640 DATA 165, 171, 5, 150, 133, 171, 165, 150, 10, 201, 128, 208, 2, 169, 1, 133
650 DATA 150, 230, 250, 208, 2, 230, 251, 202, 208, 207, 165, 171, 9, 128, 32, 210
660 DATA 255, 56, 165, 250, 233, 7, 133, 250, 165, 251, 233, 0, 133, 251, 165, 2

```

```

670 DATA 74,208,15,24,165,250,105,8,133,250,165,251,105,0,133,251
680 DATA 169,128,133,2,165,170,240,13,56,165,250,233,56,133,250,165
690 DATA 251,233,1,133,251,198,167,208,133,198,168,240,129,169,13,32
700 DATA 210,255,160,0,177,178,208,20,24,165,176,105,63,133,176,133
710 DATA 250,165,177,105,1,133,177,133,251,76,187,195,24,101,176,133
720 DATA 176,133,250,165,177,105,0,133,177,133,251,200,177,178,133,170
730 DATA 24,165,178,105,2,133,178,198,159,240,3,76,233,194,32,204
740 DATA 255,169,126,76,195,255,32,253,174,32,212,225,162,0,134,253
750 DATA 160,64,169,32,133,254,169,253,133,185,76,216,255,32,253,174
760 DATA 32,212,225,169,1,133,185,169,0,76,213,255,234,0,255,255
770 DATA 7,6,0,5,0,4,0,3,0,2,0,1,0,0,0,0
780 DATA 162,32,169,160,160,0,132,34,133,35,177,34,145,34,200,208
790 DATA 249,230,35,202,208,244,169,96,141,20,167,169,234,141,187,166
800 DATA 141,188,166,169,65,141,66,160,169,196,141,67,160,169,54,133
810 DATA 1,96,165,122,72,165,123,72,32,121,0,32,156,166,32,215
820 DATA 170,104,133,123,104,133,122,32,248,168,76,121,0,234,145,0
830 DATA 146,0,147,0,148,0,149,0,150

```

READY.

Texas

für den VC-20 + 16K

TEXAS ist fast ausschließlich ein Textadventure. Außer dem Anfang und ein paar Grafikzeichen wurde auf Grafik und Ton verzichtet, um soviel als möglich Spielvarianten aufzunehmen. Aus speicherkritischen Gründen wurde auf REM-Zellen zur Auflockerung ebenfalls verzichtet.

Am Beginn wählt der Spieler ein beliebiges Ölfeld, dessen Wert vom Startkapital abgezogen wird. Dann geht es ins eigentliche Spiel. Der Spieler muß versuchen, so gut wie möglich zu wirtschaften um nach Ende der Spielzeit Texas so weit wie möglich unter Kontrolle zu haben. Um Profit zu erzielen ist es möglich, Öl billig zu kaufen und teuer zu verkaufen oder Tanker legal und illegal auf die Reise zu schicken. Andererseits lauern auch Gefahren wie z.B.:

Feuer vernichtet einen Teil Ihrer Förderanlagen. Um größeren Schaden zu vermeiden, werden sofort 150.000 Dollar benötigt.

Am Ende des Spiels wartet der Kommentar auf den Spieler. Der Kommentar wurde aus Platzgründen ebenfalls recht kurz gehalten.

Variablenübersicht:

Strings:

\$: Übernimmt die Wahl des Ölfeldes
A1\$: Ist die von FA zufällig bestimmte Firma
A2\$: Fungiert als Platzhalter für zwei verschiedene Strings
A3\$: Fungiert wie A2\$ für verschiedene Strings.
C\$: Dient als generelle INPUT/GET-Variable.
H\$: Hilft einen String in einen numerischen Wert umzusetzen
K\$: INPUT-Variable zur Eingabe eines Kontinents
TI\$: Spielzeit
XS: Enthält eine bestimmte Anzahl an Cursor-Steuerzeichen
YS: Enthält die Positionen zur Einfärbung der Ölfelder
Z\$: Siehe XS.

Dimensionierte Variable:

G\$(x): Zählt die Ölfelder, welche bereits belegt sind.
O\$(x): Übernimmt den jeweiligen Bestimmungsort der vier Tanker.
T\$(x): Dient dem Einfärben der belegten Ölfelder.
TA(x): Enthält die Ankunftszeit des Tankers am Bestimmungsort.

TM(x): Enthält die Transportmenge des jeweiligen Tankers.

L(x): Überprüft, ob der Tanker legal oder illegal fährt.

INPUT/GET-Variablen:

D: Dient generell zur Eingabe numerischer Werte
LM: Eingabe der "Lagermenge"
LR: Eingabe der "Leistungsreduktion"
LS: Eingabe der "Leistungssteigerung"
OM: Eingabe der einzulagernden oder zu verkaufenden "Ölmenge".

Zufallsvariable:

AG: Kaufangebot der Firma in A1\$ für das Öl des Spielers
BK: Der ständig steigende und fallende Barreilkurs
FA: Wählt die Firma in A1\$ aus.
G: Bestimmt die auf dem angewählten Ölfeld liegende Ölmenge.
UF: Stellt fest, ob ein sich unterwegs befindender Tanker einen Unfall hat.
W: Bestimmt den Kaufwert des angewählten Ölfeldes.
ZK: Ermittelt zusätzlich entstehende Kosten.

Hilfsvariable:

A: Siehe H\$
B: Siehe H\$

DC: Hilft "WO" einen sich alle 5 "Wochen" wiederholenden Vorgang einzuleiten.

E: Hilft beim Einfärben der Ölfelder als "POKE-Variable".

L: Erkennt ob ein falscher Bestimmungsort/Kontinent eingegeben wurde.

PS: Überprüft wieviele Tanker unterwegs sind.

RR: Gibt eine gewisse Anzahl an Raffinerien zum Kauf frei.

T1: Gibt an, welche Ölmenge zusätzlich raffiniert werden kann.

T2: Gibt die Lagerkapazität der Raffinerielager (RL) an.

TD: Gibt eine gewisse Anzahl an Tankern zum Kauf frei.

TT: Gibt eine gewisse Anzahl an Tanks zum Kauf frei.

Laufvariable:

F: Enthält die Anzahl der belegten Ölfelder.

I: Dient generell als Zeitschleife

O: Zählt die bereits eingefärbten Positionen

U: Holt zusammen mit MID\$ die Positionen aus Y\$ z.B.: MID\$(Y\$,U,4).

X: Dient der Abfrage verschiedener Bedingungen.

Y: Enthält die Nummer eines gerade abgefertigten Tankers.

Hauptvariable:

BP: Enthält den Barreelpreis für angekommene Tanker (veränderlich)

C: Hintergrund/Rahmenfarbe 36879 (fester Wert)

FL: Ist die augenblickliche Förderleistung (veränderlich)

GO: Ist eine verfügbare, nicht eingelagerte Ölmenge (veränderlich)

H: Anzahl der Helikopter (fester Wert)

J: Anzahl der Jets (fester Wert)

Ja: Zählt die im Spiel vergangene Zeit in Jahren (veränderlich)

K: Kapital des Spielers (veränderlich)

LK: Lagerkapazität (veränderlich)

LO: Lagerölbestand (veränderlich)

R: Anzahl der Raffinerien (veränderlich)

RL: Raffinerielager (veränderlich)

RO: Lager für raffiniertes Öl (veränderlich)

RZ: Reisezeit der Tanker (veränderlich)

S1: Tongenerator 1 (36874) fester Wert

S3: Tongenerator 3 (36876) fester Wert.

T: Anzahl der Tanks (veränderlich)

TA: Anzahl der Tanker (veränderlich)

V: Versicherung der Tanker (veränderlich)

WFL: Wöchentliche Förderleistung (fester Wert)

```

1 PRINT " "
2 POKE 36879, 110
3 PRINT TAB(5) " Teufelskessel BITTE WARTEN"
4 FOR X=0 TO 2047: POKE X+5120, PEEK(32768+X): NEXT
5 POKE 36869, 205
6 FOR I=5120 TO 5127
7 POKE I, 0
8 NEXT I
9 FOR I=5632 TO 6487
10 READ D
11 IF D=-1 THEN 1000
12 POKE I, D
13 NEXT I
20 DATA 0,0,0,0,0,0,0,0
30 DATA 255,255,255,255,255,255,255,255
40 DATA 255,0,0,0,0,0,0,0
45 DATA 255,255,0,0,0,0,0,0
50 DATA 128,128,128,128,128,128,128,128
60 DATA 255,128,128,128,128,128,128,128
70 DATA 192,192,192,192,192,192,192,192
80 DATA 255,192,192,192,192,192,192,192
90 DATA 255,255,192,192,192,192,192,192
100 DATA 255,255,128,128,128,128,128,128
110 DATA 255,255,3,3,3,3,3,3
115 DATA 3,3,3,3,3,3,3,3
120 DATA 255,3,3,3,1,0,0,0
130 DATA 0,0,0,128,252,252,135,135
140 DATA 0,0,0,3,15,124,240,128
150 DATA 131,128,128,128,128,128,128,128
160 DATA 0,0,0,128,224,248,191,135
170 DATA 193,243,62,12,0,0,0,0
180 DATA 224,128,128,128,128,128,128,128
190 DATA 0,0,0,0,0,15,61,96
200 DATA 0,0,0,0,0,128,224,56
210 DATA 24,12,7,1,0,0,0,0
220 DATA 0,0,255,255,131,131,131,131
230 DATA 131,131,131,131,131,131,131,131
240 DATA 255,131,131,129,129,128,128,128
250 DATA 0,0,0,0,128,192,96,96
260 DATA 96,96,48,48,24,12,6,6
270 DATA 6,6,3,3,3,6,6,3
280 DATA 255,6,6,12,12,6,254,255
290 DATA 255,128,128,128,128,128,128,129
300 DATA 131,134,134,140,152,176,224,192
310 DATA 3,14,24,112,192,0,0,0
320 DATA 128,128,128,128,131,135,188,224
330 DATA 255,14,56,96,96,192,192,128
340 DATA 131,134,140,152,152,176,176,176
350 DATA 176,176,152,152,152,152,140,140
360 DATA 252,140,140,140,140,152,152,152
370 DATA 152,152,152,216,112,0,0,0
380 DATA 192,240,63,15,0,0,0,0
390 DATA 3,0,0,0,0,0,0,0
400 DATA 255,0,0,224,56,12,12,6
410 DATA 31,12,7,0,0,0,0,0
420 DATA 128,192,96,56,24,24,24,24
430 DATA 128,128,192,96,48,24,14,7
440 DATA 127,48,48,24,24,12,12,6
450 DATA 192,192,96,96,96,96,96,96
460 DATA 192,224,48,28,12,6,6,3
470 DATA 3,6,12,12,120,192,0,0
480 DATA 128,128,128,128,128,131,158,240

```

W0: Zählt die im Spiel vergangene Zeit in Wochen (veränderlich)
Z1: Zustand der Pumpen in % (veränderlich)
Z2: Zustand der Quellen in % (veränderlich)
Z3: Zustand der Bohrmeißel in % (veränderlich)

Unterprogramme:

90-99 Schlußkommentar
100-139: Ölverkauf
140-150: Wahl der Firmen
200-299: Öl einlagern
300-349: Öl raffinieren
350-354: Raffinerie überlastet
355-360: Lage für raffiniertes Öl überlastet
380-389: Unterprogramm zur Firmenwahl
390-399: Zusätzliche Kosten entstehen
400-430: Öl legal verschiffen
450-470: Öl illegal verschiffen
480-489: Tanker angekommen
490-494: Legaler Tanker gesunken
495-499: Illegaler Tanker gestoppt
500-599: Förderleistung regulieren
560: Zeitschleife
600-699: Ankauf verschiedener Artikel
700-719: Quellenbrand
720-739: Pumpen defekt
740-759: Bohrmeißel gebrochen
760-779: Jet abgestürzt
780: Helikopter abgestürzt
785-799: Zustand der Pumpen, Quellen und Bohrmeißel
800-899: Bericht 1
900-949: Bericht 2
950-959: Geologisches Gutachten
960-964: Barrelkursbestimmung
965-969: Kapital zu gering
970-999: Kontrolle für Woche, Jahr und Tanker
3600-3620: Hauptbild
4000-4018: Ölfelder zeichnen
4020-4200: Eingabe und Einfärben des Ölfeldes
4500-4600: Variablendefinition

Ladevorgang:

Vor dem Laden des Vorprogrammes muß POKE44,28:POKE7168,0 eingegeben werden. Nach dem Durchlauf des Vorprogrammes meldet der Computer, daß er fertig ist. Jetzt steht dem Laden des Hauptprogrammes nichts mehr im Wege.

```

490 DATA224,192,0,0,0,0,0
500 DATA96,49,27,14,0,0,0
510 DATA192,192,192,192,192,224,96,96
520 DATA255,0,0,0,0,0,128,128
530 DATA3,3,3,3,3,1,1,1
535 DATA12,6,6,6,6,3,3,3
540 DATA96,48,24,24,24,12,12,12
550 DATA255,255,0,0,0,0,128,192
560 DATA255,255,224,56,12,6,3,0
570 DATA255,192,128,128,128,128,128,128
580 DATA0,0,0,16,0,0,0,0
590 DATA0,0,0,24,0,0,0,0
600 DATA0,0,0,60,0,0,0,0
610 DATA0,0,0,255,255,0,0,0
620 DATA0,0,0,45,45,0,0,0
630 DATA0,0,0,182,182,0,0,0
640 DATA0,0,237,72,77,0,0,0
650 DATA0,0,91,155,91,0,0,0
660 DATA0,251,34,35,34,35,0,0
670 DATA0,162,20,8,20,162,0,0
680 DATA0,99,148,242,145,150,0,0
690 DATA0,0,255,255,24,24,24,24
695 DATA24,24,24,24,24,24,0,0
700 DATA0,0,127,127,96,96,96,124
710 DATA124,96,96,96,127,127,0,0
720 DATA0,0,65,65,65,34,34,28
730 DATA28,34,34,65,65,65,0,0
740 DATA0,0,28,62,99,99,99,127
742 DATA127,99,99,99,99,99,0,0
744 DATA0,0,31,63,96,96,96,126
746 DATA63,3,3,3,126,124,0,0
750 DATA255,126,126,126,255,126,126,255
760 DATA73,127,127,127,127,127,127,73
770 DATA0,0,0,1,1,1,3,3
780 DATA0,128,128,64,192,64,224,96
790 DATA2,5,7,5,12,9,10,12
800 DATA160,80,240,80,24,72,40,24
810 DATA0,0,56,56,127,255,127,0
820 DATA0,0,0,0,255,255,255,0
830 DATA0,0,0,0,254,252,248,0
840 DATA255,130,135,134,135,130,135,255
850 DATA255,129,193,129,193,193,193,255
860 DATA128,255,128,255,128,255,0,0
870 DATA1,255,1,255,1,255,0,0
880 DATA0,7,31,63,63,127,127,127
890 DATA56,212,242,254,249,255,253,253
900 DATA134,166,166,134,255,255,255,255
910 DATA223,221,221,199,253,253,255,253
920 DATA0,0,0,1,7,217,99,51
925 DATA255,255,255,255,255,255,255,255
930 DATA48,24,108,132,4,4,142,142
940 DATA23,23,23,62,62,62,54,34
950 DATA255,0,15,18,127,127,34,31
960 DATA255,128,240,191,255,254,32,252
975 DATA160,20,12,252,252,2,1,0
980 DATA96,120,126,127,253,255,0,1
990 DATA0,0,0,255,85,255,192,224
995 DATA0,0,0,224,56,255,16,56
999 DATA-1
1000 PRINT"┐":PRINT"DER NEUE ZEICHENSATZ"
1001 PRINT"┐IST FERTIG.SIE KOENNEN"
1002 PRINT"JETZT DAS HAUPTPRO-"
1003 PRINT"GRAMM(TEXAS 6) LADEN."
1004 PRINT"┐VIEL SPASS."

```

```

0 POKE36869,205:GOTO5
1 ONBOSUB4078,4079,4080,4081,4082,4083,4084,4085,4086,4087,4088,4089,4090
2 RETURN
5 REM C-BY JOERG STUMPF/6966 WOLFERSHEIM-4/1984
10 DIMT$(6),G$(13),TA(4),TM(4),L(4),O$(4):GOTO20.
20 GOSUB4500:GOSUB360:POKEC,28:POKEL,15
35 PRINT$:"L":GOSUB50
36 PRINT$:"L":GOSUB50
37 PRINT$:"L":GOSUB50
38 PRINT$:"L":GOSUB50
39 PRINT$:"L":GOSUB50
40 PRINT$:"L":GOSUB50
41 PRINT$:"L":GOSUB50
42 PRINT$:"TEXAS":GOSUB50
45 PRINT$:"HJLN":GOSUB50
46 PRINT$:"IKMO":GOTO55
50 POKES3,200:FORI=1TO100:NEXT:POKES3,0:FORI=1TO200:NEXT:RETURN
55 FORI=15TO0STEP-.05:POKEL,I:POKES1,130:NEXTI:POKES1,0:FORI=1TO3000:NEXT
60 GOTO4000
70 GOSUB350:GOTO800
85 GOTO3600
90 PRINT$:"***** KOMMENTAR *****":IFK>5000000THEN33
91 PRINT$:"SIE SIND EIN MISERAB- LER OELBARON".WERDEN SIE LIEBER STRASSENFE-ER.
92 GOTO38
93 IFK>5000000THEN35
94 PRINT$:"ICH WUERDE SIE NICHT EINSTELLEN!":GOTO38
95 IFK>3000000THEN37
96 PRINT$:"NICHT SCHLECHT ABER B.CARRINGTON SIND SIE UNTERLEGEN!":GOTO38
97 PRINT$:"BRAVO!SIE KOENNTEN J.R. DAS WASSER REICHEN!"
98 PRINT$:"BIS BALD..."
99 GOTO39
100 GOSUB370:GOSUB360:POKEC,170:PRINT$:"***** OELVERKAUF *****"
101 PRINT$:"BARRELKURS: BK:PRINT$:"SIE BESITZEN"LO " BARREL LAGEROEL."
102 PRINT$:"SIE BESITZEN"GO:PRINT$:"BARREL FOERDEROEL."
103 PRINT$:"G ODER L DRUECKEN!"
104 GETC$:IFC$<"L"ANDC$<"G"THEN104
105 IFC$="L"THEN115
106 GOTO130
110 INPUT$:"WELCHE MENGE OEL":OM
111 RETURN
114 PRINT$:"ZU WENIG LAGEROEL!":GOSUB560:RETURN
115 GOSUB110:IFOM>LOTHENGOSUB114
116 GOSUB140
117 INPUT$:"E INVERSTANDEN":C$
118 IFC$<"J"ANDC$<"N"THEN117
119 IFC$="J"THENLO=LO-OM:K=K+OM*BK:GOTO85
120 GOTO85
129 PRINT$:"IHRE FOERDERLEISTUNG, BETRAEGT NUR"FL:PRINT$:"BARREL!!!":GOSUB560
130 GOSUB110
131 IFOM>FLTHEN129
132 GOSUB140
133 INPUT$:"E INVERSTANDEN":C$
134 IFC$<"J"ANDC$<"N"THEN133
135 IFC$="J"THENK=K+OM*BK:GO=GO-OM:GOTO85
136 GOTO85
140 IFBK>35.20THENPRINT$:"KEINE FIRMA MOECHTE IHR OEL KAUFEN.":GOSUB560:GOTO85
142 FA=INT(RND(1)*5)
143 ONFAOSUB145,146,147,148,149:GOTO150
145 A1$="WESTSTAROIL":RETURN
146 A1$="SCOTTISH PETROL":RETURN
147 A1$="WENTWORTH TOOL&DI":RETURN
148 A1$="PARAL":RETURN
149 A1$="EWING OIL":RETURN
150 PRINT$:"A1$:PRINT$:"MOECHTE IHR OEL KAUFEN":RETURN
200 GOSUB370:POKEC,138:PRINT$:"*** OEL EINLAGERN ***":LK=T*10000
201 PRINT$:"IHRE LAGERKAPAZITAET, ":PRINT$:"BETRAEGT 2.2. "LK:PRINT$:"BARREL."
202 PRINT$:"S SIND BEFEITS"LO:PRINT$:"BARREL OEL EINGELAGERT"
203 PRINT$:"WIEVIEL WOLLEN SIE ":PRINT$:"EINLAGERN"
204 INPUTLM
205 IFLM+LO>LKTHENPRINT$:"LAGERKAPAZITAET UEBER-SCHRITTEN.":GOSUB560:GOTO200
206 LO=LO+LM:O=O+LM:GOTO85
300 GOSUB370:POKEC,235:PRINT$:"***** RAFFINERIE *****"
302 PRINT$:"TANKS FUER VERARBEITUNGSAEHRIGES OEL RL:PRINT$:"BARREL."
303 PRINT$:"TANKS FUER RAFFINER- IERTES OEL"RO:PRINT$:"BARREL."
304 PRINT$:"SPC(7)":RETURN<
305 GETC$:IFC$<"CHR$(13)"THEN305
306 T1=T+20000:PRINT$:"SIE KOENNEN ZUSAEZL T1"BARREL":PRINT$:"VERARBEITEN."
307 GOSUB110
308 IFOM>T1THENGOSUB350:GOTO307
309 RO=RO+OM:RL=RL-OM
310 IFRO>T2THENGOSUB355
311 INPUT$:"WOLLEN SIE IHR OEL VERKAUFEN":C$
312 IFC$<"J"ANDC$<"N"THEN311
313 IFC$="N"THEN85
314 GOSUB110:IFOM>ROTHEN340
316 GOSUB330:PRINT$:"A1$

```



```

317 PRINT "MOECHTE IHR OEL FUER "AG"$ BARREL KAU- FEN."
318 INPUT "NEHMEN SIE DAS ANGE- BOT AN";C$
319 IFC$<"J"ANDC$<"N"THEN318
320 IFC$="J"THENK=K+OM*AG:RO=RO-OM:GOTO85
321 GOTO85
340 PRINT "SIE BESITZEN NUR":PRINTRO"BARREL":PRINT "RAFFINIERTES OEL!!"
341 GOSUB560:GOTO313
350 PRINT "J+K+PQ+ ":GOSUB390
351 PRINT "RAFFINERIE UEBERLASTETES ENSTEHEN KOSTEN IN HOEHE VON"ZK"$."
352 K=K-ZK:GOSUB560:RETURN
355 PRINT "J+K+PQ+ ":GOSUB390
356 PRINT "LAAGER FUER RAFFINER- IERTES OEL UEBERLASTETES ENTSTEHEN KOSTEN IN"
357 PRINT "HOEHE VON"ZK"$."K=K-ZK:GOSUB560:RETURN
380 IFC$>35.20THENPRINT "KEINE FIRMA MOECHTE IHR OEL KAUFEN.":GOSUB560:GOTO85
381 FA=INT(RND(1)*5)
382 ONFAGOSUB145,146,147,148,149
383 AG=INT(RND(1)*5)+60:RETURN
390 ZK=INT(RND(1)*20000/1000)*1000+200000:RETURN
400 GOSUB970:POKEC,238:PRINT "OEL VERSCHIFFEN **:IFY)TATHENY=1
401 PRINT "SIE BESITZEN 2.2."TA" TANKER.":PRINT "WOLLEN SIE LEGAL O. ILLEGA
L OEL"
402 PRINT "VERSCHIFFEN ?":PRINT "L ODER I DRUECKEN!"
403 PS=0:FORX=1TO4:IFTA(X)>1THENPS=PS+1
404 IFY=XANDTA(X)>1THENY=Y+1
405 NEXTX:IFPS=TATHEN440
406 GETC$:IFC$<"L"ANDC$<"I"THEN403
407 IFC$="I"THEN450
408 PRINT "PS" TANKER UNTERWEGS."
409 L(Y)=0:PRINT "WOHIN SOLL DER TANKER FAHREN":PRINT "EINGABEFORM:"
410 INPUT "BEST.-ORT--KONTINENT";O$(Y),K$:L=ASC(LEFT$(K$,1))-64
411 GOSUB430
412 INPUT "WELCHE TRANSPORTMENGE (MAX.100000 BARREL)":TM(Y):V=TM(Y)*.4
413 IFTM(Y)>100000THENPRINT "SIE KOENNEN DEN TANKER NICHT UEBERLADEN!":GOSUB560:
GOTO412
414 IFTM(Y)>LOTHENGOSUB114:GOTO85
415 PRINT "BEST.-ORT:"O$(Y):PRINT "IN:"K$
416 PRINT "TRANSPORTMENGE:"TM(Y):PRINT "VERSICHERUNG:"V:PRINT "REISEZEIT:"RZ" WO
HEN."
417 TA(Y)=WO+RZ:PRINT "ANKUNFTSWOCHETA(Y):K=K+V:LO=LO-TM(Y)
421 Y=Y+1:GOSUB560:GOTO85
430 IFL=1THENRZ=5:RETURN
431 IFL=5THENRZ=4:RETURN
432 IFL=14ORL=19THENRZ=3:RETURN
440 PRINT "SIE KOENNEN KEIN OEL VERSCHIFFEN!":GOSUB560:GOTO85
450 POKEC,8:PRINT "ILLEGALES GESCHAEFT*"
453 PRINTPS" TANKER UNTERWEGS!":PRINT "DIE CHANCEN FUER EIN GELINGEN IHRER"
454 PRINT "AKTION" STEHEN 1:5. MIT WELCHEM EMBARGO- LAND WOLLEN SIE DAS
"
455 PRINT "GESCHAEFT MACHEN KUBA, IRAN":PRINT "USSR, AFGHANISTAN"
460 INPUTO$(Y)
461 L=ASC(LEFT$(O$(Y),1))-64:IFL<1ANDL<9ANDL<11ANDL<21THEN460
462 RZ=4:K$="X":L(Y)=1:PRINT "GOTO412
480 IFL(X)=1THENBP=(BK+10)*TM(X):GOTO487
484 BP=(BK+2)*TM(X)
487 PRINT "ANGEKOMMEN.D. OEL WIRD FUER BP"$:PRINT "VERKAUFT.":K=K+BP:GOSUB560:R
ETURN
490 PRINT "IHR TANKER IST UNTER- WEGS IN EINEN STURM GERATEN UND GESUNKEN."
491 TA=TA-1:TA(X)=1:L(X)=2:GOSUB560:GOTO395
495 PRINT "IHR ILLEGALES GE- SCHAEFT MIT "O$(X):PRINT "IST GESCHEITERT."
496 PRINT "ALS STRAFE FUER SIE WURDEN 5 MILL. DOLLAR FESTGELEGT."
497 K=K-5000000:L(X)=2:TA(X)=1:GOSUB560:GOTO395
500 GOSUB970:POKEC,61:PRINT "FOERDERLEISTUNG **"
502 PRINT "IHRE FOERDERLEISTUNG, BETRAEGT 2.2. "FL
503 PRINT "BARREL PRO WOCHE."
504 PRINT "WOLLEN SIE SIE STEI- GERN ODER REDUZIEREN?":PRINT "S ODER R DRUECKE
NI"
505 GETC$:IFC$<"S"ANDC$<"R"THEN505
506 IFC$="R"THEN530
510 PRINT "SIE KOENNEN HOECHSTENS WFL*T"BARREL PRO":PRINT "WOCHE FOERDERN."
511 INPUT "NUM WIEVIEL WOLLEN SIE STEIGERN.":LS
512 IFFL+LS>WFL*TTTHENGOSUB525:GOTO514
513 FL=FL+LS
514 UF=INT(RND(1)*3)+1
515 ONUFGOTO516,517,518,517,519,517,520,517,521
516 GOSUB700
517 GOTO85
518 GOSUB720
519 GOSUB740
520 GOSUB760
521 GOSUB780
522 GOTO517
525 PRINT "LEISTUNGSKAPAZITAET UEBERSCHRITTEN.":GOSUB390
526 PRINT "ES ENTSTEHEN ZUSAETZ- LICHE KOSTEN IN HOEHE"
527 PRINT "VON"ZK"$."K=K-ZK:GOSUB560:RETURN
530 PRINT "SIE MUESSEN MIND. WFL":PRINTFL/5"BARREL":PRINT "PRO WOCHE FOERDERN."
531 INPUT "NUM WIEVIEL WOLLEN SIE REDUZIEREN":LR
532 IFFL-LR<FL/5THENGOSUB540

```

programme

```

533 FL=FL-LR:GOTO514
540 PRINT"QUELQUELLEN NICHT AUS- BELASTET.ES ENTSTEHEN KOSTEN IN HOEHE":GOSUB
390
541 PRINT"VON"ZK*$.":K=K-ZK:GOSUB560:RETURN
560 FORI=1TO8000:NEXT:RETURN
600 GOSUB370:POKEC,221:PRINT"***** ANKAUF *****"
601 PRINT"ZUR ERWEITERUNG IHRER ANLAGEN STEHEN ZUR VERFUEGUNG:"
602 PRINT"ART. ANZ."
603 PRINT"1 TANKS "TT
604 PRINT"2 TANKER "TD
605 PRINT"3 RAFFINERIE "RR
606 PRINT"4 ROHOEL "7"
607 PRINT"5 OELFELDER"
608 PRINT"WAS WOLLEN SIE KAUFEN?(NUMMERN)"
609 GETD:IFD<1ORD>5THEN609
610 ONDGO TO620,630,640,650,660
620 IFTT=0THEN690
621 PRINT"JEDER TANK KOSTET 500000 $."
622 PRINT"WIEVIEL VON DEN TANKS WOLLEN SIE KAUFEN?"
623 INPUTD:IFD>TTTHEN623
624 IFK<D*500000THENGOSUB365:GOTO85
625 K=K-D*500000:T=T+D:TT=TT-D:GOTO85
630 IFTD=0THEN690
631 PRINT"JEDER TANKER KOSTET 2 MILL. $."
632 PRINT"WIEVIEL VON D. TANKERN WOLLEN SIE KAUFEN?"
633 INPUTD:IFD>TDTHEN633
634 IFD*2000000>KTHENGOSUB365:GOTO85
635 K=K-D*2000000:TA=TA+D:TD=TD-D:GOTO85
640 IFRR=0THEN690
641 INPUT"DIE RAFFINERIE KOSTET 5 MILL. $. WOLLEN SIE SIE KAUFEN?":C$
642 IFC$<>"J"THEN615
643 IFK<4000000THENGOSUB365:GOTO85
644 K=K-4000000:R=R+1:RR=0:GOTO85
650 GOSUB381:PRINT"1"
651 PRINT"IST BEREIT IHNEN OEL ZUM DERZEITIGEN BAR- RELKURS VON "BK
652 PRINT"ZU VERKAUFEN.WIEVIEL WOLLEN SIE NEHMEN"
653 INPUTD:IFD*BK>KTHENGOSUB365:GOTO653
654 IFLO+D>T*100000THENPRINT"SOVIEL KOENNEN SIE NICHT EINLAGERN!":GOTO653
655 PRINT"DAS ERWORBENE OEL WIRD EINGELAGERT!"
656 K=K-D*BK:LO=LO+D:GOSUB560:GOTO85
660 IFD<50000THEN4000
661 PRINT"SO LANGE AUF IHREM ALTEN BOHRFELD NOCH OEL GEBOHRT WIRD,GIBT "
662 PRINT"D.REGIERUNG KEIN NEUES BOHRFELD ZUM KAUF FREI":GOSUB560:GOTO85
690 PRINT"NICHT VORHANDEN!":GOSUB560:GOTO85
700 POKEC,24:PRINT"1"
710 PRINT"FEUER VERNICHTET EINEN TEIL IHRER FOERDERAN- LAGEN.UM GROESSEREN"
711 PRINT"SCHADEN ZU":GOSUB390:PRINT"VERHINDERN,WERDEN SO- FORT"ZK*2*$"
712 PRINT"VERBRAUCHT."K=K-ZK*2:GOSUB560:RETURN
720 POKEC,30:PRINT"2"
721 PRINT"IN TEIL IHRER PUMPEN IST AUSGEFALLEN."
722 GOSUB390:PRINT"ZU DIESEM ZWECK,WERDEN SOFORT"ZK
723 PRINT"BENOETIGT."K=K-ZK:GOSUB560:RETURN
740 POKEC,238:PRINT"3"
741 PRINT"BEI EINER PROBEBOHRUNG IST DER BOHRMEISEL GE- BROCHEN.ZUR BEREIT-"
742 PRINT"STELLUNG EINES NEUEN BOHRERS WERDEN 80000 $ BENOETIGT."
743 K=K-80000:GOSUB560:GOTO517
760 A2$="JET":PRINT"4"
761 POKEC,122:PRINT"5 AUF DEM WEG,IHRE AN- WEISUNGEN AUSZUFUEHREN"
762 PRINT"IST IHR GESCHAFTS- FUEHRER MIT DEM"
763 PRINTA2$ ABGESTUERZT."PRINT"ES WERDEN 200000 $ ALS ABFINDUNG FUER"
764 PRINT"DIE FAMILIE UND 500000 FUER EINEN NEUEN":PRINTA2$ BEREITGE-"
765 PRINT"STELLT."K=K-1000000:GOSUB560:GOTO517
780 A2$="HUBSCHR."PRINT"6"
785 PRINT"7"
786 PRINT"8"
787 PRINT"9"
788 PRINT"10"
790 PRINT"11"
795 PRINT"12"
800 GOSUB370:POKEC,237:PRINT"***** BERICHT1 *****:PRINTWO".WOCH "JA".JAHR."
801 PRINT"13"
802 PRINT"14"
803 PRINT"15"
804 PRINT"16"
805 PRINT"17"
806 PRINT"18"
807 PRINT"19"
808 PRINT"20"
809 FORI=1TOF
810 PRINTG$(I)"",;
811 NEXTI
812 GETC$:IFC$=""THEN812
813 GOTO85
800 GOSUB370:POKEC,170:PRINT"***** BERICHT2 *****"
901 PRINT"FOERDERL.: "FL
902 GOSUB360:PRINT"BARRELK.: "BK
903 PRINT"RAFFINERIELAGER: "RL
904 PRINT"RAFFINERIERTES OEL: "RO

```

```

905 PRINT "ZUSTAND D.FOERDERAN- LAGEN IN %."
906 PRINT "PUMPEN : "21:PRINT "QUELLEN : "22:PRINT "MEISEL : "23:GOSUB560
907 IF21<=50THENGOSUB920
908 IF22<=50THENGOSUB925
909 IF23<=50THENGOSUB930
911 GETC$:IFC$=""THEN911
912 GOTO85
920 GOSUB785:Z1=Z1+40:RETURN
925 GOSUB790:Z2=Z2+40:RETURN
930 GOSUB795:Z3=Z3+40:RETURN
950 G=INT(RND(1)*1000000)+500000:G=G
951 PRINT "*** GEOLOGISCHES ***:PRINT "***** GUTACHTEN *****"
952 PRINT "GESCHAEZTE":PRINT "OELMENGE:"
955 PRINTG"BARREL":GOSUB560:RETURN
960 BK=INT(RND(1)*5*100)/100+31:RETURN
965 PRINT "Y2 00000000"
966 PRINT "SIE BESITZEN NICHT GE-NUG KAPITAL,UM DIESE "
967 PRINT "AKTION DURCHZUFUEHREN.":GOSUB560:RETURN
970 IFT1$="020000"THEN90
972 GOSUB990
973 IFW0=52THENW0=1:JA=JA+1:DO=5
974 IFDO>W0THEN980
975 DO=DO+5:GOSUB985:Z1=Z1-5:Z2=Z2-7:Z3=Z3-10
976 LO=LO+FL/10:RL=RL+70000:G0=G0+FL/5*2
977 ONRGOSUB978,979:GOTO980
978 RL=RL-60000:RO=RO+60000:RETURN
979 RL=RL-120000:RO=RO+120000:RETURN
980 T2=R*150000:IFRO>T2THENGOSUB355
981 IFRL>T2THENGOSUB350
982 RETURN
985 PRINT "FUER BETRIEBSKOSTEN (LOEHNE/GEHAELTER U. RAFFINERIE) WERDEN "
986 PRINT "800000 $ ABGEBUCHT.":GOSUB560:K=K-800000
987 IF0<FLTHEN0=0-FL:RETURN
988 IF0<FLTHEN0=0-0
989 PRINT "IHR OELFELD IST ER- SCHOEPFT.KAUFEN SIE EIN NEUES.":GOSUB560:G0
TO4000
990 IFW0=1THENRETURN
991 UF=INT(RND(1)*5)+1:X=1
992 IFTA(X)<>1ANDUF=2ANDL(X)=0THEN490
993 IFTA(X)<>1ANDUF=2ANDL(X)=1THEN495
994 X=X+1:IFX=TATHEN995
995 FORX=1TO4
996 IFTA(X)=WOTHENTA(X)=1:PRINT "TANKER IN "0$(X):GOSUB480
997 NEXTX:RETURN
3600 GOSUB970:W0=W0+1:RL=RL+10000:POKEC,204:PRINT "***** AKTIONEN *****"
3601 PRINTW0".WOCHEN "JA".JAHR"
3602 PRINT "01 OEL VERKAUFEN"
3603 PRINT "02 OEL EINLAGERN"
3604 PRINT "03 OEL RAFFINIEREN"
3605 PRINT "04 OEL VERSCHIFFEN"
3606 PRINT "05 FOERDERMENGE"
3607 PRINT "06 ANKAUF "
3608 PRINT "07 BERICHT1"
3610 PRINT "08 BERICHT2"
3611 PRINT "09 AKTION ?"
3612 OPEN1,0:INPUT#1,D:IFD<1ORD>8THEN3612
3614 CLOSE1:ONDGOTO100,200,300,400,500,600,800,900
4000 IFF=14THENPRINT "KEIN FELD MEHR UEBRIG.":GOTO90
4001 POKEC,30:PRINT "WELCHES FELD??"
4002 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4003 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4004 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4005 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4006 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4007 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4008 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4009 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4010 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4011 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4012 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4013 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4014 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4015 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4016 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4017 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4018 PRINT "01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20"
4019 GOSUB4052
4020 PRINTX$WELCHES OELFELD
4045 INPUTA$
4046 IFA$(CHR$(65)ORA$(CHR$(77)THEN4000
4047 IFF=14THENT1$="015900"
4048 FORX=1TO12:IFA$(G$(X)THEN4045:NEXTX
4050 A=ASC(A$):H$=STR$(A):B=VAL(H$)-64
4051 GOTO4070
4052 IFF=1THENRETURN
4053 FORI=1TOF-1
4054 B=ASC(G$(I))-64:GOSUB1:GOSUB4055:NEXTI
4055 U=1

```

Achtung Programmierer!

Haben Sie eigentlich schon bedacht, daß Ihre eingesandten Programme für "Compute mit" bei uns eine viermal so große Abdruckchance haben, als bei anderen Verlagen. Warum? Weil die "Compute mit" nicht einmal – wie alle anderen Computerzeitschriften – sondern viermal im Monat, spricht wöchentlich, erscheint. Außerdem können Sie 100%ig sicher sein, daß Ihr Programm von uns nach der Qualität beurteilt und natürlich auch dementsprechend honoriert wird.


```

4056 FORO=1T06
4057 T$(O)=MID$(Y$,U,4)
4058 U=U+5
4053 NEXTO
4060 FORO=1T06
4061 E=VAL(T$(O))
4062 POKEE,65:POKEE+33792,2
4063 NEXTO
4064 RETURN
4070 W=INT(RND(1)*1300000/1000)*1000+700000
4071 PRINTX$="WERT D. FELDES: "W"$ DOLLAR.":GOSUB560
4072 PRINTX$="WOLLEN SIE ES KAUFEN":INPUTC$
4073 IFC$(">")J"ANDC$("<")N"THEN4072
4074 IFC$="N"THEN4000
4075 IFK(WTHENGOSUB365:GOTO4000
4076 G$(F)=A$:F=F+1
4077 K=K-W:GOSUB1:GOTO4094
4078 Y$="4126 4127 4148 4149 4170 4171":RETURN
4079 Y$="4192 4193 4214 4215 4236 4237":RETURN
4080 Y$="4194 4195 4216 4217 4238 4239":RETURN
4081 Y$="4256 4257 4278 4279 4300 4301":RETURN
4082 Y$="4258 4259 4280 4281 4302 4303":RETURN
4083 Y$="4260 4261 4282 4283 4304 4305":RETURN
4084 Y$="4262 4263 4284 4285 4306 4307":RETURN
4085 Y$="4264 4265 4286 4287 4308 4309":RETURN
4086 Y$="4266 4267 4288 4289 4310 4311":RETURN
4087 Y$="4326 4327 4348 4349 4370 4371":RETURN
4088 Y$="4328 4329 4350 4351 4372 4373":RETURN
4089 Y$="4330 4331 4352 4353 4374 4375":RETURN
4090 Y$="4334 4335 4416 4417 4438 4439":RETURN
4094 GOSUB4055:GOSUB560:GOTO70
4500 C=36879:L=36878:S=36876:S1=36874
4501 Z$="XXXXXXXXXXXXXXXXXXXX":X$="XXXXXXXXXXXXXXXXXXXX"
4503 T=1:H=1:J=1:K=2500000:FL=100000:Z1=85:Z2=92:Z3=98
4504 TT=5:TD=2:RR=1:TA=2:R=1:Y=1:F=1:WO=1:JA=1:DO=5:GO=10000:WFL=100000
4506 TI$="000000":FORX=1TO4:TA(X)=1:L(X)=2:NEXTX
4600 RETURN
READY.

```



SCHLUSS mit dem unproduktiven Stöbern in überall herumliegenden Programmierunterlagen.

SCHLUSS MIT FRUST UND ZEITVERLUST!

Von einigen hunderttausend VC20- und C64-Besitzern sind nur wenige bis zu Sound und Grafik vorgedrungen.

DAS SOLL SOFORT ANDERS WERDEN!

I. Dinkler, Idee-Soft, Am Schneiderhaus 7, D-5760 Amsberg 1, Tel.: 02932/32947

C-64 – OBERSEITE

50 BASIC-Begriffe. Ausgabezeichen für Zeichenfarben, F-Tasten, Cursor, INST/DEL und CLR/HOME. POKEs für Bildschirm-Rahmen, -hintergrund, -positionen, Zeichenfarben, Sound (3 Stimmen, Geräusch, 4 Oktaven). Fehler. BIN/DEZ-Umrechnung. Uhrzeit. 18 Floppy- und Druckerbefehle. Hüllkurve, Welle, Filter, Notenwerte. – Steuertastenerläuterung. Sprite-Pokes.

RÜCKSEITE: Felder, Edier- und Formationskommandos. Logische und Rechenoperationen. BIN/DEZ/HEX-Tabelle. Ungeeignete Variablen. Bildschirm-, ASCII- und CHR\$-Codes. Scrolling-POKEs. Sound- und Grafik-Demoprogramme. Sprite-Data-Tafel.

VC 20

Wie C 64 ohne die für diesen geltenden Besonderheiten. Dazu: Farbentabelle für Bildschirmrahmen und -grund. Speicherorganisation.

COMPUMask zeichnet sich aus durch guten Sitz, abrieb- und reinigungsmittelfesten Mehrfarbendruck in HI-RES unter flexiblem, reißfestem Edelkunststoff in computerdeckenden Maßen.

Erhältlich in allen Fachgeschäften zum Preis von ca. DM 30,-. Für Lieferrnachweis genügt Postkarte oder Anruf.

Cave of Death

(C) 1984 BY
TOMORROW - SOFT

für den Commodore 64

Sie haben die Aufgabe, mit Ihrem Helikopter durch ein unterirdisches Höhlensystem zu fliegen, um in das Computerzentrum zu gelangen. Die Höhle wurde dazu benutzt, radioaktives Material zu lagern.

Durch die starke Strahlung ist ein Defekt am Steuerungscomputer aufgetreten. Sie haben die Fähigkeit, diesen zu beheben.

Sie müssen sich einen Weg durch die Höhle bahnen, in dem Sie sich durch das verseuchte Material schießen! Es müssen alle Steine zerstört werden, bevor sich die Tür automatisch öffnet. Eine Be-

rührung des Gesteins oder der wild pulsierenden Energie-Laser führt zur sofortigen Zerstörung Ihres Helikopters!

Seien Sie also vorsichtig!
Viel Glück!

Nachdem man das MC-Program in Form des Data-Listings abgetippt und gespeichert hat, geht man folgend vor:

- 1) RUN
- 2) Speicherprogramm eingeben
- 3) Diskette einlegen
- 4) RUN

Nun wird das MC-Program direkt auf Diskette geladen. Das Data Listing kann entfernt werden.

Die obenstehende Lade- und Speicheranweisung gilt für die Speicherung auf Diskette. Zur Speicherung des Programmes auf einer Kassette muß in dem Speicherprogramm Zeile 14 geändert werden in:

14 OPEN1,1,1,"CAVE MC CODE"

Das Basic-Program wird wie immer abgetippt und gesaved.

```
100 FOR I= 12288 TO 13945
110 READ X:POKEI,X:GOTO X:NEXT
120 DATA 60,66,153,161,161,153,66,60,62,50,34,62
130 DATA 34,98,98,0,62,50,34,60,34,98,126,0
140 DATA 62,50,32,32,32,98,126,0,60,50,34,34
150 DATA 34,98,124,0,62,48,32,56,32,96,126,0
160 DATA 62,48,32,56,32,96,96,0,62,50,32,46
170 DATA 34,98,126,0,34,34,50,62,34,98,98,0
180 DATA 28,12,8,8,8,24,28,0,14,6,4,4
190 DATA 4,100,124,0,34,34,50,60,36,98,98,0
200 DATA 48,32,32,32,32,98,126,0,33,51,61,45
210 DATA 33,97,97,0,34,50,58,38,34,98,98,0
220 DATA 62,50,34,34,34,98,126,0,62,50,34,62
230 DATA 32,96,96,0,62,50,34,34,106,126,2
240 DATA 62,50,34,62,36,98,98,0,62,50,32,62
250 DATA 2,98,126,0,62,12,8,8,8,24,24,0
260 DATA 50,34,34,34,34,98,126,0,50,34,34,34
270 DATA 98,98,28,0,49,33,33,45,45,115,97,0
280 DATA 50,34,20,8,20,98,98,0,50,34,34,20
290 DATA 8,24,24,0,62,6,4,8,16,96,126,0
300 DATA 251,114,0,0,0,0,0,0,255,111,7,7
310 DATA 3,3,1,0,255,255,246,224,224,192,128,0
320 DATA 255,255,118,100,0,0,0,0,0,0,0,0
330 DATA 0,1,211,247,0,0,0,0,0,0,0,0
340 DATA 24,16,16,16,0,48,48,0,102,102,102,0
350 DATA 0,0,0,0,128,128,192,192,224,226,246,255
360 DATA 1,1,3,3,7,135,223,255,255,255,255,255
370 DATA 255,255,255,255,192,224,224,192,128,192,128,128
380 DATA 3,1,3,7,3,1,3,3,128,64,32,16
390 DATA 8,4,2,1,1,2,4,8,16,32,64,128
400 DATA 24,36,66,153,153,66,36,24,24,60,126,121
410 DATA 231,126,60,24,0,0,0,0,0,16,56,126
420 DATA 126,124,28,8,0,0,0,0,0,128,192,224
430 DATA 240,192,128,0,0,1,1,3,15,7,3,0
```

```
440 DATA 62,50,38,42,50,98,126,0,12,24,40,8
450 DATA 8,24,24,0,62,50,2,62,32,96,126,0
460 DATA 62,50,2,14,2,98,126,0,50,34,34,62
470 DATA 2,6,6,0,62,48,32,62,2,98,126,0
480 DATA 62,50,32,62,34,98,126,0,63,2,2,7
490 DATA 2,6,6,0,62,50,34,62,34,98,126,0
500 DATA 62,50,34,62,2,98,126,0,0,0,0,0
510 DATA 0,0,0,0,1,1,1,1,1,1,1,1
520 DATA 255,0,0,0,0,0,0,0,0,128,128,128
530 DATA 128,128,128,128,0,0,0,0,0,0,0,255
540 DATA 62,50,2,30,16,0,48,48,166,224,232,224
550 DATA 20,240,3,134,224,96,169,0,133,224,165,225
560 DATA 201,0,240,3,76,48,50,169,60,141,81,49
570 DATA 141,86,49,169,126,141,82,49,141,85,49,169
580 DATA 231,141,83,49,141,84,49,169,1,133,225,96
590 DATA 169,36,141,81,49,141,86,49,169,66,141,82
600 DATA 49,141,85,49,169,153,141,83,49,141,84,49
610 DATA 169,0,133,225,96,0,166,226,232,224,250,240
620 DATA 3,134,226,96,169,0,133,226,165,227,201,0
630 DATA 240,3,76,120,50,169,0,162,16,157,63,49
640 DATA 202,208,250,169,1,133,227,96,169,1,141,72
650 DATA 49,141,71,49,169,2,141,73,49,141,70,49
660 DATA 169,4,141,74,49,141,69,49,169,8,141,75
670 DATA 49,141,68,49,169,16,141,67,49,141,76,49
680 DATA 169,32,141,66,49,141,77,49,169,64,141,78
690 DATA 49,141,65,49,169,128,141,64,49,141,79,49
700 DATA 169,0,133,227,96,0,0,0,173,0,220,133
710 DATA 250,41,1,201,0,240,3,76,216,50,174,1
720 DATA 208,202,142,1,208,76,234,50,165,250,41,2
730 DATA 201,0,240,3,76,234,50,174,1,208,232,142
740 DATA 1,208,165,250,41,4,201,0,240,3,76,32
750 DATA 51,174,0,208,202,224,0,240,6,142,0,208
760 DATA 76,32,51,173,16,208,41,1,201,1,240,6
770 DATA 142,0,208,76,32,51,173,16,208,73,1,141
```

```

780 DATA 15,208,169,255,141, 0,208,234,165,250, 41, 8
790 DATA 201, 0,240, 3, 96,234,234,174, 0,208,232,224
800 DATA 255,240, 4,142, 0,208, 96,173, 16,208, 41, 0
810 DATA 201, 0,240, 6,169, 0,141, 0,208, 96,173, 16
820 DATA 208, 9, 1,141, 16,208,169, 0,141, 0,208, 96
830 DATA 166,228,232,224, 6,240, 3,134,228, 96,169, 0
840 DATA 133,228,173, 0,220,133,250, 41, 4,201, 0,240
850 DATA 3, 76,134, 51,174,248, 7,202,224,152,144, 6
860 DATA 142,248, 7, 96, 0, 0,169,152,141,248, 7, 96
870 DATA 0, 0,165,250, 41, 8,201, 0,240, 3, 96, 0
880 DATA 0,174,248, 7,232,224,154,176, 6,142,248, 7
890 DATA 96, 0, 0,169,154,141,248, 7, 96,173, 0,220
900 DATA 41, 16,201, 0,240, 1, 96,173, 16,208, 41, 1
910 DATA 201, 0,240, 20,169, 3,141, 16,208,173, 0,208
920 DATA 141, 2,208,173, 1,208,141, 3,208, 76,221, 51
930 DATA 169, 0,141, 16,208,173, 0,208,141, 2,208,173
940 DATA 1,208,141, 3,208,173,248, 7,201,152,240, 3
950 DATA 76,236, 51,169, 1,133,230, 96,201,153,240, 3
960 DATA 76,248, 51,169, 2,133,230, 96,201,154,240, 3
970 DATA 76, 4, 52,169, 3,133,230, 96,169, 0,133,230
980 DATA 169, 0,141, 2,208,141, 3,208, 96, 0,165,230
990 DATA 201, 0,240, 3, 76, 28, 52, 96,201, 2,240, 3
1000 DATA 76, 58, 52,174, 3,208,232,142, 3,208,224,250
1010 DATA 240, 1, 96,169, 0,133,230,141, 2,208,141, 3
1020 DATA 208, 96,201, 1,240, 3, 76,114, 52,174, 2,208
1030 DATA 202,142, 2,208,224, 0,240, 1, 96,173, 16,208
1040 DATA 41, 2,201, 2,240, 3, 76,103, 52,169,255,141
1050 DATA 2,208,173, 16,208, 73, 2,141, 16,208, 96,169
1060 DATA 0,133,230,141, 2,208,141, 3,208, 96,201, 3
1070 DATA 240, 3, 76, 0, 52,174, 2,208,232,142, 2,208
1080 DATA 224,255,240, 1, 96,173, 16,208, 41, 2,201, 2
1090 DATA 240, 3, 76,156, 52,169, 0,133,230,141, 2,208
1100 DATA 141, 3,208, 96,173, 16,208, 9, 2,141, 16,208
1110 DATA 169, 0,141, 2,208, 96, 32, 0, 50, 32, 82, 50
1120 DATA 32,192, 50, 32, 84, 51, 32,165, 51, 32, 18, 52
1130 DATA 32, 18, 52, 32,208, 52, 32,230, 52, 32, 5, 53
1140 DATA 32, 32, 54, 76, 80, 54,234,234,166,239,232,224

```

```

1150 DATA 100,240, 3,134,239, 96,174, 1,208,232,142, 1
1160 DATA 208,169, 0,133,239, 96,174, 2,208,232,232,138
1170 DATA 74, 74, 74,170,202,202,134,252,173, 3,208, 74
1180 DATA 74, 74,170,202,202,202,202,134,253, 96, 0
1190 DATA 0, 24,169, 0,133,250,169, 4,133,251,165,250
1200 DATA 101,252,133,250,162, 0, 24,165,253,101,250,133
1210 DATA 250,176, 8,232,224, 40,208,242, 76, 48, 53,230
1220 DATA 251, 76, 31, 53, 0, 0, 0, 0,160, 0,177,250
1230 DATA 201, 42,240, 3, 76, 72, 53,169, 32,145,250,169
1240 DATA 0,141, 2,208, 76,220, 53, 0,201, 40,240, 10
1250 DATA 76, 79, 53,201, 41,240, 3, 76, 87, 53, 96,201
1260 DATA 32,240, 3, 76, 95, 53, 96,169, 0,141, 2,208
1270 DATA 141, 3,208, 96, 0, 8,255,255, 0, 0,255,255
1280 DATA 28, 84, 79, 77, 79, 82, 82, 79, 87, 17, 17,157
1290 DATA 157,157,157,157,157, 83, 79, 70, 84, 17, 17,157
1300 DATA 157,157,157,157,157, 80, 82, 69, 83, 69, 78, 84
1310 DATA 83, 17, 17, 17,157,157,157,157,157,157, 31, 67
1320 DATA 65, 86, 69, 17, 17,157,157,157, 79, 70, 17, 17
1330 DATA 157,157,157,157, 68, 69, 65, 84, 72, 32, 33, 17
1340 DATA 17, 17,157,157,157,157,157,157,157,158, 64, 32
1350 DATA 49, 57, 56, 52,160, 8, 24, 32,240,255,162, 0
1360 DATA 189,112, 53, 32,210,255,232,224, 88,208,245, 96
1370 DATA 141, 3,208,166,249,232,134,249,224, 97,240, 3
1380 DATA 96,234,234,169, 2,141, 32,208,169, 32,162, 24
1390 DATA 157,248, 6,157, 32, 7,157, 72, 7,162, 25,157
1400 DATA 248, 6,157, 32, 7,157, 72, 7,160,255,162, 0
1410 DATA 142, 32,208,232,208,250,136,208,245,169, 0,141
1420 DATA 32,208, 96, 0, 0,191,255, 0,173, 0,208,201
1430 DATA 220,176, 1, 96,173, 1,208,201,192,176, 1, 96
1440 DATA 169, 6,141, 32,208,160,255,162, 0,142, 33,208
1450 DATA 202,208,250,136,208,245,169, 0,141, 33,208,169
1460 DATA 1,133, 2, 96,234,164,234,234,162, 10,160, 0
1470 DATA 136,208,253,202,208,248,165, 2,201, 1,240, 3
1480 DATA 76,100, 54, 96,169, 0,141, 31,208,173, 31,208
1490 DATA 41, 1,201, 1,240, 3, 76,170, 52,169, 2,133
1500 DATA 2, 96
1510 IF<> 161245 THEN PRINT"FEHLER IN DATAS !!"
```

SPEICHER-PROGRAMM

```

10 AA=12280
12 EA=13950
14 OPEN1,8,1,"CAVE MC CODE"
16 HB=INT(AA/256):LB=AA-HB*256
18 PRINT#1,CHR$(LB);CHR$(HB);
20 FORI=ATOEA
22 PRINT#1,CHR$(PEEK(I));
24 NEXTI
26 CLOSE1

```

READY.

[illegible]


```

28 PRINT"&*****%'%%J)(<>'&"
29 PRINT"&*****'↑↑↑( )<>'&"
30 PRINT"&***** ** ( )<>'&"
31 PRINT"&***** ** ( )<>'&"
32 PRINT"&*****+ + + + +( $ % # + + + + $"
33 PRINT"# + + + $ / / / / / # + + + + + $ / / / / / "
35 PRINT"$ SPC(255):PRINTSPC(96):SYS13774
40 FORX=9728TOX+254:READQ:POKEV,Q:NEXT
41 DATA 255,254,0,1,0,0,3,128,0
42 DATA 7,192,0,12,224,1,24,240,1
43 DATA 48,248,1,96,255,255,64,255,253
44 DATA 127,255,225,63,255,1,31,240,0
45 DATA 15,224,0,7,192,0,11,160,0
46 DATA 144,16,0,127,255,0,0,0,0
47 DATA 0,0,0,0,0,0,0,0,0,0
48 DATA 15,255,224,0,16,0,0,56,0
49 DATA 0,124,0,0,198,0,1,131,0
50 DATA 3,1,128,6,0,192,4,0,64
51 DATA 7,255,192,3,255,128,1,255,0
52 DATA 0,254,0,0,124,0,0,186,0
53 DATA 1,1,0,1,1,0,0,0,0
54 DATA 0,0,0,0,0,0,0,0,0,0
55 DATA 0,127,255,0,0,128,0,1,192
56 DATA 0,3,224,128,7,48,128,15,24
57 DATA 128,31,12,255,255,6,191,255,2
58 DATA 135,255,254,128,255,252,0,15,248
59 DATA 0,7,240,0,3,224,0,5,208
60 DATA 0,8,9,0,255,254,0,0,0
61 DATA 0,0,0,0,0,0,0,0,0,0
62 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,44,0,0,22,0,0,89,0,0,62
63 DATA0,0,91,0,0,44,0,0,82,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
64 DATA0,0,0,0
65 POKE224,0:POKE225,0:POKE226,0:POKE227,0:POKE228,0:POKE230,0:POKE229,0
66 POKE249,0:POKE2,0:V=53248:POKE2040,153:POKEV+39,5
67 POKEV,238:POKEV+1,72
68 POKE2041,155:POKEV+2,0:POKEV+3,0:POKEV+40,1:POKEV+21,3
82 TI$="000000":SC=0
1000 SYS13482
1001 POKE53281,2:POKE53280,6:POKEV+21,0
1002 IFPEEK(2)=1THENGOTO1050
1005 PRINT"SORRY YOU ARE DEAD !!"
1010 PRINT"TOMORROW SOFT"
1012 PRINT"HOPE YOU"
1014 PRINT"TO MEET AGAIN !"
1043 POKEV+31,0:END
1050 FRINT"MISSION COMPLETED !!!"
1060 PRINT"TIME \"TI$"
1080 PRINT"PLEASE TRY\""
1090 PRINT"A HIGHTER LEVEL !"
1100 POKEV+31,0:END
10000 REM * LEVEL AENDERN *
10001 REM *
10002 REM * POKE 13905,(18)
10003 REM * PAUSEDAUER
10004 REM *
10005 REM * POKE 12836,(250)
10006 REM * LASER TAKT
10007 REM * POKE 13524,(34)
10008 REM * SINKEN DES
10009 REM * HELICOPTERS
10010 REM *
10011 REM * JE KLEINER DIE WERTE
10012 REM * DESTO GROESSER DIE
10013 REM * GESCHWINDIGKEIT
10014 REM *
10015 REM * 1984 TOMORROW SOFT

```

INTERPOD IM TEST

Ein Interface von OCS als Schlüssel zu vielen neuen Möglichkeiten

Eines derjenigen Probleme, die leidenschaftlichen Commodore-Usern am stärksten zu schaffen machen, ist der Umstand, daß weder VC-20 noch C-64 einen IEEE-Bus besitzen. Gerade angesichts der Leistungsfähigkeit des 64ers ist es schade, daß man immer wieder nur auf die vom Hersteller für die Heimcomputer-Modelle angebotenen Peripherien verwiesen wird, mag es sich um eine Floppy oder für jede ernsthafte Arbeit nötigen Drucker handeln.

Die Rettung scheint für Insider mit dem seit geraumer Zeit in England erhältlichen und ziemlich erfolgreichen INTERPOD-Interface zu nahen, das von Oxford-Computer-Systems angeboten wird. Vielseitig, wie Interpod zu sein scheint, war es natürlich auch für Ihre "Compute mit" einen Test wert...

INTERPOD - diese acht Buchstaben stehen für ein kleines weißes Kästchen, das trotz der geringen Maße (148 x 96 x 24mm) über ein beachtliches Innenleben verfügt.

Man steckt es ganz einfach an die serielle Schnittstelle des jeweiligen Rechners, woran normalerweise Diskettenlaufwerk oder Drucker angeschlossen werden.

INTERPOD selbst besitzt zunächst einmal zwei für IN- und OUTPUT getrennte Schnittstellen, eine dritte, ähnlich konzipierte, dient der Stromversorgung.

Diese niedliche Sammlung von Anschlüssen wird durch einen IEEE-Bus (Commodore-Version/ nicht zu verwechseln mit dem bekannten 488er) und eine RS-232-C komplettiert.

Letztere Schnittstelle ist besonders leicht zu bedienen, zumal sie, wie wir feststellen, den über sie angesprochenen Peripherien die Adresse 4 zuweist. Dadurch wird die Bedienung des jeweiligen Gerätes genauso unkompliziert wie die Ansteuerung des

Druckers unter normalen Voraussetzungen (d.h. bei Direktanschluß). Was uns darüber hinaus sehr gut gefiel, ist der Umstand, daß man die Baudrate den persönlichen Gegebenheiten anpassen kann: von kläglichen 50 Baud (im Verlag haben wir einige Leute, die sogar schneller sprechen) bis hin zur schon respektablen Rate von 7200 Baud.

Doch dies ist nicht das einzige, was sich individuell anpassen läßt. Nahezu alles, einschließlich Parität, Stopbits und Länge des Datensatzes ist durchaus bereit, sich eine persönliche Note verpassen zu lassen. Selbstverständlich dürfen Sie aber auch die Standardraten wählen und mit 1200 Baud, 8Bit Wortlänge und einem Stopbit arbeiten (paritätslos).

Was den vorgenannten Commodore-IEEE-Bus betrifft, so dürfte allerdings auch er nicht uninteressant sein: denken Sie nur an die vielen schönen Commodore-Floppies, die nur darauf warten, über IEEE angesprochen zu werden. Selbst ein Full-Professionel-Laufwerk wie die CBM 8050 wäre als Partnerin des VC-20 oder C-64 denkbar (wenngleich schon ein wenig Snobismus dazu gehört, wenn ich an die Preisrelation denke).

INTERPOD, das uns ausnehmend gut gefiel (trotz des nicht unerheblichen Inve-

stitionsvolumens von immerhin knapp 600 Mark), leistet aber noch mehr: wenn Sie eine Peripherie ansteuern, die nicht verfügbar ist, so heißt es nicht kurz und bündig "Device not present" - das schlaue Interface sucht vielmehr erst einmal nach geeignetem Ersatz. Erst nach erfolgloser Suche (wenn z.B. nicht nur die eigentlich gewünschte Floppy, sondern auch das als Ersatzadresse denkbare Kassettengerät fehlt) gibt's einen freundlichen Hinweis, der Sie ersucht, doch wenigstens ein einziges Gerät bereit zu stellen.

FAZIT:

Das recht teure, dafür aber ganz schön pfiffige Gerät bedeutet eine mehr als brauchbare Erweiterung Ihrer Commodore-Ausrüstung. Anwendungsprobleme gibt es kaum,

da sich die eingebaute CPU (INTERPOD verfügt über eigenen 6502-Prozessor) um praktisch alles kümmert. Dadurch ist es auch nicht tragisch, daß die beigegepackte Information ausgesprochen wortkarg ist.

Gerade für Leute, die ihren Commodore-Rechner nicht nur privat, sondern auch als Personal Computer im Büro einsetzen wollen, bringt das gute Stück ein hohes Maß an Profi-Kompatibilität.

Unseres Wissens wird es bislang nicht in Deutschland vertrieben. Als Kontaktadresse muß also die "Heimatzeitschrift" des Herstellers genügen:

Oxford-Computer-Systems
Hensington Road
Woodstock, Oxford, 'X7 1JR
England

(sk)

★ ★ ★

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

- eingebautes Interface für seriellen Port
- Adresse 4 oder 5 einstellbar
- Schreibmaschine mit Korrekturspeicher
- ON-OFF-Line Taste trennt Betriebsarten
- Zeichenabstand (10, 12 und 15 Z/Zeile) 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



Tips & Tricks

für den VC-20

Bei veröffentlichten Listings finden Sie des öfteren die folgende Lade-Anweisung: POKE 44,28: POKE 7168,0: NEW.

Meist hat man damit ja auch Erfolg – in Einzelfällen allerdings kann es zu Schwierigkeiten führen, da geringfügige Bestückungsunterschiede innerhalb der verschiedenen Produktionsserien (ein und desselben Modells) schon zu Ladefehlern führen können. In einem solchen Fall ist es ratsam, das jeweilige ProgrZmm im Speicher zu verlagern. Sie laden und starten zunächst wie gewöhnlich, testen sorgfältig alle Data-Zeilen und löschen die damit zusammenhängenden GOSUB's. Dann fragen Sie mit PEEK (45) und PEEK (46) das Basic-Ende ab (Werte sollten unbedingt notiert werden) und poken wie folgt: POKE 43,1:POKE 44,18:NEW – kein sichtbares Programm befindet sich mehr im Speicher.

Jetzt geben Sie ein POKE 43,1:POKE 44,28:POKE 7168,0:RUN. Und nun fehlen lediglich noch die Werte von vorhin (wir geben sie, da sie von Rechner zu Rechner unterschiedlich sein können, hier mit X an): POKE 45,X:POKE 46,X:SAVE "PRG-NAME". Bitte beachten: statt "PRG-NAME" muß natürlich die Bezeichnung Ihres jeweiligen Programmes stehen!

Danach können die Programme normal geLOAded und geSAVED werden.

Nachfolgend haben wir für Sie als engagierten VC-20 User eine – wie wir meinen – nützliche Tabelle der wichtigsten PEEK's und POKE's zusammengestellt. Da finden Sie so ziemlich alles, was für die Programmierarbeit von Nutzen ist...

POKE UND PEEK für VC-20

- 10:** Kassettenoperationen (0=Load/1=Verify)
- 43,44:** Anfangsadresse für Basicprogramm
- 45,46:** Beginn der Variablen
- 47,48:** Beginn der Felder
- 49,50:** Ende der Felder
- 55,56:** Endadresse des freien Speichers

- 57,58:** Aktuelle Basic-Zeilenummer
- 59,60:** Zeilenummer, in der das Programm gestoppt wurde
- 69,70:** Letzter benutzter Variablenname
- 144:** Nach fehlerfreiem Ladevorgang steht 0
- 152:** Zahl der offenen Files
- 153:** Eingabegerät (normal 0)
- 154:** Ausgabegerät (normal 4)
- 157:** 0=Prg. Modus/128=Direktmodus
- 160,162,162:** Entspricht TI:65536
PEEK(160)+256PEEK(161)+PEEK(162)
- 178,179:** Beginn des Kassettenbuffers
- 183:** Nr. des Zeichens im Filename
- 184:** Momentanes logisches File
- 186:** Momentanes Gerät (Peripherie)
- 197:** Mol. entan gedrückte Taste
- 198:** Anzahl der Zeichen im Tastaturpuffer
- 199:** 0=RVS on/18=RVS off
- 201,202:** Eingabecursor (Reihe,Spalte)
- 203:** Momentan gedrückte Taste
- 204:** Cursor einschalten (0=Blinkcursor)
- 206:** Zeichen unter dem Cursor
- 211:** Spaltenposition des Cursors
- 214:** Zeilenposition des Cursors
- 631-640:** Tastaturpuffer: POKE631,19: forl=1 to Anzahl der Prints +1 (Max 10):POKE631+I,13:NEXT:POKE198, Anzahl der Prints+1:POKEd Eingaben in Data Zeilen
- 641,642:** Beginn Speicher Betriebssystem
- 643,644:** Ende Speicher Betriebssystem
- 646:** Code für momentane Farbe

- 647:** Code für Farbe unter dem Cursor
- 649:** Größe des Tastaturpuffers (Max.10)
- 650:** Repeat Funktion (0=normal/64=keine/128=keine Tasten)
- 651:** Repeat Zähler
- 657:** Shift(0=frei/128=verriegelt)
- 775:** List Schutz:POKE775,1
- 788:** RUN/STOP-TASTE (191 ein/ 194 aus)
- 801,802,818:** POKE 801,0:POKE 802,0: POKE 818,165:SAVEschutz
- 828-1019:** Kassettenpuffer
- 36864:** Horizontale Zentrierung des Bildausschnittes
- 36865:** Vertikale Zentrierung des Bildausschnittes
- 36866:** Spaltenzahl
- 36867:** Anzahl Zeilen am Bildschirm
- 36869:** Zeichensatz und andere Informationen
- 36870:** Horizontaler Wert des Lightpens
- 36871:** Vertikaler Wert des Lightpens
- 36872:** X-Wert Paddles
- 36873:** Y-Wert Paddles
- 237148:** Kassettenmotor (251=aus/253=ein)
- 37150:** POKE37150,2: Setzt RESTORE-Taste außer Betrieb
- 37151:** Wait 37151,64,64: Wartet bis Taste am Kassettenrecorder gedrückt wurde: WAIT 37151,64: Wartet bis Taste am Tape ausgedrückt wurde
- 37879:** Geschwindigkeit des Cursors (normal 72)
- SYS 64818 oder SYS 64821:** Rechner im Einschaltzustand. ACHTUNG: RAM wird gelöscht
- CHR\$(8):** Schaltet Zeichensatzumschaltung über Tastatur aus
- CHR\$(9):** Schaltet Zeichensatzumschaltung über Tastatur wieder ein.

Bildverschiebung

Ein Hilfsprogramm, das den Text nach unten wegrollen läßt, den Bildschirm löscht und nach oben kommt.

```
10 FOR X = 39 TO 150: POKE 36881,X:NEXT: PRINT
"Clear-Taste"
20 REM NEUER TEXT
30 FOR X = 150 TO 39 STEP-1:POKE 36881,X:NEXT
X
```

Bei Anleitungen kann diese Routine sehr nützlich sein, denn die 1. Seite verschwindet automatisch und es folgt der nächste Text ohne Unterbrechung.

Tips & Tricks

für den Commodore 64

Maschinenunterstütztes Scroll in alle Richtungen

Das nachfolgende Hilfsprogramm ist eigentlich schon eine ganze Programmsammlung.

In jeweils 2 Zeilen befinden sich kurze, in Basic ansprechbare Schleifen, die es ermöglichen, den Bildschirm mit hoher Geschwindigkeit in jede Richtung zu verschieben.

Die Basicadressen dafür sind:

53265 (Y Richtung)

53270 (X Richtung)

Die Geschwindigkeit, in der der Scroll-

vorgang ablaufen soll, wird bestimmt durch die Step-Anweisung in der Bewegungsschleife!

Bedenken Sie, daß bei Anwendung der Routine in Programmen die Geschwindigkeit durch eingefügte Anweisungen und Abfragen verlangsamt wird.

Richten Sie sich daher beim Setzen der Verzögerungsschleife nicht nach geschätzten Werten, sondern nach dem Ausprobieren bei Testläufen.

```
*****
*          TASTATUR-PIPPSER          *
*          EIN PROGRAMM VON:         *
*          HORST HENKLER             *
*          *****                   *
```

```
100 DATA 160,001,162,140,140,000,212,140
110 DATA 001,212,169,000,141,002,212,142
120 DATA 003,212,160,255,142,005,212,140
130 DATA 006,212,162,015,160,033,140,004
140 DATA 212,142,024,212,096,032,135,234
150 DATA 192,000,240,015,162,040,142,001
160 DATA 212,032,135,234,192,000,240,003
170 DATA 076,049,234,162,000,142,001,212
180 DATA 076,049,234,078,068,000,000,000
190 FOR I=0 TO 71: READ A:S=S+A:POKE16384+I,A
200 NEXT
210 IF SC<>7963 THEN PRINT "FFHIER IN DEN DATA ZEILEN":END
220 POKE788,37:POKE789,64:SYS16384
230 PRINT "O . K . ■"
```

```
10 .O $4000
20 LDY #1
30 LDX #100
40 STY 54272
50 STY 54273
60 LDA #0
70 STA 54274
80 STX 54275
90 LDY #255
100 STX 54277
110 STY 54278
120 LDX #15
130 LDY #33
140 STY 54276
150 STX 54296
160 RTS
170 START JSR 60039
180 CPY #0
190 BEQ END
200 LDX #40
210 STX 54273
220 JSR 60039
230 CPY #0
240 BEQ END
250 JMP 59953
260 END LDX #0
270 STX 54273
280 JMP 59953
```

Unsichtbare Zeilen

Wenn Sie nach ganz einfachen Mitteln suchen, bei Ihrem Commodore 64 und VC-20 einen Programmschutz einzubauen, können wir Ihnen die Geistermethode, nämlich Unsichtbarkeit empfehlen.

In jeder Befehlszeile können, sofern es sich um weniger als 35 Zeichen handelt, diese bei einem späteren Listen zum Verschwinden gebracht werden.

Dies wird erreicht, in dem man

:REM""

am Ende der Zeile eintippt, die man schützen möchte. Dann löscht man die

zweiten Anführungsstriche und rückt vierzigmal die Tasten SHIFT- und INST/DEL. Danach drückt man vierzigmal die DEL Taste.

Dies bewirkt, daß das reverse T vierzigmal auf den Bildschirm gebracht wird.

Danach wird die RETURN Taste gedrückt, um die Zeile festzuschreiben.

Gibt man nun den LIST-Befehl ein, wird diese so geschützte Zeile zwar vom Computer erkannt und angenommen, auf dem Bildschirm jedoch nicht sichtbar sein. Mit Hilfe dieser Funktion kann man sich mit etwas Geschick auf einen eigenen Listenschutz aufbauen, der von anderen nur schwer geknackt werden kann.

TANKBATTLE



Vergessen Sie alles, was Sie über Video-Panzerschlachten bisher gehört haben.
Jetzt kommt Tankbattle!

Tankbattle besteht aus vier Teilprogrammen, die nacheinander einzutippen und abzuspeichern sind:

- 1) Tankbattle Relay (Definition der Sonderzeichen und Spielanleitung)
- 2) Tankbattle (Panzerschlacht)
- 3) Nightbattle (Panzerschlacht bei Nacht)
- 4) Robotbattle (Panzerschlacht gegen den Computer)

Zu 1.:

Tankbattle Relay beginnt, sobald man es gestartet hat, mit einem Einleitungseffekt. Danach werden die Sonderzeichen eingegeben.

Jetzt erscheint auf dem Bildschirm die Spielanleitung.

In ihr wird der Spielablauf sowie die Sonderzeichenklärung dem Benutzer klargemacht. Ebenfalls kann man in der Anleitung zwischen drei verschiedenen Panzerschlachten wählen. Danach wird die gewünschte Panzerschlacht geladen.

Zu 2.:

Tankbattle ist ein Spiel für 2 Personen. Der grüne Panzer wird mit der Tastatur

(Tasten: A,W,D,X) gesteuert, der rote mit dem Joystick. Nachdem man sich geeinigt hat, wer welchen Panzer fährt, muß man den Schwierigkeitsgrad (Level 1-12) eingeben.

Je höher der Schwierigkeitsgrad ist, desto weniger werden die Hindernisse und desto mehr werden die Minen.

Nun muß jeder Spieler versuchen, den gegnerischen Panzer zu zerstören, ohne dabei auf die gelben Minen zu steuern. Die anderen Hindernisse machen dem Panzer nichts aus, er kommt allerdings da nicht weiter.

Wer als erster 10 Punkte gesammelt hat, hat gewonnen.

Am oberen Bildschirmrand werden die jeweiligen Punkte eingeblendet, dazwischen steht "HI:". Das bedeutet "HIGH", und zeigt den Spielern an, welcher Panzer in Punkten führt.

Wenn es unentschieden steht, wird dort ein Strich hingedruckt.

Bei einem Zusammenstoß der Panzer bekommt keiner der Spieler einen Punkt. Nachdem es einem Spieler gelungen ist, den gegnerischen Panzer zu zerstören, oder nach einem Zusammenstoß der beiden Panzer, oder nach einem Auffahren auf eine Mine, werden die Panzer in ihre Ausgangsstellung gebracht.

Wenn ein Spieler 10 Punkte erreicht hat, ist die Schlacht beendet und der Sieger wird in den Bildschirm eingeblendet. Aber es muß ja nicht bei einer Panzerschlacht bleiben.

Übrigens: Die Schüsse der Panzer haben eine begrenzte Schußweite.

Zu 3.:

Diese Panzerschlacht funktioniert genau wie die erste, nur daß hier die Panzer nur sichtbar sind, wenn man den Feuerknopf betätigt.

Hier weiß man also nie ganz genau, wo sich der eigene Panzer zur Zeit befindet.

Zu 4.:

Robotbattle ist eine Panzerschlacht für eine Person. Hier muß man also gegen den Computer spielen, aber aufgepaßt: Der Computer ist so programmiert, daß er schnell reagieren und vor allen Dingen schnell schießen kann.

Auch bei diesem Spiel kann man den Schwierigkeitsgrad vorher bestimmen.

Bei allen Spielen kann man, falls ein Spieler von Hindernissen oder Minen umzingelt ist, die "F1"-Taste drücken. Danach wird ein neuer Bildschirmaufbau geliefert.

Wie schon gesagt, ist es am besten, wenn die ganzen Spiele hintereinander abgespeichert werden, dann gibt es danach keine Probleme mehr beim Laden. Bei dem Programm Tankbattle muß man das Listing nur um einige Zeilen verändern, um auf das Spiel Nightbattle zu kommen.

programme

Die zu ändernden Zeilen sind:

```

5 PRINT "X":F=30720:POKES-9,255
14 PRINTA$:"X":FORT=0TOR:Z=INT(RND(1)*417+7746):POKEZ+F,6:POKEZ,1:NEXT
58 IFD=41THEN600
60 IFD2=0THEN610
70 GOTO40
98 POKER1,32:GOTO60
99 POKER2,32:GOTO40
600 POKER1,G1:IFPEEK(R1+T1)<2THEN98
601 FORS3=R1+T1TOR1+T1*4STEPT1:IFS3=R2THEN1000
602 IFPEEK(S3)>1THENPOKES3+F,5:POKES3,S1:POKES3-T1,32:GOSUB801:NEXT
603 POKES3-T1,32:POKER1,32:GOTO60
610 POKER2,G2:IFPEEK(R2+T2)<2THEN99
611 FORS4=R2+T2TOR2+T2*4STEPT2:IFS4=R1THEN1005
612 IFPEEK(S4)>1THENPOKES4+F,2:POKES4,S2:POKES4-T2,32:GOSUB800:NEXT
613 POKES4-T2,32:POKER2,32:GOTO40
702 IFPEEK(R1+T1)>1THENR1=R1+T1:POKER1+F,5:POKER1-T1,32:GOSUB801
712 IFPEEK(R2+T2)>1THENR2=R2+T2:POKER2+F,2:POKER2-T2,32:GOSUB800
1602 PRINT "||||| NIGHTBATTLE |||||"
1604 PRINT "||||| PANZERSCHLACHT B. N. |||||"
1605 PRINT "|||||"
2000 A1=195:D1=.3:GOSUB2010
2002 A1=175:D1=.4:GOSUB2010
2003 A1=175:D1=.4:GOSUB2010:A1=183:D1=.2:GOSUB2010
2004 A1=175:D1=.4:GOSUB2010
2005 A1=0:D1=.2:GOSUB2010
2006 A1=175:D1=.2:GOSUB2010
2007 A1=195:D1=.4:GOSUB2010

```

TANKBATTLE

```

0 S=36878:POKES+1,8:INPUT "LEVEL (1-12)":R:A=60-(R*5):A3=R*5
2 IFR<10RR>12THENRUN
3 P1=0:P2=0:PRINT "SPC(227)" "LEVEL" R
4 FORT=15TORSTEP-1:FORG=254TOR235STEP-1:POKES,T:POKES-4,G:NEXT:NEXT:POKES-4,0
5 PRINT "X":F=30720:POKES-9,255
10 A$="AAAAAAAAAAAAAAAAAAAAA":PRINT "M" A$;:FORT=0TOR17:PRINT "A" SPC(20) "A":NEXT
14 PRINTA$:"M":FORT=0TOR:Z=INT(RND(1)*417)+7746:POKEZ+F,3:POKEZ,1:NEXT
15 FORT=1TOR3:K=INT(RND(1)*418)+7746:IFPEEK(K)>1THENPOKEK,0:POKEK+F,7:NEXT:GOTO1
7
16 NEXT
17 R1=7923:R2=7942:POKER1+F,5:POKER1,7:POKER2+F,2:POKER2,6:T1=1:T2=-1:POKES,15
18 IFP1>90RPP2>9THEN1600
19 S1=3:S2=3:G1=7:G2=6:GOSUB3000
20 IFP1=P2THEN40
21 POKE7714,8:IFP1>P2THENPOKE38434,5:GOTO40
22 IFP2>P1THENPOKE38434,2
40 D=PEEK(197):SYS673:D1=PEEK(255):D2=PEEK(254)
41 IFD=39THEN5
42 IFD=9THEN1=-22:G1=8:S1=2:GOTO700
44 IFD1=152THEN1=-22:G2=8:S2=2:GOTO710
46 IFD=26THEN1=22:G1=9:S1=2:GOTO700
48 IFD1=143THEN1=22:G2=9:S2=2:GOTO710
50 IFD=18THEN1=1:G1=7:S1=3:GOTO700
52 IFD1=28THEN1=1:G2=7:S2=3:GOTO710
54 IFD=17THEN1=-1:G1=6:S1=3:GOTO700
56 IFD1=140THEN1=-1:G2=6:S2=3:GOTO710
58 IFD=41ANDPEEK(R1+T1)>1THEN600
60 IFD2=0ANDPEEK(R2+T2)>1THEN610
99 GOTO40
600 FORS3=R1+T1TOR1+T1*4STEPT1:POKER1,G1:IFS3=R2THEN1000
601 IFPEEK(S3)>1THENPOKES3+F,5:POKES3,S1:POKES3-T1,32:GOSUB801:NEXT
602 POKES3-T1,32:GOTO60
610 FORS4=R2+T2TOR2+T2*4STEPT2:POKER2,G2:IFS4=R1THEN1005
611 IFPEEK(S4)>1THENPOKES4+F,2:POKES4,S2:POKES4-T2,32:GOSUB801:NEXT
612 POKES4-T2,32:GOTO40
700 IFR1+T1=R2THEN1010
701 IFPEEK(R1+T1)=0THEN1015
702 IFPEEK(R1+T1)>1THENR1=R1+T1:POKER1+F,5:POKER1,G1:POKER1-T1,32:GOSUB800
704 IFT1=-22THEN44
705 IFT1=-1THEN56
706 IFT1=1THEN52
707 IFT1=22THEN48
710 IFR2+T2=R1THEN1010
711 IFPEEK(R2+T2)=0THEN1020
712 IFPEEK(R2+T2)>1THENR2=R2+T2:POKER2+F,2:POKER2,G2:POKER2-T2,32:GOSUB801
713 IFT2=-22THEN46
714 IFT2=-1THEN58
715 IFT2=1THEN54
717 IFT2=22THEN50
800 POKES-2,200:POKES-2,0:RETURN
801 POKES-2,205:POKES-2,0:RETURN
1000 POKES3-T1,32:POKER2,4:GOSUB1500
1001 POKER2,32:POKER1,32:P1=P1+1:GOTO17
1005 POKES4-T2,32:POKER1,4:GOSUB1500
1006 POKER1,32:POKER2,32:P2=P2+1:GOTO17
1010 POKER1,32:POKER2,32:GOSUB1500:GOTO17
1015 POKER1,32:POKER1+T1,4:GOSUB1500:POKER1+T1,32:POKER2,32:P2=P2+1:GOTO17
1020 POKER2,32:POKER2+T2,4:GOSUB1500:POKER2+T2,32:POKER1,32:P1=P1+1:GOTO17
1500 POKES-1,255:FORT=15TORSTEP-1:POKES,T:NEXT:POKES-1,0:RETURN
1600 PRINT "X" SPC(249) "GAME OVER" :GOSUB3000:GOSUB2000
1601 PRINT "||||| " :POKES-9,240
1602 PRINT "||||| TANKBATTLE |||||"
1604 PRINT "||||| PANZERSCHLACHT |||||"
1605 PRINT "|||||"
1610 PRINT "LEVEL :R:PRINT "GREEN TANK :P1:PRINT "RED TANK :P2

```

Die Programmbeschreibung und Variablenliste von Nightbattle sind genauso aufgebaut, wie das Programm Tankbattle.

Variablenliste Tankbattle Relay:

D = Grundwert für Tongeneratoren und Farbe
TY = Schrittvariable für Ton
A\$ = Zeichenketten für großen Panzer und Einleitungseffekt
G/T = Schleifenvariablen
X = Datalesevariable
D\$D1 = Abfragevariable
A\$(1)-A\$(3) = Art der Panzerschlacht, die geladen werden soll.

Variablenliste und Programmbeschreibung für Tankbattle und Nightbattle:

S = Grundwert für Tongeneratoren und Farbe
R = Eingabevariable (Level)
A = Anzahl der Hindernisse
A3 = Anzahl der Minen
P1 = Punkte des grünen Panzers
P2 = Punkte des roten Panzers
T/G = Schleifenvariablen
F = Wert, den man zu R1 oder R2 hinzu addieren muß, um die Farbramadresse zu erhalten
A\$ = Zeichenkette für Spielfeld, Abfragevariable für den Panzer, der gewonnen hat
Z = Zufallszahl für Hindernisse in den Bildschirm zu poken
K = Zufallszahl für Minen in den Bildschirm zu poken
R1 = Position grüner Panzer
R2 = Position roter Panzer
T1 = Richtung in die der grüne Panzer fährt
T2 = Richtung in die der rote Panzer fährt
S1 = Aussehen des Schusses grüner Panzer
S2 = Aussehen des Schusses roter Panzer
G1 = Aussehen des grünen Panzers
G2 = Aussehen des roten Panzers
D = Tastaturabfrage
SYS673/D1/D2 = Joystickabfrage
S3 = Schuß grüner Panzer
S4 = Schuß roter Panzer
A1 = Wert für Tongenerator
D1 = Ausklingzeit des Tones

0-3: Abfrage des Levels und setzen der Punkte

4-16: Bildschirmaufbau mit Minen und Hindernissen

17-19: Definition der Variablen und Abfrage ob Spiel zu Ende

20-22: Setzen des Panzers, der in Punkten führt


```

1614 IFP1>P2THEN A$="THE GREEN TANK" C1=C1+1 GOTO1617
1616 A$="THE RED TANK" C2=C2+1
1617 PRINT "THE RED TANK IS THE WINNER"
1620 PRINT "SPC(255)SPC(46)"PRESS [KEY]FOR [NEW] GAME
1621 PRINT "NEW LEVEL (J/N)?"POKE198,0
1622 GETA$:IFA$=""THEN1622
1623 IFA$="J"THENRUN
1624 GOTO3
1629 POKE198,0:WAIT198,1:P1=0:P2=0:A=A-5:A3=A3+4:IFAC1THEN A=0:A3=60
1630 GOTO1
2000 A1=225:D1=.7:GOSUB2010
2002 A1=221:D1=.6:GOSUB2010
2003 A1=225:D1=.07:GOSUB2010:A1=0:D1=.29:GOSUB2010
2004 A1=215:D1=.16:GOSUB2010
2005 A1=219:D1=.16:GOSUB2010
2006 A1=209:D1=.16:GOSUB2010
2007 A1=215:D1=.05:GOSUB2010
2008 POKES=4,0:POKES=3,0:POKES=2,0:RETURN
2010 POKES=4,A1:POKES=3,A1:POKES=2,A1:FORT=15TO0STEP-D1:POKES,T:NEXT:RETURN
3000 PRINT "TANK P1" HI:- "TANK P2: RETURN

```

ROBOTBATTLE

```

0 DIMR(11):S=36873:POKES+1,8:INPUT "LEVEL (1-12)";R:A=60-(R*5):A3=R*5
2 DEFFNG(X)=RND(1)*X:IFR(10RR)12THENRUN
3 P1=0:P2=0:PRINT "SPC(227)"LEVEL"R
4 FORT=15TO0STEP-1:FORG=254TO235STEP-1:POKES,T:POKES=4,G:NEXT:NEXT:POKES=4,0
5 PRINT "F=30720:POKES=9,255:R(0)=-22:R(1)=22
6 R(2)=-1:R(3)=1:R(4)=0:R(5)=9:R(6)=6:R(7)=7:R(8)=2:R(9)=2:R(10)=3:R(11)=3
10 A$="AAAAAAAAAAAAAAAAAAAAA":PRINT "A$";FORT=0TO17:PRINT "A"SPC(20)"A";NEXT
14 PRINTA$:FORT=0TOA:Z=INT(FNG(417))+7746:POKEZ+F,4:POKEZ,1:NEXT:A$=""
15 FORT=1TOA3:K=INT(FNG(418))+7746:IFPEEK(K)>1THENPOKEK,0:POKEK+F,7:NEXT:GOTO17
16 NEXT
17 R1=7923:R2=7942:POKER2+F,2:POKER2,6:T2=-1:POKES,15:IFP1>90RP2>9THEN1600
18 S2=3:G2=6:GOSUB3000:IFP1=P2THEN40
21 POKE7714,8:IFP1>P2THENPOKE38434,5:GOTO40
22 IFP2>P1THENPOKE38434,2
40 D=PEEK(197):SYS673:D1=PEEK(255):D2=PEEK(254):IFR1=R2THEN1010
41 IFD2=0ANDPEEK(R2+T2)>1THEN610
42 IFD=39THEN5
44 IFD1=152THENY=0:GOTO710
48 IFD1=143THENY=1:GOTO710
52 IFD1=23THENY=3:GOTO710
56 IFD1=140THENY=2:GOTO710
99 GOTO900
610 FORS4=R2+T2TOR2+T2*4STEPT2:IFS4=R1THEN1005
611 IFPEEK(S4)>1THENPOKES4+F,2:POKES4,S2:POKES4-T2,32:GOSUB800:NEXT
612 POKES4-T2,32:POKER2,G2:GOTO42
710 T2=R(Y):G2=R(Y+4):S2=R(Y+8):IFPEEK(R2+T2)=0THEN1020
712 IFPEEK(R2+T2)>1THENR2=R2+T2:POKER2+F,2:POKER2,G2:POKER2-T2,32:GOSUB800:IFR1=
R2THEN1010
720 GOTO900
800 POKES=2,210:POKES=2,0:RETURN
801 POKES=2,200:POKES=2,0:RETURN
900 IFV2>0THEN903
901 IFR1+T1=R2ORR1+T1*2=R2ORR1+T1*3=R2ORR1+T1*4=R2THEN950
902 IFV2=0THEN960
903 O=FNG(4)
904 T1=R(0):G1=R(0+4):S1=R(0+8):IFV2>0THENV2=V2-1
910 V=R1+T1:IFPEEK(V)<>32THENV2=2:GOTO40
911 IFV=R2+T2*2ORV=R2+T2*3ORV=R2+T2*4THEN40
920 R1=R1+T1:POKER1+F,5:POKER1,5:POKER1-T1,32:GOSUB801:GOTO40
950 V2=3:FOR S3=R1+T1TOR1+T1*4STEPT1:IFS3=R2THEN1000
952 IFPEEK(S3)>1THENPOKES3+F,5:POKES3,S1:POKES3-T1,32:GOSUB801:NEXT
953 POKES3-T1,32:POKER1,01:GOTO40
960 X=R2-R1:V1=3:IFABS(X)>21THENV1=1
970 O=INT(V1+SGN(X)/2):GOTO904
1000 POKES3-T1,32:POKER2,4:GOSUB1500
1001 POKER2,32:POKER1,32:P1=P1+1:GOTO17
1005 POKES4-T2,32:POKER1,4:GOSUB1500
1006 POKER1,32:POKER2,32:P2=P2+1:GOTO17
1010 POKER1,32:POKER2,32:GOSUB1500:GOTO17
1020 POKER2,32:POKER2+T2,4:GOSUB1500:POKER2+T2,32:POKER1,32:P1=P1+1:GOTO17
1500 POKES=1,255:FORT=15TO0STEP-1:POKES,T:NEXT:POKES=1,0:RETURN
1600 PRINT "SPC(249)"GAME OVER"GOSUB3000:GOSUB2000
1601 PRINT "
1602 PRINT "ROBOTBATTLE";
1604 PRINT "PANZERSCHLACHT G. C. I";
1605 PRINT "
1610 PRINT "LEVEL :R:PRINT "ROBOT TANK P1:PRINT "RED TANK P2
1614 IFP1>P2THEN A$="THE ROBOT TANK":C1=C1+1:GOTO1617
1616 A$="THE RED TANK":C2=C2+1
1617 PRINT "THE RED TANK IS THE WINNER"
1620 PRINT "SPC(255)SPC(46)"PRESS [KEY]FOR [NEW] GAME
1621 PRINT "NEW LEVEL (J/N)?"POKE198,0
1622 GETA$:IFA$=""THEN1622
1623 IFA$="J"THENRUN
1624 GOTO3
1629 POKE198,0:WAIT198,1:P1=0:P2=0:A=A-5:A3=A3+4:IFAC1THEN A=0:A3=60
1630 GOTO1
2000 A1=223:D1=.2:GOSUB2010
2002 A1=223:D1=.3:GOSUB2010
2003 A1=219:D1=.3:GOSUB2010:A1=215:D1=.2:GOSUB2010
2004 A1=209:D1=.2:GOSUB2010
2005 A1=207:D1=.2:GOSUB2010
2006 A1=201:D1=.2:GOSUB2010
2007 A1=195:D1=.2:GOSUB2010
2008 POKES=4,0:POKES=3,0:POKES=2,0:RETURN
2010 POKES=4,A1:POKES=3,A1:POKES=2,A1:FORT=15TO0STEP-D1:POKES,T:NEXT:RETURN
3000 PRINT "ROBOT P1" HI:- "TANK P2: RETURN

```

40-99: Hauptroutine: Tastatur- und Joystickabfrage zur Steuerung der Panzer

600-602: Unterroutine für den Schuß des grünen Panzers

610-612: Unterroutine für den Schuß des roten Panzers

700-707: Abfrage für grünen Panzer, ob Zusammenstoß, ob Auffahren auf eine Mine und Bewegung des Panzers, Verzweigung zu den jeweiligen Steuerungsabfragen

710-717: Abfrage für den roten Panzer

800: Geräusch für Bewegung grüner Panzer

801: Geräusch für Bewegung roter Panzer

1000-1001: Zerstörung des roten Panzers, Punkteverteilung

1005-1006: Zerstörung des grünen Panzers, Punkteverteilung

1010: Kollision beider Panzer

1015: Auffahren des grünen Panzers auf eine Mine

1020: Auffahren des roten Panzers auf eine Mine

1500: Explosionsgeräusch

1600-1630: Spiel zu Ende, Abfrage ob neuer Level

2000-2010: Schlußmelodie

3000: Punkte Einblendung mit "HI" in den oberen Bildschirm

Restliche Variablenliste für Robotbattle und Programmbeschreibung:

FNG(I) = Zufallszahlen für den Robotpanzer

O = Berechnung für die Zusteuerung auf den roten Panzer

V = Position des grünen Panzers + Richtung in die er fährt

V1 = Berechnung für Zusteuerung auf den roten Panzer

V2 = V2=0: Direktes Zusteuern auf roten Panzer

V2> : Zufälliges Steuern

R (0)-R (3) = Richtung in die der grüne Panzer fährt

R (4)-R (7) = Aussehen des grünen Panzers

R (8)-R (11) = Aussehen des Schusses der grünen Panzers

6: Definition der Feldvariable für die Richtungsangabe des Robotpanzers, sowie das Aussehen des Schusses

710-720: Steuerung des roten Panzers

900-920: Steuerung des grünen Panzers, Abfrage ob roter Panzer in Schußweite

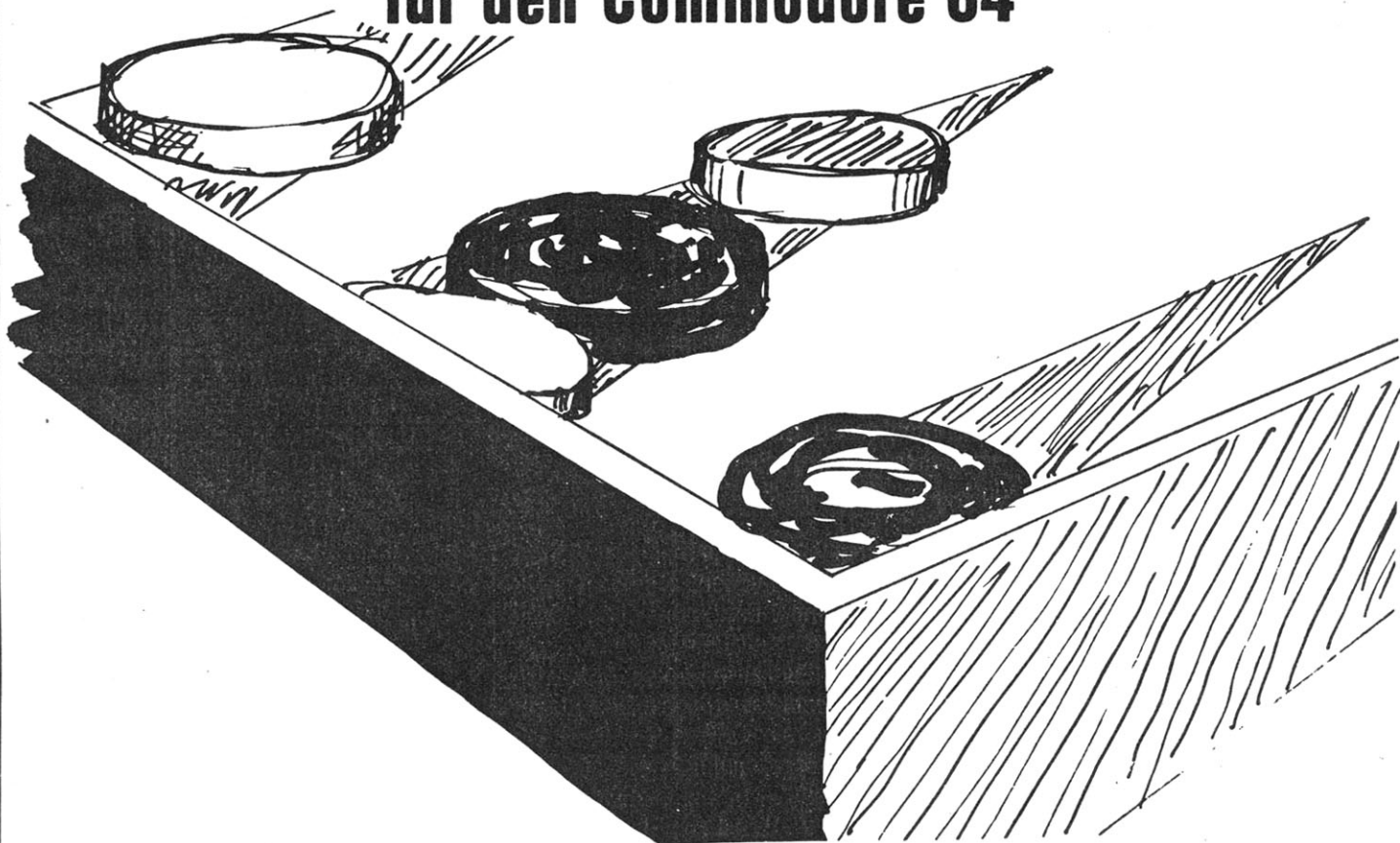
950-953: Schuß des Robotpanzers

960-970: Berechnung für die Zusteuerung auf den roten Panzer

Ansonsten gilt die Variablenliste und Programmbeschreibung von Tankbattle

Backgammon & Acey Deucey

für den Commodore 64



Nach dem Starten des Programmes erscheint auf dem Bildschirm das Backgammon-Spielfeld. Durch Drücken der Taste A kann das ACEYDEUCEY-Spielfeld aufgebaut werden. Die Spielregeln von Backgammon dürften allgemein bekannt sein. Dieselben gelten für das Spiel ACEYDEUCEY. Der Unterschied der Spiele besteht jedoch darin, daß bei ACEYDEUCEY die Spielfiguren zunächst von den Startfeldern in das Spiel eingesetzt werden müssen, während sie bei BACKGAMMON ja schon auf dem Feld verteilt sind.

Der zweite Unterschied besteht darin, daß man bei ACEYDEUCEY nach dem Würfeln einer 2,1 bzw. 1,2 einen Pasch aussuchen kann und danach erneut würfeln darf. Dadurch erhält das Spiel einen besonderen Reiz. Die Steuerung erfolgt durch zwei Joysticks. Nach dem Drücken des Feuerknopfes von Joystick 2 kann

das Spiel beginnen. Zuerst muß der linke Spieler würfeln. Mit einem Druck auf den Feuerknopf seines Joysticks setzt er den Zufallszahlengenerator in Bewegung. Durch einen zweiten Druck hält er diesen an. Seine beiden Würfe werden in der linken unteren Ecke dargestellt.

Nun ist der Spieler rechts an der Reihe, in der gleichen Weise.

Wer die höchste Quersumme gewürfelt hat, darf beginnen.

Dazu erscheint ein Kreuz am Rande des Spielfeldes, welches er mittels seines Joysticks zu dem Feld bewegen muß, von dem aus er setzen will. Dabei wird immer die linke Zahl des Wurfes zuerst gesetzt. Durch Hoch- bzw. Runterbewegen seines Joysticks kann der Spieler die Position der Zahlen verändern.

Durch Druck auf den Feuerknopf wird nun automatisch ein Stein von dem Feld um die Zahl von Feldern vagesetzt, die die

linke Ziffer aufweist. Nach zweimaligem Setzen ist der andere Spieler an der Reihe. Wird ein Pasch gewürfelt, so wird dies durch Blinken der Ziffernanzeige kenntlich gemacht und der Spieler kann nun viermal einen Stein setzen.

Sind von einem Spieler 2 oder mehr Steine geschmissen, und kann er diese nicht einsetzen, weil die in Frage kommenden Felder vom Gegner besetzt sind, so übergibt der Computer an den anderen Spieler.

Das Aussuchen des Pasches bei ACEYDEUCEY geschieht durch Rechts- und Linksbewegen des Joysticks und einem anschließenden Druck auf den Feuerknopf.

Mogeln ist bei beiden Spielarten unmöglich, da der Computer den Spielablauf überwacht.

```
100 AD=1:POKE53280,1:POKE53281,1:A$="** COPYRIGHT BY M. RUEDIG ":DIMF(25,1),P(25)
110 PRINT"
BACKGAMMON:GOSUB160:GOTO220
```



```

730 IFG<0 THEN G=0:GOTO1160
740 IFG=25 AND W0<15 OR G=0 AND W1<15 THEN 570
750 A=80+40*F(G,SP):IFG>12 THEN A=-A
760 IFF(P,SP)=0 OR F(G,SG)>1 THEN 580
770 IFABS(B)>360 THEN POKE(P)+B-320*SGN(B),81:GOTO800
780 IFAD=0 AND (P=0 OR P=25) THEN POKE(P)+B,32:GOTO800
790 POKE(P)+B,46+47*SGN(P/2-INT(P/2)):POKE(P)+B+54272,5
800 F(P,SP)=F(P,SP)-1
810 IFABS(A)>360 THEN POKE(G)+A-320*SGN(A),87:GOTO830
820 POKE(G)+A,81:POKE(G)+A+54272,F
830 F(G,SP)=F(G,SP)+1
840 IFF(G,SG)=0 THEN 890
850 B=40*X(SG):IFSG=1 THEN B=-B
860 IFSG=0 AND G>18 THEN W0=W0-1
870 IFSG=1 AND G<7 THEN W1=W1-1
880 POKE1204+640*SG+B,81:POKE55476+640*SG+B,FG:X(SG)=X(SG)+1:F(G,SG)=F(G,SG)-1
890 IFF(0,1)=15 OR F(25,0)=15 THEN 1060
900 IFSP=0 AND P<19 AND G>18 THEN W0=W0+1
910 IFSP=1 AND P>6 AND G<7 THEN W1=W1+1
920 X%=Y%:Y%=0:POKEW,32:IFX%=0 THEN POKEW+1,32:POKE(P),32:GOTO410
930 GOTO560
940 C=C+1:IFC=10 THEN POKE54272+W,1:POKE54273+W,1:RETURN
950 IFC=20 THEN C=0:POKE54272+W,F:POKE54273+W,F:RETURN
960 RETURN
970 POKE(P),32:POKE(P)+13,91:G=X%:IFSP=1 THEN G=25-X%
980 B=40*X(SP):IFP>12 THEN B=-B
990 A=80+40*F(G,SP):IFG>12 THEN A=-A
1000 Q=PEEK(J):IF(QAND2)=0 OR (QAND1)=0 THEN 1190
1010 IF(QAND16)<>0 THEN 1000
1020 IFF(G,SG)>1 THEN 1040
1030 POKE1164+720*SP+B,160:POKE55436+720*SP+B,5:X(SP)=X(SP)-1:POKE(P)+13,32:GOTO810
1040 V%=X%:X%=Y%:Y%=V%:K=K+1:POKEW,X%+48:POKEW+1,Y%+48:IFK=2 THEN K=0:Y%=0:GOTO890
1050 GOTO970
1060 PRINTTAB(9)"COPYRIGHT BY M. RUEDIGER":FORI=1186TO1826STEP40:POKEI,32:NEXT
1070 POKE1950,32:POKE1949,32:POKE1977,32:POKE1980,32
1080 J=36*SP:FORI=1306+JTO1706+JSTEP80:READA,B:POKEI,A:POKE54272+I,14:POKEI+36-J,B
1090 POKE54308+I,10:NEXT
1100 FORI=55418TO56098STEP80:POKEI,14:POKEI+36,10:FORJ=1TO40:NEXT:NEXT
1110 FORJ=1TO300:NEXT:FORI=55418TO56098STEP80:POKEI,1:POKEI+36,1:FORJ=1TO40:NEXT:NEXT
1120 FORI=1TO70:NEXT:IFPEEK(56321)=239 AND PEEK(56320)=111 THEN RUN
1130 GOTO1100
1140 FORI=17TOP-1:IFF(I,0)>0 THEN 570
1150 NEXT:GOTO740:DATA23,12,9,15,14,15,14,19,5,5,18,18
1160 FORI=P+1TO6:IFF(I,1)>0 THEN 570
1170 NEXT:GOTO740
1180 IF(PEEK(J)AND16)=0 THEN NONSP+1GOTO1220,1250
1190 IFX%=0 OR Y%=0 THEN 1210
1200 V%=X%:X%=Y%:Y%=V%
1210 V%=PEEK(W):POKEW,PEEK(W+1):POKEW+1,V%:GOTO560
1220 FORI=1TO24-X%:IFF(I+X%,1)<1 THEN 570
1230 NEXT:FORI=1TO24-Y%:IFF(I+Y%,1)<1 THEN 570
1240 NEXT:Y%=0:GOTO920
1250 FORI=X%+1TO24:IFF(I-X%,0)<1 THEN 570
1260 NEXT:FORI=Y%+1TO24:IFF(I-Y%,0)<1 THEN 570
1270 GOTO1240
1280 POKEW,X%+48:POKEW+1,X%+48:Q=PEEK(J):IF(QAND4)=0 THEN X%=X%-1:IFX%=0 THEN X%=1
1290 IF(QAND16)=0 THEN FORI=1TO60:NEXT:RETURN
1300 IF(QAND8)=0 THEN X%=X%+1:IFX%=7 THEN X%=6
1310 FORI=1TO150:NEXT:GOTO1280
READY.

```

ABQ

für den VC-20 + 16K

Vor dem Laden und Eingeben sollte man folgende Pokes eingeben:

POKE 44,28
POKE 7168,0
NEW

Spielbeschreibung:

Die Nachricht hat sich wie ein Lauffeuer über das Land verbreitet.

Ein Angriff der Hungulus ist geplant!

Sie wurden mit der ehrenvollen Aufgabe betraut, die Welt zu retten!!!

Schon tauchen am Horizont die ersten Invasoren auf. Der Chef der Hungulus leitet persönlich mit seinem Raumschiff den Angriff ein. Ihr Computer in der Zentrale meldet Ihnen, daß alle drei Kanonen mit jeweils 9 Schuß geladen und zum Kampf bereit sind. Sie haben gerade noch Gelegenheit, den Joystick zu ergreifen und Ihren Mut zu beweisen.

Programmbeschreibung:

Ihre Aufgabe ist es, vier Städte, die am unteren Bildschirmrand aufgebaut sind, zu verteidigen. Zu diesem Zweck steht Ihnen ein Hochleistungslager zur Verfügung. Von oben kommen die Invasoren.

Man muß nun sein Zielkreuz kurz vor einem Invasor setzen und den Feuerknopf drücken (siehe Programm). Bei einem Treffer entsteht eine Wolke, die sich verflüchtigt.

Das Abschließen eines Hungulus bringt 100 Punkte x Bonus, das Raumschiff 500 Punkte x Bonus, jeder Schuß, der übrig bleibt 10 x Bonus, jede gerettete Stadt 100 Punkte x Bonus, jeder Blitz, der am Boden aufschlägt 100 Punkte x Bonus. Nach jedem Angriff werden die Kanonen neu geladen.

Grafik:

Selbstverständlich werden selbstdefinierte Zeichen verwendet. Als eine Besonderheit soll erwähnt werden, daß manche Zeichen in Multicolor-Modus erstellt wurden.

Eine "Freeze"-Funktion ist ebenfalls vorhanden. Diese erreicht man durch Drücken des Feuerknopfes und Bewegen des Joysticks nach unten. Als Zeichen für "Freeze" geht an den Kanonen das Licht aus.

Mit "F1" wird das Spiel fortgesetzt. Bei jedem Aufschlag eines Blitzes wird das

Licht an den Kanonen ebenfalls schwächer.

Neustart nach "Game Over" mit "F3".

Programmaufbau:

00005-00011: INIT Game
00100-00120: Kanonen printen
00130-00145: Stadt prüfen
00150-00160: INIT. Blitz und Ufo
00250: Hauptschleife
00260-00430: Scoreauswertung
00500-00550: Bewegung Blitz
00560-00600: Treffer
00700-00760: Bewege Ufo
01000-01150: Joystick und Fadenkreuz
02000-02020: Schuß
03000-03030: Stadtaufbau
03100-03130: Ramtop
03900-04020: MC Programm einlesen
04030-04080: Zeichen einlesen
04090-04980: Zeichendatas
05000-05003: Freeze
06000-06007: Musik
07000-07020: Explosion
10000: Datas (Laser & Ufo)
10099-10370: Anleitung
20000-20003: Data MC-Routine
40000-42240: Sonderzeichen in REMs

```
0 REM **ABM (C) 1984 BY H. SCHADEN &
M.KOSTOPULOS**
1 POKE830,4:POKE831,0:POKE832,0:GOTO3100
2 GOSUB4030:POKE36869,205
3 GOSUB10100:GOSUB380
5 [IMS*(12),T*(12):TN=36874:FORT=0T012:R
[ADT*(T):NEXT
6 GOSUB3900:FORT=0T04:O*(T)=1:NEXT:O*=1:
SW*=1:SC=0:MY=0:GOSUB3030:GOTO250
10 CT*=5:X=10:Y=7:LP*=4096+X+Y*22:PY=LP*
FORT=0T03:B*(T)=9:NEXT:BB*=32:POKE198,0
11 K=33792
99 REM
100 POKE36879,239:POKE646,14:PRINT"00000
0000000000000000R4L" R01L
R4L
110 PRINT"5R011111111":POKE646,14:PRIN
T"23011111111":POKE646,14:PRINT"5",
120 PRINT"R 9 7 9 6 9L":POK
E4601,160:POKE37855+505,7
30 O*=4538:FORT=0T03:TT=T*4:IFTT>4THENT
T=TT+3
140 IFD*(T)>0THENGOSUB3000
145 NEXT
150 FORT=0T05:P(T)=4075+RND(1)*22:R(T)=R
ND(1)*Y1.2-.6:IFABS(R(T))>.5THENS(T)=1*SG
```

ERKLÄRUNG DER SONDERZEICHEN:

LINE 100
 "S" CLR
 "Q" CRSR DOWN
 "R" RVS ON
 "4" SHIFT+H
 "L" RVS OFF
 "0" COMMODORE+A
 "1" COMMODORE+E

LINE 110
 "5" COMMODORE+J
 "L" CTRL+6
 "O" COMMODORE+X
 "J" SHIFT+J
 "2" COMMODORE+R
 "3" COMMODORE+W

LINE 120
 "9" A REAL NINE (9)
 "7" COMMODORE+Y
 "6" COMMODORE+L

LINE 310
 "L" CTRL+3

LINE 350
 "L" F3

LINE 356
 "L" SHIFT+X
 "a" SHIFT+A
 "b" SHIFT+B

LINE 10110-10114
 "c" - "z" SHIFT + C - Z

LINE 10120
 "c" SHIFT + C

LINE 10350
 "q" SHIFT + Q
 "L" CRSR LEFT

```

N(R(T))
152 X%=4162+INT(RND(1)*7)*22:UF%=1: I%=IN
T(RND(1)*2)*2-1
154 IFSGN(I%)=-1THENT%(12)=219:T%(10)=22
1:GOTO160
156 T%(12)=221:T%(10)=219
160 R(T)=R(T)+22:NEXT N%=6:E%=RND(1)*10+
4:RETURN
250 GOSUB1000:FORG=1TOSW%:GOSUB500:GOSUB
700:NEXT:GOTO250
260 IFM%>0THENM%=M%-1:GOSUB150:RETURN
270 POKETN+3,0:POKETN+4,175:F%=B%(1)+B%(
2)+B%(3):FORT=1TOF%:IFF%>0THENPOKE37998+
T,0:POKE4206+T,30
271 SC=SC+10*SW%:GOSUB370:POKETN,230:FOR
R=1TO20:NEXT:POKETN,0:NEXT
280 SK%=20:F%=0:O%=4275:FORT=0TO3:TT=T*4
:IFD%(T)>0THENGOSUB3000:SC=SC+100*SW%:GO
SUB370:F%=1
290 POKETN,240:FORR=0TOSK%:NEXT:POKETN,0
:NEXT
300 G%=0:M%=U%:U%=U%+1:IFU%>4THENSW%=SW%
+1:U%=1:M%=0
310 PRINT"0011111BONUS X"SW%:FORT=0TO20
00:NEXT:IFF%>0THENPOKETN+4,160:GOTO3030
340 POKETN+3,230:FORT=30TO0STFP-.5:POKET
N+4,T/2+160:POKETN+5,42
341 FORTT=1TO70:NEXT:POKETN+5,25:FORTT=1
TO70:NEXTTT,T:POKETN+5,236:POKETN+3,0
345 PRINT"1111111GAME OVER!"
350 GETA$:IFA%<>"I"THEN350
355 PRINT"A NEW GAME IS CREATET!"
356 PRINT"000R-T Rar Rbl - SOFTWARE"
360 RUN2
370 PRINT"0000000000000011111SC ="SC:GOSU
B380:GOTO390
380 K1%=RIGHT$("000"+STR$(PEEK(830)),3)
381 K2%=RIGHT$("000"+STR$(PEEK(831)),3)
382 K3%=RIGHT$("000"+STR$(PEEK(832)),3)
383 HS=VAL(K1%+K2%+K3%):RETURN
390 IFSC>HSTHENHS=SC:SK%=10:HS%=RIGHT$("
000000"+STR$(HS),6):GOTO400
395 GOTO430
400 POKE830,VAL(LEFT$(HS$,2))
410 POKE831,VAL(MID$(HS$,2,2))
420 POKE832,VAL(RIGHT$(HS$,2))
430 PRINT"0011111HI ="HS:RETURN
500 U%-U%+1:IFU%>CT%THENU%=0
515 IFP(U%)<4075THENRETURN
524 R=PEEK(P(U%)):IFR=1050RR=2330RR=2230
RR=950RR=1710RR=160THEN600
525 R=PEEK(P(U%)+R(U%)):IFR=1050RR=2230R
R=2330RR=950RR=1710RR=160THEN600
526 P(U%)=P(U%)+R(U%):IFP(U%)>4536THEN55
0
527 IFP(U%)>4360ANDP(U%)<4360+SW%*22ANDU
%>0THEN529
528 GOTO530
529 IFP(U%-1)=0THENP(U%-1)=P(U%):R(U%-1)
=-R(U%)+44:N%=N%+1
530 C=(P(U%)-INT(P(U%)))*8:IFABS(R(U%-2
2)=1THENC=8-SGN(R(U%-23))
540 POKEP(U%),T*(C):RETURN
550 POKEP(U%),32:SC=SC-100*SW%:J%=P(U%)-
4536:IFJ%>21THENJ%=J%-22
560 IFJ%=0THENB%(1)=0:POKE4581,176
561 GOSUB7000:POKE4558+J%,86:POKE4558+J%
+K,2:IFJ%>1ANDJ%<5THEND%(0)=0
562 IFJ%>5ANDJ%<9THEND%(1)=0
563 IFJ%>9ANDJ%<12THEND%(2)=0:POKE4590,1
76
564 IFJ%>12ANDJ%<16THEND%(2)=0
565 IFJ%>16ANDJ%<20THEND%(3)=0
566 IFJ%>21THEND%(3)=0:POKE4600,176
600 SC=SC+100*SW%:N%=N%-1:FORT=P(U%)TO40
96STEP-R(U%):POKET,32:NEXT:P(U%)=0:GOTO3
040
700 IFU%=0THENRETURN
709 I%=I%+SGN(I%):L%=X%+I%:IFABS(I%)=20T
HEN760
710 IFABS(I%)=E%THENFORT=6TO8:P(T)=L%+22
.5:R(T)=T+15:NEXT:CT%=8:N%=N%+3:POKEL%+2
2,93
740 FORT=0TO32:TT=L%+SGN(I%)*T:R=PFEK(TT)
742 IFR=2330RR=750RR=2230RR=1050RR=950RR
=1600RR=1710THENSC=SC+500*SW%:GOTO760
750 POKETT,T%(T+10):POKETT+K,14:NEXT:POK
EL%-SGN(I%),32:POKEL%-SGN(I%)+K,1:RETURN
760 FORT=-1TO2:POKEL%+SGN(I%)*T,32:POKEL
%+SGN(I%)*T+K,1:NEXT:UF%=0:RETURN
999 REM *****JOYSTICK*****
1000 TA=USR(0):C%=0
1005 IFTA=130THENGOSUB5001

```

```

1010 IFTA=1THENY=Y-1
1020 IFTA=4THFNX=X-1
1030 IFTA=8THFNX=X+1
1040 IFTA=2THFNX=Y+1
1050 IFTA=10THFNX=X+1:Y=Y+1
1060 IFTA=6THFNX=X+1:Y=Y-1
1070 IFTA=6THFNX=X-1:Y=Y+1
1080 IFTA=5THFNX=X-1:Y=Y-1
1081 IFX>20THENX=20
1082 IFX<0THFNX=1
1083 IFY>18THENY=18
1084 IFY<1THENY=1
1085 C%=ABS((TA=132)+(TA=129)*2+(TA=136)
*3):IFC%>THENGOSUB2000:POKE4581+(C%-1)*9.
5,B%(C%)+176
1090 P%=4096+X+Y*22:POKELP%,BB%:G%=G%-1:
IFG%>0THENPOKETN+4,G%+160
1100 GOSUB3040:R%=R%+1:IFR%>12THENR%=0:I
FN%=0THEN1140
1120 IFS%(R%)>4096THENFORT=-1TO1:FORTT=-
1TO1:GOTO1125
1124 GOTO1130
1125 F%=S%(R%)+T*22+TT:POKEF%,32:NEXTTT,
T:S%(R%)=4096:GOSUB3040
1130 LP%=P%:RETURN
1140 IFM%=0THENFORT=0TO12:R%=T:GOSUB1120
:NEXT
1150 LP%=P%:GOTO260
2000 IFB%(C%)<1THENRETURN
2010 S%(R%)=P%:FORT=-1TO1:POKES%(R%)+T*2
2,160:POKES%(R%)+T,160:NEXT:BB%=171:B%(C
%)=B%(C%)-1
2020 POKES%(R%)-23,233:POKES%(R%)-21,223
:POKES%(R%)+23,105:POKES%(R%)+21,95:G%=1
5:RETURN
3000 FORC=0%+TT+KT00%+TT+3+K:POKEC,4:NEX
T
3001 POKE0%+TT+22,237:POKE0%+TT+22+K,14:
POKE0%+TT+23,238:POKE0%+TT+23+K,6:POKE0%
+TT+24,239
3002 POKE0%+TT+24+K,14
3010 POKE0%+TT+K,14:POKE0%+TT+1+K,14:POK
E0%+TT+2+K,14
3020 POKE0%+TT,234:POKE0%+TT+2,236:POKE0
%+1+TT,235:RETURN
3030 GOSUB10:POKETN+4,161:POKETN+3,230:R
ETURN
3040 R=PEEK(P%):IFR<>219ANDR<>220ANDR<>2
21ANDR<>43THENRB%=PFEK(P%):POKEP%,43
3050 RETURN
3100 AC=PEEK(55)+256*PEEK(56)-53
3110 POKE55,AC-(INT(AC/256)*256)
3120 POKE56,AC/256:CLR
3130 GOTO2
3900 AC=PEEK(55)+256*PEEK(56)+2:IFPEEK(A
C)=173THENRETURN
4000 FORSH=0TO49:READKM%:POKESH+AC,KM%:N
EXT
4010 POKE0,76:POKE1,AC-(INT(AC/256)*256)
4020 POKE2,AC/256:RETURN
4030 IFPEEK(5120)=PEEK(32768)THEN4050
4040 FORJ=0TO2047:POKEJ+5120,PEEK(J+3276
8):NEXT
4050 READX:IFX=-1THENRETURN
4060 X=X*8
4070 FORJ=5120+XT05120+X+7:READA:POKEJ,A
:NEXT
4080 GOTO4050
4090 REM ***** USER DEFI
NED CHARACTERS *****
*****
4092 REM
4095 REM ***A C B***
4100 DATA192,60,126,255,231,195,255,255,
195
4110 DATA193,126,254,224,192,192,224,254
,126
4112 DATA81,66,24,36,66,126,66,66,66
4115 REM ***CROSS***
4120 DATA194,254,255,195,206,206,195,255
,254,43,24,24,24,231,231,24,24,24
4130 DATA171,231,231,231,24,24,231,231,2
31
4135 REM ***C)***
4140 DATA67,126,129,157,161,161,157,129,
126
4145 REM ***ABM***
4150 DATA195,0,0,0,0,0,3,6,6
4160 DATA196,0,0,0,0,0,192,96,96
4170 DATA197,12,12,12,24,24,24,24,48
4180 DATA198,48,48,48,24,24,24,24,12
4190 DATA199,48,63,63,48,96,96,96,96
4200 DATA200,12,252,252,12,6,6,6,6
4210 DATA201,96,96,192,192,192,192,192,1

```


ID-CHANGER

für den C-64

Mit diesem kleinen Programm ist es möglich, den Namen oder die ID einer Diskette zu ändern. Das Programm ist menuegesteuert.

Beschreibung:

MENÜ:

- 81) - Name lesen
- 82) - Name ändern
- (3) - ID lesen
- (4) - ID ändern

(5) - ENDE

Bei Drücken der Taste "1" wird der Name der Diskette gelesen. Weiter (J/N) N = Programmende

Bei Drücken der Taste "2" wird ein neuer Diskettenname verlangt. Das Programm nimmt nur 16 Buchstaben von den eingegebenen. Weiter (J/N) N = Programmende

Bei Drücken der Taste "3" wird der ID

(Identifizier) gelesen. Weiter (J/N) N = Programmende

Bei Drücken der Taste "4" wird ein neuer ID verlangt. Das Programm nimmt nur zwei Werte von den eingegebenen. Weiter (J/N) N = Programmende

Bei Drücken der Taste "5" = Programmende.

```

10 PRINT " "
20 PRINT
30 PRINT
40 PRINT
50 PRINT "WAELLEN SIE GEWUENSCHTE FUNKTION: "
60 PRINT
70 PRINT "      (1)-NAME LESEN"
80 PRINT "      (2)-NAME AENDERN"
90 PRINT "      (3)-ID LESEN"
100 PRINT "      (4)-ID AENDERN"
110 PRINT "      (5)-ENDE"
120 GETX$:IFX$<"1"ORX$>"5"THEN 120
130 ON VAL(X$)GOSUB150,290,450,590,750
140 PRINT " ":GOTO 20
150 OPEN1,8,15
160 OPEN2,8,2, " #"
170 PRINT#1, "U1 2 0 18 0"
180 PRINT#1, "B-P 2 144"
190 PRINT "NAME: "
200 FOR I=1 TO 16
210 GET#2,A$:IF A$=CHR$(160)THEN 230
220 PRINTA$;:NEXT
230 CLOSE2:CLOSE1
240 PRINT " "
250 PRINT "WEITER (J/N)"
260 GET Z$:IF Z$<>"J"ANDZ$<>"N"THEN 260
270 IF Z$="N" THEN 750
280 RETURN
290 OPEN1,8,15
300 OPEN2,8,2, " #"
310 PRINT#1, "U1 2 0 18 0"
320 INPUT "NEUER NAME: ";B$
330 B$=LEFT$(B$,16)
340 PRINT#1, "B-P 2 144"
350 IF LEN(B$)<16 THEN B$=B$+CHR$(160):GOTO 350
360 PRINT#2,B$;
370 PRINT#1, "U2 2 0 18 0"
380 CLOSE2
390 PRINT#1, "ID":CLOSE1
400 PRINT " "
410 PRINT "WEITER (J/N)"
420 GETZ$:IFZ$<>"J"ANDZ$<>"N"THEN 420
430 IFZ$="N"THEN 750
440 RETURN
450 OPEN1,8,15
460 OPEN2,8,2, " #"
470 PRINT#1, "U1 2 0 18 0"
480 PRINT#1, "B-P 2 162"
490 PRINT "ID: "
500 FOR I=1 TO 2
510 GET#2,C$:IF C$=CHR$(160)THEN 530
520 PRINTC$;:NEXT
530 CLOSE2:CLOSE1
540 PRINT " "

```

```

550 PRINT "WEITER (J/N)"
560 GETZ$:IFZ$<>"J"ANDZ$<>"N"THEN 560
570 IF Z$="N" THEN 750
580 RETURN
590 OPEN1,8,15
600 OPEN2,8,2, " #"
610 PRINT#1, "U1 2 0 18 0"
620 INPUT "NEUER ID: ";D$
630 D$=LEFT$(D$,2)
640 PRINT#1, "B-P 2 162"
650 IF LEN(D$)<2 THEN D$=D$+CHR$(160):GOTO 650
660 PRINT#2,D$;
670 PRINT#1, "U2 2 0 18 0"
680 CLOSE2
690 PRINT#1, "ID":CLOSE1
700 PRINT " "
710 PRINT "WEITER (J/N)"
720 GETZ$:IF Z$<>"J"ANDZ$<>"N"THEN 720
730 IF Z$="N" THEN 750
740 RETURN
750 PRINT " "
760 PRINT " "
770 PRINT " "
780 END
READY.

```

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 Kassette: 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 Bedienungsanleitung 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	

CM-LEXIKON

Die Arbeit mit dem Computer macht viel Spaß, das wissen wir. Jeder, der selbst Programme schreiben will, weiß aber auch, daß es manchmal nicht ohne Probleme funktioniert. Für viele Einsteiger ist zunächst einmal die Computersprache ein Handicap. Da wird von ROM und RAM, PEEK und POKE, Bit und Bytes geredet, für den Laien sicher nicht einfach zu verstehen. Wir möchten den Slang der Computereeks etwas durchsichtiger machen und beginnen heute mit einem Wörterbuch. In jeder Ausgabe von "Compute mit" werden wir in alphabetischer Reihenfolge die wichtigsten Begriffe der Computerwelt erklären. Bleiben Sie am Ball, dann können Sie mitreden.

ADA:

Höhere Programmiersprache, wurde im Auftrag des amerikanischen Verteidigungsministeriums entwickelt und von der EG zur Normsprache erklärt.

Algorithmus:

Alles was systematisch abläuft, folgt einem Algorithmus.

Man kann Algorithmus auch als Verfahren bezeichnen. Wird dieses Verfahren programmiert, so entsteht ein Programm.

Ohne einen eindeutigen Algorithmus ist ein Programm meist unlogisch und damit nicht funktionsfähig.

Abakus:

Ist das älteste Rechengesetz der Menschheit. Je nach Ausführung werden fünf oder sieben Kugeln auf Stäbchen hin- und hergeschoben. Aufgrund der Position der Kugeln ergibt sich dann der Wert. Der Abakus ist heute in Indien, Rußland und Asien aufgrund seiner Einfachheit noch verbreitet.

Akustikkoppler:

Hilfsmittel zur Datenfernübertragung. Der Akustikkoppler sendet und empfängt Daten von anderen Rechnern und wandelt die Töne in Computersprache um. Die Datenübertragung läuft über das

Fernmeldenetz, die Koppler müssen eine FTZ-Genehmigung haben.

ASCII-Code:

Der ASCII (sprich: Aßki) ist die amerikanische Fassung des weltweit verbreiteten ISO 7-bit Code. Es werden Zeichen mit je 7 bit und einem Kontrollbit dargestellt. Mit diesen 7 Variationsmöglichkeiten lassen sich 128 Zeichen darstellen.

Assembler:

Programmiersprache, die jeder Computer intern zur Verfügung hat. Assembler ist eine maschinenorientierte Programmiersprache, da ein Assemblerprogramm auf einem anderen Computertyp nicht läuft. Die Basis sind die Zahlen 0 und 1.

ANSI:

steht für American National Standards Institute.

Ziel der Behörde ist es, für die Industrie einen einheitlichen Standard festzulegen, vergleichbar mit der deutschen DIN-Norm.

Es wird eine größere Kompatibilität der Hard- und Software angestrebt.

Analog-Digital-Wandler:

elektronischer Baustein, der die analoge Umwelt in digitale, d.h. computerver-

ständige Form umwandelt.

So können z.B. Temperaturen, Feuchtigkeit, Bilder etc. in digitale Form gebracht werden. Der Computer kann diese Informationen verstehen und bearbeiten.

Anwender-Programm:

Mit Hilfe eines Anwenderprogramms kann der Benutzer spezifische Probleme lösen. Dabei sind keine Programmierkenntnisse notwendig, der Benutzer bekommt genaue Anweisungen, z.B. durch Menuesteuerung.

Textverarbeitung, Kalkulation, Dateiverwaltung sind einige Anwenderprogramme.

ALGOL:

höhere Programmiersprache für den technisch-wissenschaftlichen Bereich. Verliert immer mehr an Bedeutung, da inzwischen andere Programmiersprachen leistungsfähiger sind.

Austesten:

Softwaregesteuerter Selbsttest für Computer.

Das Programm überprüft sämtliche Register und Peripherie des Computers (z.B. Farbe, Ton, Tastatur).

KFC-Super

Toolkit und Monitor im Eprom

Auf dem Markt sind inzwischen mehrere Basicerweiterungen für den C-64 erhältlich z.B. Simon's Basic, Ex.-Basic Level II usw.).

Diese Zusätze werten das schwache Basic des C-64 erheblich auf, einer der interessantesten dürfte wohl das steckbare Modul, "KFC-Super", mit eingebauter Centronics-Schnittstelle und eingebautem Maschinensprache-Monitor sein.

KFC-Super wird in den Adreßbereich \$8000-\$9FFF geladen und besteht aus folgenden Funktionseinheiten:

- Programmierhilfen
- Schnelle Kassettenroutinen
- komfortable Floppy-Befehle
- Centronics-Druckerschnittstelle am User-Port
- Maschinensprache-Monitor
- Repeat-Funktion der Tasten (abschaltbar)
- Escape- und Pause-Taste

Das Modul besteht aus einer offenen Platine, zwei Steckplätze für weitere Eproms sind frei.

Mit den auf der Platine befindlichen DIP-Schaltern können die Eproms angesteuert oder ganz abgeschaltet werden. So kann auch bei aufgestecktem Modul im Normalzustand gearbeitet werden. Sehr nützlich auch die RESET-Taste, mit der verlorengegangene Speicherinhalte noch gerettet werden können.

Direkt am User-Port anzuschließen ist jeder Drucker mit Centronics-Schnittstelle, das teure Interface entfällt. Man benötigt lediglich ein geeignetes Verbindungskabel.

Je nach Anwahl der Primäradressen wird im Text-, List- oder Grafikmodus ausgedruckt.

KFC-Super ist eine Alternative zu bisherigen Basicerweiterungen, allerdings wird der freie Speicherplatz um ca. 8 KByte reduziert (30719 Bytes free). Die zusätzlichen Befehle sind eine tolle Hilfe für Anwender, mit dem eingebauten Maschinensprache-Monitor wird KFC-Super zur Kompakterweiterung.

Befehlsübersicht:

AUTO: Automatisches Numerieren, Vorgaben möglich.

RENUM: Automatisches Umnummerieren,

Vorgaben möglich.

DEL: Löschen von Basic-Zeilen oder Zeilen-Gruppen.

TRACE: Zeilennummern-Anzeige beim Programmablauf.

FIND: Suchen nach Zeichen und Begriffen.

DUMP: Anzeige von Variablen und ihren Werten.

HELP: Anzeige der Stelle, an der ein Programmfehler vorkam.

OLD: Rettet nach einem NEW oder RESET das Basic-Programm.

CATALOG: Zeigt Disk-Verzeichnis ohne Programm-Zerstörung.

STATUS: Anzeige einer Floppy-Fehlermeldung.

DLOAD: Programm von Diskette laden.

DVERIFY: Programm von Diskette vergleichen.

DMERGE: Programm von Diskette an bestehendes Programm anhängen.

DSAVE: Programm auf Diskette speichern.

DISK (dl): Vereinfachte Befehlsübergabe in die Floppy.

dl'I': Floppy-Gerät initialisieren.

dl'V': Diskette bereinigen (Collect).

dl'N:NAME,ID': Diskette neu formatieren mit ID.

dl'N:NAME': Diskette neu formatieren ohne ID.

dl'S:NAME': Programm NAME wird auf der Diskette gelöscht.

dl'R:N= A': Programm A wird in N umbenannt.

dl'C:N= A': Programm A wird als N auf die gleiche Diskette kopiert.

PUT: Schnelles SAVE auf Kassette (mit VC SUPER 20 kompatibel).

GET: Schnelles LOAD von Kassette (mit VC SUPER 20 kompatibel).

COMP: Schnelles VERIFY von Kassette.

MERGE: Programm von Kassette anhängen (im Schnellmodus).

PAUSE: SHIFT-Taste hält Listing oder Programm-Ablauf an.

REPEAT: Schaltet REPEAT nach OFF wieder ein.

ESC: Aktivieren einer ESCAPE-Taste.

OFF: Schaltet PAUSE, REPEAT und ESC aus.

MMON: Maschinensprache-Monitor (4 K) einschalten.

MOVE: Programm aus dem \$9000 ROM-Bereich auf \$C000 kopieren.

VIEW: Speicherbelegung und ihre Aufteilung sind angezeigt.

ORDER: Alle 'VC SUPER'-Befehle werden angezeigt.

ORDER*: Alle BASIC-Befehle werden angezeigt.

KILL: Schaltet VC-Super aus.

Der Maschinensprache-Monitor

Der Befehl MMON oder SYS 36864 startet den Maschinensprache-Monitor im Adreßbereich \$9000-9FFF.

Außer den üblichen Maschinensprache-Befehlen beinhaltet dieser Monitor noch eine SPRITES-Editerroutine.

Alle Darstellungen von Speicherbereichen lassen sich mit Hilfe des Cursor verschieben, die Fortsetzung erfolgt am unteren bzw. oberen Bildschirmrand.

Die Monitorbefehle:

A: Assemblieren

B: Bits (Sprite-Edit)

C: Speicherbereichsinhalte vergleichen

D: Disassemblieren

F: Füllen eines Adreßbereichs mit einem vorgegebenen Byte

G: Starten eines MC-Programmes

H: Durchsuchen eines Adreßbereichs nach vorgegebenen Bytes

L: Laden von Programmen oder Daten

M: Listen der Speicherinhalte (HEX-Ausgabe)

N: Verschieben eines Adreßbereichs

R: Anzeigen der Registerinhalte

S: Abspeichern der Inhalte eines Adreßbereichs

T: Verschieben eines Adreßbereichs im Speicher

X: Rücksprung zum Basic

: Umrechnung Dezimal-Hexadezimal

\$: Umrechnung Hexadezimal-Dezimal

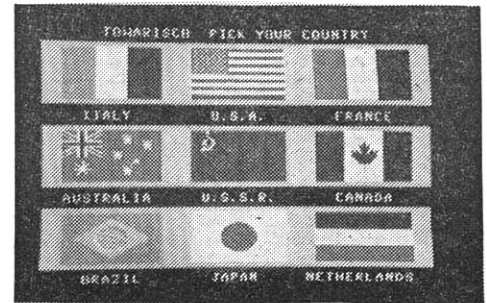
Der Preis für KFC-Super ist mit DM 198,- für den C-64 nicht zu hoch, vergleicht man Preis und Leistung mit anderen Basicerweiterungen. Die offene Platine ist zwar nichts für Ästheten, KFC-Super beseitigt aber viele Schwachstellen der C-64, die von Anwendern immer wieder beklagt werden.

Das "KFC-Super"-Steckmodul ist ebenfalls für den VC-20 erhältlich. (sr)

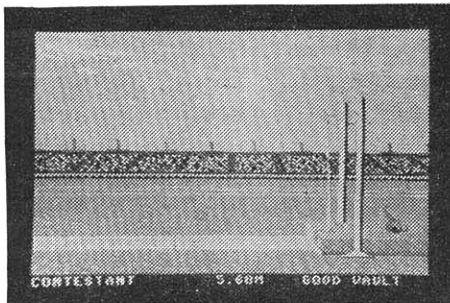
Nachruf auf Olympia

Go for Gold!

Tag und Nacht war es in den letzten Wochen aus den Olympia-Berichterstattungen zu hören. Inzwischen gibt es auch die Olympiade am Heimcomputer, wichtigstes Utensil ist der Joystick. Es gilt jetzt, den Medallensegen zuhause zu empfangen, manche Joystickprofis kommen nur noch zu der lapidaren Feststellung: Wer ist denn schon Carl Lewis?

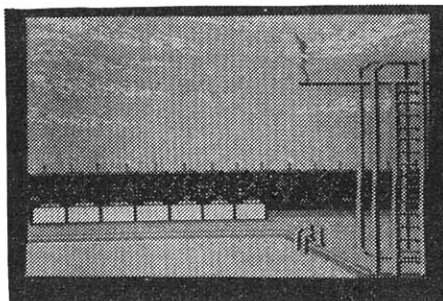


Summer Games von Epyx ist eine Olympiade von acht Disziplinen, an der ebenso viele Wettkämpfer teilnehmen können. Nach dem Laden der Diskette die erste Überraschung: Da kommt zeichentrickreif ein Fackelträger und entzündet das olympische Feuer, während weiße Friedenstauben über den Bildschirm fliegen. Anschließend muß Name und Nationalität eingegeben werden, der C-64 spielt sogar die Nationalhymnen. Die Epyx Summer Games sind eröffnet. Als erste Disziplin

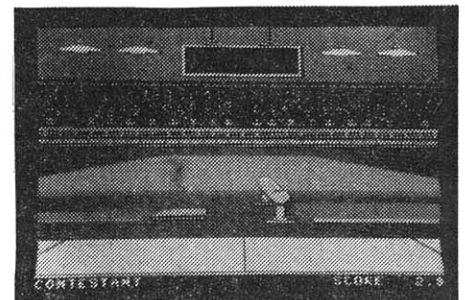


steht der Stabhochsprung an, je 3 Versuche pro Höhe sind erlaubt. Der Joystick muß gefühlvoll bewegt werden, um das Männchen über die Latte zu manövrieren. Die Anfangshöhe beträgt 4,00m; kann aber bis zu einer Höhe von 6,90m ausgebaut werden. Bestechend hier, wie in allen übrigen Disziplinen, der Bewegungsablauf und die grafischen Raffinessen. Der Realitätsbezug ist ohne Zweifel vorhanden. Dem Sieger einer jeden Disziplin winkt als Belohnung die Nationalhymne. Jetzt geht's auf den Sprungturm, als nächster Wettkampf steht das Kunstspringen auf dem Programm.

Mit dem Joystick wird ein Mädchen durch die Luft bewegt, aber aufgepaßt: Wie im echten Wettkampf, muß auch hier die Eintauchphase korrekt sein, d.h. es darf nur wenig Wasser spritzen. Anschließend gibt es für jeden Sprung Punkte von den Kampfrichtern, die mit dem



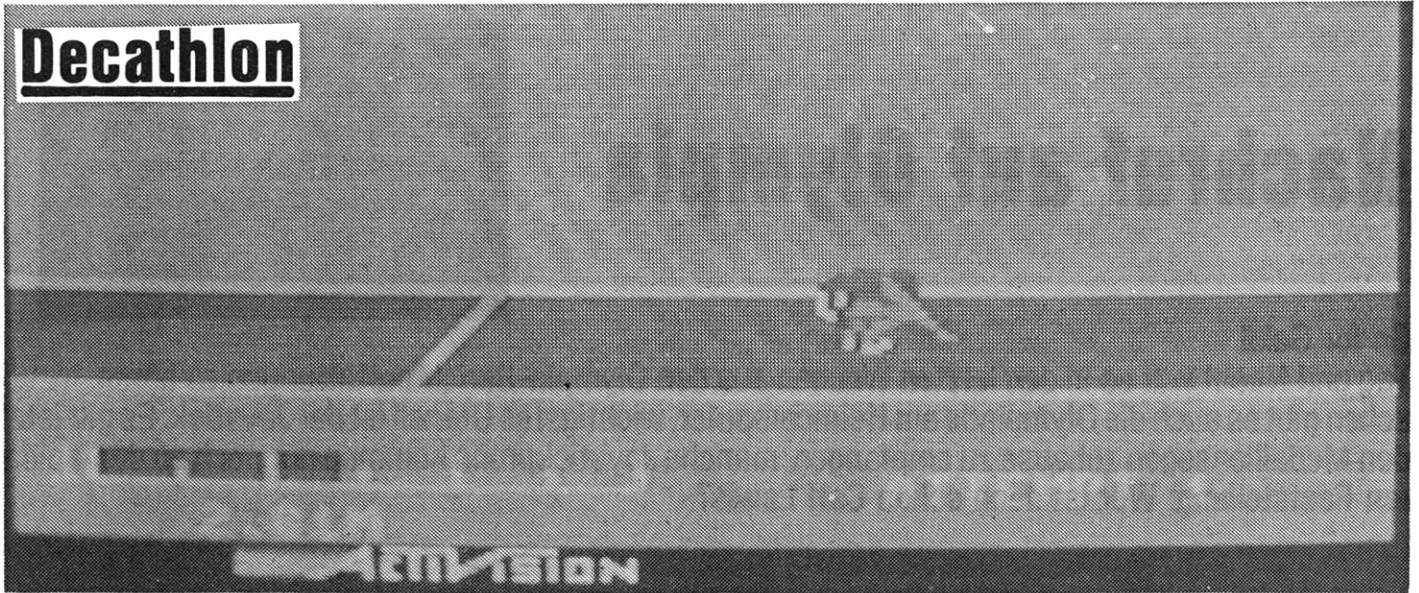
Schwierigkeitsgrad des Sprunges multipliziert werden. Blitzschnelles Umdenken auf andere Disziplinen ist gefragt, allerdings dauert das Nachladen einer Disziplin etwa eine Minute. Gelegenheit zur Erholung ist also reichlich. Der 4x400m Staffellauf steht auf dem Programm, hier kommt es auf Ausdauer an. Ein Konditionsbalken am unteren Bildschirmrand bestätigt den jeweiligen Zustand des Läufers. Ist also keine Energie mehr vorhanden, muß das "Gas" weggenommen werden. Anders dagegen beim 100m Sprint. Hier wird die Kondition des Joystick-Armes gefordert, nur durch schnellstmögliches Hin- und Herbewegen des Joysticks ist der Läufer in das Ziel zu bringen. Dagegen erfordert der Umgang mit dem Steuerknüppel beim Pferdsprung wieder Geschick. Die Turnerin muß heil wieder auf dem Boden landen



um gute Punkte zu erreichen. Anschließend geht's wieder in das Wasser, die 4x400m Freistil-Staffel steht zum Start. Auch hier eine hervorragende Mischung aus Grafik und Bewegung, die Schwimmer gleiten regelrecht über das Wasser. Das aber auch nur solange, wie der Feuerknopf des Joysticks in regelmäßigen Abständen betätigt wird. Das Drücken des Feuerknopfes simuliert hier das Atmen des Schwimmers. Nachdem die Staffeltwettbewerbe beendet wurden, geht's in die Einzeldisziplin des 100 m Freistilfinales. Der abschließende Wettbewerb ist das Tontaubenschießen. Zugleich die letzte Gelegenheit, noch Punkte im Kampf um die Goldmedaille wettzumachen. Insgesamt 25 Schuß sind auf die fliegenden Scheiben abzugeben, dann wird es Ernst. Es folgt die Aufrechnung der Ergebnisse aller Disziplinen mit Medallenspiegel. Dem verdienten Sieger erklingt die Nationalhymne, die Summer Games sind vorüber. Aber hier ist Kritik über die Leistung der Sportler fehl am Platze, dabei sein ist schließlich alles.

(sr)

Decathlon

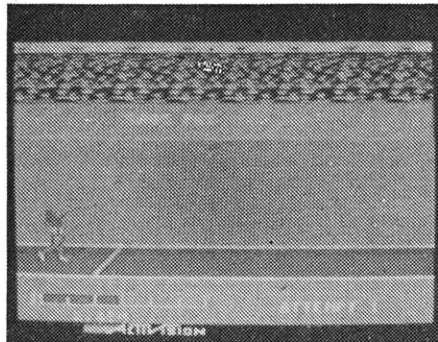


Man nennt sie die "Könige" der Athleten. In zehn verschiedenen Disziplinen wird der beste Allroundsportler ermittelt, wobei in jeder einzelnen Höchstleistungen vollbracht werden. Decathlon von Activision ist der Zehnkampf am Computer. Wie schon bei Summer Games wird die Wohnstube zur Sportarena.

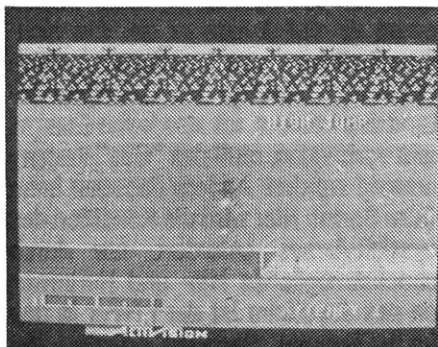
3.) Kugelstoßen

Bis zu 4 Teilnehmer können bei Decathlon an den Start gehen und den Kampf um die Punkte aufnehmen. Wieder einmal übernimmt der Joystick die Fortbewegung der Sportler. Empfehlenswert ist allerdings ein stabiler Joystick, denn durch alle Disziplinen wird dieser auf eine harte Probe gestellt. Die Zehnkämpfer sind nur durch blitzartiges Hin- und Herbewegen des Steuerknüppels zu bewegen, je schneller desto besser. Auch den Spielern werden Höchstleistungen abverlangt, Ausdauer ist Trumpf. Als erster Wettkampf steht der 100 m Sprint an, je 2 Teilnehmer laufen gegeneinander. Sollte sich kein Mitstreiter finden, übernimmt der Computer den gegnerischen Läufer. Die erreichte Zeit wird in Punkte umgerechnet und von Disziplin zu Disziplin aufaddiert. Grafische und akustische Sonderleistungen sind bei Decathlon nicht vorhanden, der Reiz liegt eindeutig in der Joystickarbeit. Interessant ebenfalls einige Disziplinen, die es bei vergleichbaren Programmen bisher noch nicht gab. So wird anschließend zum Weitsprung angetreten. Je mehr der Steuerknüppel bearbeitet wird, desto größer ist schließlich die erreichte Weite.

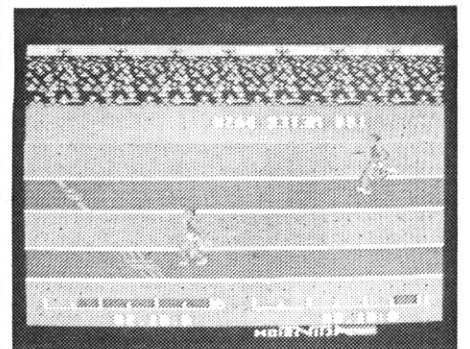
Die übrigen Disziplinen in ihrer Reihenfolge, die übrigens dem Original-Wettkampf entsprechen:



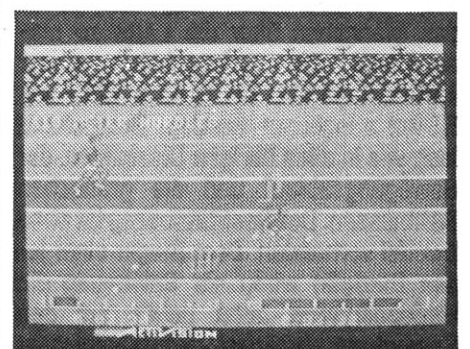
4.) Hochsprung



5.) 400m Lauf



6.) 110m Hürden



ACHTUNG!

**Bei uns finden Sie
fast alles für Ihren
VC-20/64**

Hardware:

Speichererweiterungen 8K-64K RAM, Moduladapter 2/3/5-fach, Recorder-interface, Schnell-Save Module, Toolkitmodule, 40/80 Zeichenkarten, Epromprogrammierer, Epromkarten, PIO IN/OUT Module, Interface, Joysticks, Paddle, und...und...und...

Zubehör:

Kassetten, Disketten, Staubschutzhauben, Bücher, Bausätze, Resettaster, Alle Stecker, Bauteile, und...und...

Software:

Außergewöhnliche Spiele und Programme, z.B. Flugsimulatoren, Datenprogramme, Krankheitsdiagnose, Biorhythmus, Lottoberechnung und Spiele Spiele Spiele

Prüfen Sie unser Angebot der Tiefstpreise.

„Schnell das neue Spitzeninfo anfordern, für 2 DM in Briefmarken. Garantiert 24 Std. Infoversand.“

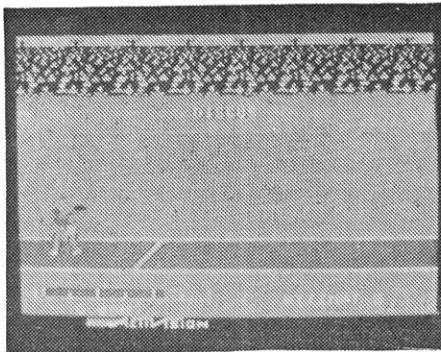
Händleranfragen erwünscht.

mükra
DATEN-TECHNIK

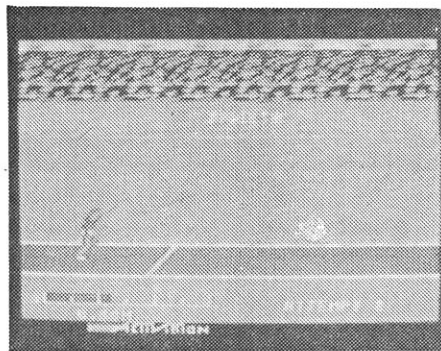
Rotdornweg 15
1000 Berlin 45
☎ 030-817 38 57
341 45 73

3 x duftige Spiele,
Tips und Tricks, News,
Berichte, Software-News,
in: Homecomputer, CPU
und Computerposter,
das Superposter mit dem
Spiel auf der Rückseite.
Jeden Monat neu bei
Ihrem Zeitschriftenhändler.

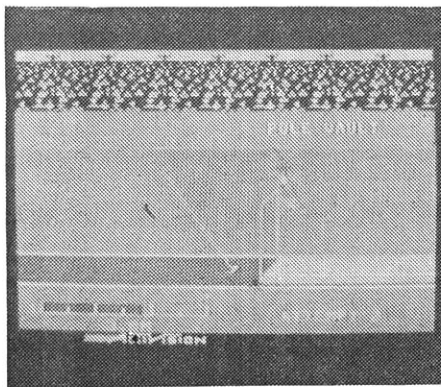
Roeske Verlag
Fuldaer Straße 6
3440 Eschwege



7.) Diskuswerfen



9.) Speerwerfen
10.) 1.500m Lauf



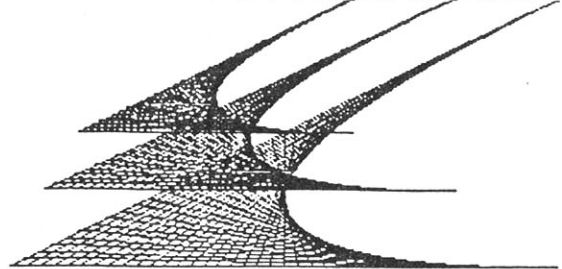
8.) Stabhochsprung

Die Wurf- und Sprungsdisziplinen sind dabei die einfachsten und bei weitem nicht so schweißtreibend wie die Laufübungen. Sie erfordern nur einen kurzen Anlauf und rechtzeitigen Druck auf die Feuertaste. Spätestens nach dem 400m Lauf benötigt der Joystickarm eine kleine Erholung, für Ungeübte ist die Gefahr eines Krampfes sehr hoch. Der wiederrum kostet im Wettstreit um den Sieg wert-



**GÖRLITZ
COMPUTERBAU**

Ihr Commodore 64 kann viel mehr



wenn Sie ihn mit unserem Görlitz-VC-EPSON-INTERFACE an einen Epson-Drucker anschließen.

Unser VC-EPSON-INTERFACE Best.-Nr. 8422 bringt alle CBM-Grafikzeichen in vierzig verschiedenen Schriftarten und -breiten * HARDCOPY mit SIMON's BASIC sauber und hochauflösend * hat einen eingebauten Selbsttest * einen 2K Pufferspeicher * einen eigenen Z80-Mikroprozessor * zwei IEC-Buchsen (wie Floppy) * IEC-Adresse einstellbar * keine Modifikation des Druckers * definierbarer Zeichensatz (FX) bleibt frei * ein Interface für C-64, VC-20 an alle Epson-Drucker * Lieferung komplett mit DIN-Verbindungskabel und ausführlicher Bedienungsanleitung.

EPSON RX-80	mit VC-Interface Best.-Nr. 8422 DM 1330, -
EPSON RX-90 F/T	mit VC-Interface Best.-Nr. 8422 DM 1502, -
EPSON FX-8J	mit VC-Interface Best.-Nr. 8422 DM 1889, -
EPSON FX-110	mit VC-Interface Best.-Nr. 8422 DM 2362, -

Preise inkl. MwSt., Katalog 1/84 kostenlos.

Besuchen Sie unsere Ausstellung in der Rhein-Mosel-Halle in Koblenz vom 29. Okt. bis 31. Okt. 84. Wir freuen uns auf Ihren Besuch!

GÖRLITZ COMPUTERBAU-Postf. 852 - 5400 Koblenz - Tel. 0261-27500

volle Punkte. Wie auch im "echten" Wettkampf ist der abschließende 1.500 m Lauf die von allen gefürchtetste Disziplin. Und das mit Recht. Wer sich bis jetzt noch nicht ausgetobt hat, der ist spätestens nach diesem Lauf völlig geschafft. Die 1.500 m scheinen endlos lang, die Kraft muß eingeteilt werden. Wer am Anfang zu schnell angeht, wird die Nase am Schluß nicht mehr vorn haben. Die heißeste Phase des Rennens sind die letzten 200m, die im Sprinttempo zurückgelegt werden. Diese Disziplin ist zugleich auch die härteste Bewährungsprobe für den Joystick. Nicht zuletzt wegen der aktiven Teilnahme der Spieler kann dieses Programm als gut gelungen gelten, schon das Zuschauen macht Spaß. Zu einem erneuten Durchgang gehört dann allerdings schon eine Bärenkondition und vielleicht sogar ein neuer Joystick. Etwas unübersichtlich schien uns die aktuelle Ergebnisanzeige. Nur die erreichten Punkte werden nach Abschluß jeder Disziplin angezeigt, die Zwischen- und Bestergebnisse muß man sich entweder notieren oder merken. Das Programm wird in einem Stück in den Rechner geladen, es entstehen somit keine Wartezeiten zwischen den einzelnen Disziplinen. Durchgehender Spaß ist gewährleistet!!

Arbeitsgemeinschaft Software-Anwender

In den vergangenen Monaten ist ein Thema immer wieder in den verschiedenen Fachzeitschriften aufgetaucht, daß zur Zeit auch im privaten Bereich sehr viel Aufmerksamkeit erregt.

Es handelt sich dabei um die Frage, was der Einzelne mit den Programmen für seinen Computer anfangen kann bzw. ob diese Software getauscht, verkauft oder kopiert werden darf. Die Rechtsunsicherheit ist inzwischen so groß geworden, daß niemand eine konkrete Stellungnahme abgeben kann. Händler, die sich durch sog. "Raubkopierer" geschädigt fühlen, beantragen wahllos Strafanzeigen und treffen damit fast immer die Falschen.

Bei den Betroffenen, die eventuell sogar Besuch von der Kriminalpolizei erhalten, ist in vielen Fällen die Panikstimmung so groß, daß sie sogar mit dem Gedanken spielen, ihr Hobby aufzugeben und den Computer zu verkaufen. Daß sich die Händler damit selbst schädigen, ist offensichtlich. Auch sind bereits Fälle vorgekommen, bei denen eindeutig eine betrügerische Absicht zugrunde gelegen hat. Erst vor einigen Wochen hat eine Pseudo-Firma aus Berlin unter Ausnutzung der allgemeinen Unsicherheit an Kleinanzeigen-Inserenten Abmahnungen verschickt, um "auf die Schnelle" Geld zu verdienen. Durch solche Machenschaften erhält die Com-

puterei einen schlechten Ruf, der im Endeffekt alle Betroffenen schädigt.

Da der einzelne Computerfreund (es handelt sich ja in der Regel um Jugendliche) kaum in der Lage ist, hier Klarheit zu schaffen, soll die ARBEITSGEMEINSCHAFT SOFTWARE-ANWENDER gegründet werden. Ziel dieser AG soll sein:

1. Klärung der Rechtslage durch Kontakte mit Anwälten, Händlern und auch Zeit-schriften etc.
2. Information, Beratung und Aufklärung der Mitglieder und der Öffentlichkeit.
3. Erfahrungs- und Gedankenaustausch zu diesem Thema, daß alle betrifft.

Die ersten Verbindungen sind

bereits geknüpft. Durch intensiven Arbeitseinsatz soll schon in kürzester Zeit eine solide Grundlage für die weitere Tätigkeit der AG geschaffen werden.

Wer sich für dieses Produkt interessiert oder der Arbeitsgemeinschaft beitreten will, kann sich an

ROLF KNORRE
POSTFACH 200102
5600 WUPPERTAL

wenden. Jeder Interessent erhält dort weitere Informationen (bitte einen frankierten Rückumschlag beilegen). Die Arbeitsgemeinschaft ist für jeden wichtig, der Software kauft, verkauft, kopiert und tauscht etc., unabhängig vom Computertyp.

Computer im Fernsehen

Liebe Freunde von COMPUTE MIT...!

Alles, was sich im Fernsehen mit dem unerschöpflichen Thema Homecomputer beschäftigt, nennen wir Ihnen jeweils an dieser Stelle, damit Sie genau zur rech-

ten Zeit Ihren Fernseher einschalten können und nichts wesentliches versäumen. Ein toller COMPUTE MIT...-Service – denn COMPUTE MIT...-Leser wissen Bescheid und können mitreden.

24.09.84	ZDF
Mikroprozessoren (1)	16.04 h
01.10.84	ZDF
Mikroprozessoren (2)	16.04 h

21.09.84	SW3
Mikro-Prozessoren-Computer	21.15 h

Was sind Mikroprozessoren, wo werden sie eingesetzt?
1. Von der Dampfmaschine zum Chip

28.09.84	SW3
Mikro-Prozessoren-Computer	21.15 h
2. Computer in der Produktion	

WERSIBOARD FÜR COMMODORE C 64/SX 64

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 (067 47) 71 31 · Telex 42323

BESTELL-COUPON abtrennen und senden an WERSI electronic, Industriestraße, 5401 Halsenbach, Telefon 067 47/71 31
Hiermit bestelle ich/wir zu Ihrem Allgemeinen Verkaufs-, Lieferungs-, Zahlungs- und Servicebedingungen zur sofortigen Lieferung:
☐ Stück WERSIBOARD MUSIC 64 Einzelpreis DM 495,-
☐ Personal Computer Music zuzüglich Versandkosten
Zahlung erfolgt: ☐ per Nachnahme ☐ per beigefügtem Scheck ☐ per Vorauskasse
☐ Übersenden Sie mir vor Lieferung die Rechnung ☐ Überweisungsbeleg ist beigefügt

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

MUSIC 64

PERSONAL
COMPUTER
MUSIC

EINFÜHRUNGSPREIS

DM **495,-**

inkl. MWSt./zuzgl.
Versandselbstkosten



Noch heute bestellen!

Teil 3

Logik im Basic-Programm

Wie man Systematik in sein Selbstgestricktes bringt

So, da wollen wir uns mal die Hausaufgabe vom letzten Mal vornehmen: Haben Sie an einer Verbesserung unseres "Rechnung-Schreib-Programmes" gearbeitet? Prima! Dann vergleichen wir am besten einmal.

Ich habe bei meiner Überarbeitung versucht, schon so etwas wie Systematik ins Konzept zu kriegen. Und um mein Vorgehen optisch zu verdeutlichen, habe ich die funktions- und sinnbedingt zusammengehörenden Zeilen zu Gruppen verbunden. Das ermöglicht es mir nun, den Programmaufbau zu erläutern:

Zeile 10:

Zunächst wird durch CHR\$(147) der Bildschirm freigemacht. Das erhöht die Übersichtlichkeit, bringt also ein wenig Komfort ins Spiel. Danach lasse ich den Rechner nach der Anzahl der in Rechnung zu stellenden Artikel fragen: Diese Angabe ist für die Reservierung des Speicherplatzes wichtig. Anhand der Variablen A wird die Matrix X(B,1) DIMensioniert, die die Angaben bezüglich Warenart und Einzelpreisen aufnehmen soll. Für die nächsten Eingaben löschen wir erneut das Display.

Zeile 20:

Variable K für Kundennummer wird angefordert: Mit dem erhaltenen Wert sucht der Computer aus unserer Kunden-Kartei den jeweiligen Rechnungsempfänger heraus. Auch nach dieser Eingabe wird der Bildschirm gelöscht.

Zeile 30-33:

Hier liegt die erste Schleife unseres Rechnungsprogrammes. Bis zur angegebenen Zahl der berechneten Warenposten fragt der Rechner nach

Artikelnummer und Einzelpreis. Zeile 32 leistet dabei erste eigentliche "Rechenarbeit", indem Gesamtwarenwert (G), Mehrwertsteuer (M), Endbetrag (E) und Skonto (S) ermittelt werden. Dabei habe ich die Schreibweise $M = G * 14 / 100$ bzw. $S = * 3 / 100$ gekürzt.

Zeile 40-44:

schreiben den Rechnungskopf mit Adresse des durch K ermittelten Kunden unseres Computer-Ladens.

Zeile 50-59:

Unsere zweite Schleife dient der Umsetzung der Artikelnummern in Klartext, damit der Kunde sofort weiß, für welche Ware er zahlen soll. Dabei steht in Zeile 52 eine recht interessante Anweisung: der Sprung (GOTO) erfolgt nämlich nicht nach einer festen Adresse, sondern hängt direkt von der Warennummer ab. Die Zeilen 54 bis 57 definieren die Stringvariable N\$, die für den ausgedruckten Text verantwortlich ist. Dabei sind in den Zeilen 54,55 und 56 die GOTO's wichtig, um nach Wahl des Textes sofort für den Ausdruck zu sorgen. Nehmen wir an, in Zeile 54

stände lediglich N\$="VC20", so ginge der Rechner zur nächsten Zeile weiter, würde 'VC20' durch 'C-64', 'C-64' wiederum durch '1541' und letztlich '1541' durch '****' ersetzen.

In 57 darf das GOTO wegfallen, weil nun (wie jeder einsehen dürfte) nichts mehr schiefgehen kann.

Zeile 58 gibt Warenbezeichnung und jeweiligen Einzelpreis in DM aus.

Zeile 60-68:

Hier erfährt der Kunde, was er zu bezahlen hat: Ich nenne ihm zunächst den Gesamtwarenwert (also die Summe aller Einzelpreise), dann den Endbetrag (schließlich will ich die Mehrwertsteuer nicht selbst tragen) sowie als Anreiz zum schnellen Überweisen die Angabe des Barzahlungsrabattes (Skonto 3%).

Zeile 70-71:

liefert die Kontoverbindung, die ich gegenüber der langatmigen Erstfassung ein wenig gestrafft habe. Kontonummer und Bankleitzahl müssen genügen.

Zeile 80:

Die END-Anweisung wird dann nötig, wenn sich entwe-

der weitere Programme im Arbeitsspeicher des Computers befinden oder aber Unterprogramme folgen, in die nicht wahllos hineingefahren werden soll. In unserem Fall würde ohne END die Kundenkartei zumindest teilweise ausgedruckt, was nicht unbedingt erwünscht ist.

Zeile 100-138:

beinhaltet die gerade erwähnte Kartei. Unsere Kunden Möglich, Abernicht und Gewollt (die Familiennamen beziehen sich, wie mancher gemerkt haben mag, auf eventuelle Ähnlichkeiten mit lebenden Personen) sind Ihnen aus der letzten Folge sicher noch ein Begriff. Zeile 100 nutzt wieder die Möglichkeit des "bedingten Sprunges", wie er bereits in Zeile 52 behandelt wurde.

Zeile 140:

ist ein Miniatur-Unterprogramm zum Unterstreichen. Es wird zweimal (Zeile 44 und 60) angesprungen.

Wie Sie sehen, wird die Struktur des relativ kurzen und einfachen Programmes besonders deutlich durch die o.g. Funktionsgruppen sowie die strikt ans Programm-

Ende verlegten Subroutinen.

```
10 PRINT CHR$(147):INPUT "ANZ ARTIKEL "; A:B=A-1:DIM x(B,1):PRINT CHR$(147)
```

```
20 INPUT "KDN-NR ";K: PRINT CHR$(147)
```

```
30 FOR I=1 TO A
```

```
31 B=I-1: PRINT I;" ARTIKEL": INPUT X(B,0): PRINT "E-PREIS":INPUT X(B,1)
```

```
32 G=G+X(B,1):M=G*.14:E=G+M:S=E*.03
```

```
33 PRINT CHR$(147): NEXT I
```

```
40 PRINT "RECHNUNG FUER"
```

```
41 PRINT " "
```

```
42 PRINT "HERRN/FRAU/FRL": GOSUB 100
```

```
43 PRINT "ART","E-PREIS/DM
```

```
44 GOSUB 140
```

```
50 FOR I = 1 TO A
```

```
51 B= I-1
```

```
52 ON X(B,0) GOTO 54,55,56
```

```
53 GOTO 57
```

```
54 NS="VC20":GOTO 58
```

```
55 NS="C-64":GOTO 58
```

```
56 NS="1541":GOTO 58
```

```
57 NS="*****"
```

```
58 PRINT NS,X(B,1)
```

```
59 NEXT I
```

```
60 PRINT " ":GOSUB 140
```

```
61 PRINT "GES.-WW",G
```

```
62 PRINT " "
```

```
63 PRINT "RECH.-BETRAG"
```

```
64 PRINT "INKL 14% MWST".E
```

```
65 PRINT " "
```

```
66 PRINT "B.ZAHL INNERHALB 8 TAGEN"
```

```
67 PRINT "SKONTO 3% (=","S;")"
```

```
68 PRINT " "
```

```
70 PRINT "UEBERWEISUNG AUF PGIROKTO"
```

```
71 PRINT "NR 4289 31-104 (BLZ 100 100 10)
```

```
80 END
```

```
100 ON K GOTO 110,120,130
```

```
101 PRINT"NICHT IN DER DATEI"
```

```
102 GOTO 138
```

```
110 PRINT"FRITZ MOEGlich"
```

```
111 PRINT"TANNENWEG 2"
```

```
112 PRINT"9191 DINGSDA 1"
```

```
113 GOTO 138
```

```
120 PRINT"SUSANNE ABERNICHT"
```

```
121 PRINT"POSTFACH"
```

```
122 PRINT"9100 GROSSTADT 50"
```

```
123 GOTO 138
```

```
130 PRINT"BRIGITTE GEWOLLT"
```

```
131 PRINT"MARKTSTRASSE 17"
```

```
132 PRINT"9567 DADORF"
```

```
138 PRINT" ":RETURN
```

```
140 PRINT"- - - -","- - - - - - - -":RETURN
```

Das wär's für heute: In der kommenden Woche werden wir uns mit ein paar neuen Befehlen und dem Kapitel "Entscheidungen" unterhalten. Tschüß bis dann ...

(sk)

MECKERN ist gesund

Ihnen hilft's. Frust am Computern
erst gar nicht aufkommen zu
lassen – uns ist es nur recht,
weil wir auf diese Art und Weise
eine "Compute mit" machen
können,

die für Sie das Optimale
aus Ihrem Hobby heraus
und in Ihren Commodore
herein holt ...

Schreiben Sie also regelmäßig an Ihre
CM - Redaktion !

Frisch nach der Goldenen Regel :

*Wenn Sie mit uns zufrieden sind,
sagen Sie es weiter,
sind Sie's nicht,
sagen Sie es raus !*



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 problemlose 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

Der Sensible C-64	4. Q. 84, DM 29,80
CP/M und WordStar	DM 29,80
C-64 Programmsammlung	4. Q. 84, DM 29,80
VisiCalc (mit CBM Diskette)	DM 79,—
77 BASIC Programme	DM 39,—
Mikrocomputer-Grundwissen	DM 36,—
C-64 Computer für Kinder	4. Q. 84, DM 29,80
VC20 Computer für Kinder	4. Q. 84, DM 29,80

Coupon

Hiermit bestelle(n) ich (wir):

Datum

Unterschrift

Name/Firma

Straße/Postfach

Ort

Assemblerkurs

Teil 3

Wie Sie sicher festgestellt haben, ist im 2. Assemblerkurs ein Teil der Schere zum Opfer gefallen – was wir sehr bedauern und fahren gleich mit dem 3. Teil fort.

All diese Bestandteile sind im Computer miteinander verbunden. Diese Verbindung nennt man Busse. Von diesem gibt es mehrere, die unterschiedliche Aufgaben erfüllen:

- 1) zur Datenübertragung – der Datenbus
- 2) zur Adressierung der einzelnen Bauteile – der Adressbus
- 3) der Steuerbus, über den die CPU die anderen Einheiten steuert.

Das Herzstück des Computers ist die CPU (Central Processing Unit), die Zentraleinheit. Zur Erfüllung ihrer Aufgaben verfügt sie über unterschiedliche Funktionseinheiten (Register), die zur Adressierung, zur Steuerung und zur Rechnerei nötig sind. Der 65XX Prozessor besitzt folgende Register:

PC -Befehlszähler (Program Counter)

S -Stapelzeiger (Stack Pointer)

A -Rechenregister (Akkumulator)

X,Y -Indexregister zur Tabellenverwaltung bzw. indizierten Adressierung

P -Statusregister, welches die Ergebnisse der einzelnen Operationen anzeigt.

Die Verwendung dieser Register werden wir in den folgenden "Compute mit"-Ausgaben anhand von Beispielen kennenlernen.

Bevor wir mit unserem Kurs fortfahren, möchten wir Ihnen

2 Programme vorstellen, die für das Arbeiten in Assembler unentbehrlich sind.

Man beginnt mit dem Einladen des Editors. Nach dem Starten erhält man eine Tabelle mit den Befehlen, die der Editor kennt und den möglichen Parametern.

Es genügt, wenn die inversen Buchstaben eingegeben werden und auch Blanks sind optional. Man kann aber auch den gesamten Befehl eingeben oder Teile davon, sofern die signifikanten Buchstaben am Anfang enthalten sind.

ADD

Um ein Assembler-Programm einzugeben, benutzt man den Befehl A(DD) (Anm.: die Buchstaben in Klammern sind optional).

Dann erscheint eine "1" mit einem inversen Blank dahinter. Dies ist der Cursor, während die "1" die laufende Zeilennummer des Befehls ist. Wenn man einmal einen Befehl unter einer bestimmten Zeilennummer eingegeben hat, so heißt dies nicht, daß zu dieser Zeilennummer immer der eingegebene Befehl gehört, denn man kann auch noch nachträglich Befehle einfügen, und dabei ändern sich dann entsprechend die Nummern der folgenden Befehle.

ADD heißt immer Anhängen von weiteren Befehlen. Wenn der Speicher leer ist, dann lautet die "nächste" Zeile 1, wenn bereits 43 Zeilen vorhanden sind, dann ist 44 die Zeile, in die nach ADD geschrieben wird.

INSERT

Hat man irgendwo im Assembler-Listing einen oder

mehrere Befehle vergessen nachzutragen, so benutzt man den Befehl I(INSERT). Und zwar gibt man nach dem Befehlswort die Zeilennummer an, hinter der die Befehle eingeschoben werden sollen. Fehlen Befehle direkt am Anfang, so wird nur der Befehl ohne Parameter eingegeben und die Zeilen werden solange eingeschoben, bis eine neue Editoranweisung dieses beendet.

DELETE

Sind Assembler-Zeilen überflüssig geworden, so kann man sie mit D(DELETE) entfernen. Wie bei dem BASIC-Befehl LIST kann man hinter dem Delete-Befehl entweder nur eine Zeilennummer eingeben, wenn man nur eine Zeile löschen will; oder z.B. "D-5" eingeben, wenn man die ersten 5 Zeilen löschen möchte; oder "D5-" wenn alle Zeilen ab Nummer 5 gelöscht werden sollen; oder einfach "D", um sämtliche Zeilen zu löschen.

Damit man nicht aus Versehen Zeilen entfernt, die man noch benötigt, wird zur Sicherheit noch einmal gefragt, ob man auch sicher ist. Die Frage (in Englisch) taucht immer dann auf, wenn mehr als 10 Zeilen gelöscht werden sollen.

LIST

Auch bei L(IST) ist die Parameterübergabe wie im BASIC. Wird beim Listen die Taste "S" gedrückt, so stoppt der Editor und erwartet zur Fortsetzung einen weiteren Tastendruck. Drücken der Taste "Q" beendet das Listen bevor die letzte Zeilennummer erreicht ist.

EDIT

Stellt man fest, daß eine Zeile falsch eingegeben wurde, so kann sie mit dem E(DIT)-Befehl korrigiert werden. Die Parameterübergabe ist wie unter DELETE beschrieben. Der Editor listet die zu korrigierende Zeile und setzt den Cursor an den Zeilenanfang. Diesmal ist es ein "echter Cursor" und es zählt, wie von CBM gewohnt, alles, was in der oder den entsprechenden Bildschirmzeilen steht. Deshalb kann man nach Durchführen der Korrektur sofort "RETURN" drücken, und auch der Rest nach dem Cursor wird übernommen. Oder nur "RETURN", wenn in der Zeile nichts geändert werden soll. Auch bei dem EDIT-Befehl kann man einfach eine neue Editor-Anweisung eingeben, wenn man aufhören will, sofern man den Rest der Zeile (oder Zeilen) mit Spaces löscht.

COPY

Wenn man einen Block, den man schon eingegeben hat, an einer anderen Stelle haben möchte, so verwendet man den Befehl C(OPY). Direkt hinter dem Befehl gibt man aber noch keine Parameter an. Der Editor fragt diese nämlich der Reihe nach ab. Er beginnt mit dem Bereich der zu kopierenden Zeilen. Die Zeilennummern werden wie bei DELETE, LIST oder EDIT behandelt.

Danach erfolgt nach Anweisung die Eingabe der Zeilennummer, hinter der die Zeilen eingeschoben werden sollen (fast wie bei INSERT).

Man kann den Block nämlich nicht nur anhängen, sondern an jede Stelle innerhalb des Assemblerlistings einfügen,

sogar innerhalb des Blocks selbst. Lediglich wenn man den Block direkt an den Anfang kopieren möchte muß man sich eines kleinen Tricks bedienen. Man fügt dann nämlich einfach eine neue Zeile ein, kopiert den Block (Vorsicht! die zu kopierenden Zeilennummern haben sich jetzt um 1 verschoben!) hinter die Zeile 1 und löscht dieselbe mit D1. Der zu kopierende Block steht jetzt am Anfang.

APPEND SAVE VERIFY

Drei Befehle dienen dem Laden, Retten und Verifizieren von Assembler-Files auf Diskette oder Kassette.

Der Befehl LOAD, den man in diesem Zusammenhang erwartet gibt es nicht. Ihn ersetzt der Befehl AP(PEND), der einen Parameter haben kann; die Zeilennummer, hinter der das File vom Datenträger in das im RAM befindliche Programm geladen werden soll.

Ist der Speicher leer, so wird sich nach APPEND nur das Programm, das zugeladen wurde, im Speicher befinden. APPEND ohne Parameter hängt das Programm an das im Speicher befindliche an!

Um das Assembler-Listing auf Kassette oder Diskette aufzunehmen, wird der Befehl S(AVE) benutzt. Ohne Parameter speichert man das gesamte, im Speicher befindliche Programm; und mit entsprechenden Parametern (wie oben unter DELETE beschrieben) können Teile des Programms übertragen werden.

Um sich zu vergewissern, daß alles richtig gespeichert wurde, kann mit V(ERIFY) die Korrektheit der Aufnahme überprüft werden. Dabei ist zu beachten, daß wenn Parameter übergeben wurden, diese identisch sind mit denen, die beim Kommando SAVE verwendet wurden.

GOTO 280

Hat man aus irgend einem Grunde Stop oder Break ge-

drückt, so kann mit GOTO 280 der Programmablauf fortgesetzt werden, ohne daß das Assembler-Listing gelöscht wird (was ja nach RUN der Fall wäre).

QUIT

QU(IT) sorgt als Befehlseingabe dafür, daß Assembler und Disassembler ausgeführt werden.

Achtung! Zuerst muß man das Assembler-Programm im Speicher auf einen Datenträger (Kassette oder Diskette) aufgenommen werden! Erst dann kann das Programm assembliert werden.

Ist der Assembler nachgeladen, dann werden wie vom Bildschirm her befohlen, die Kassette zurückgespult und die weiteren Befehle befolgt. Dies entfällt bei Diskettenbetrieb, wo lediglich darauf zu achten ist, daß sich Assembler und Assembler-Programm auf der selben Diskette befinden.

SYNTAX und Features

Gibt man direkt hinter einer Zeilennummer einen Editorbefehl ein, so wird dieser ausgeführt – und nicht in die Zeile übernommen!

Alles andere, was direkt hinter einer Zeilennummer steht und nach einem Blank mit weiteren Eingaben versehen ist, wird in die entsprechende Zeile geschrieben.

Labels werden direkt hinter die Zeilennummer geschrieben und dürfen keine Blanks enthalten. Ein Blank, das einem Label folgt, bedeutet dem Editor, daß nun ein Mnemonic-Befehl folgt. (Mnemonics, soviel sei für Anfänger gesagt, nennt man die Codewörter wie z.B. LDA (Load Akku), STA (Store Akku), TAX (Transfer Akku to X-Register), usw.

Folgt direkt hinter der Zeilennummer ein Blank, dann gehört hinter dieses Blank ein Mnemonic.

Dahinter gehört, wenn es sich um einen Mehrbytebefehl handelt, wieder ein Blank und dann ein Operand.

Ein Semikolon wird Kommentaren, die zur besseren Verständlichkeit des Programms einer Programmzeile beigelegt werden können, vorangestellt.

Der Assembler versteht Hex- und Dezimalzahlen. Hexzahlen haben ein Dollarzeichen vorangestellt. Direkte und relative Sprünge zu Labels und zu labelabhängigen Punkten sind möglich. z.B.:

JSR LABEL

oder

JSR LABEL+86

BNE MARKES8-9

Auch Sprünge relativ zum Programm-Counter sind gestattet. Das Sternchen "*" ist in diesem Fall das Symbol für den Programm-Counter. Beispiel: BNE *-3. Die bisher zusätzlich vom Assembler verstandenen Mnemonics sind:

ORG Programmbeginn

EQU (Equate)

DFB (Define Byte)

ORG hat kein Label und muß

eine Hex- oder Dezimalzahl im Operandenfeld haben.

Vor EQU muß sich ein Label befinden und dahinter eine entsprechende Hex-, Dual- oder Dezimalzahl.

Vor DFB kann ein Label stehen. Dem Mnemonic folgen ein oder mehrere Hex-, Dual- oder Dezimalzahlen. Mehrere Zahlen werden durch Kommata getrennt.

Eine weitere nachträglich eingeführte Bequemlichkeit ist, daß mit den Größer-als- und Kleiner-als-Zeichen Labels in High und Low-Byte zerlegt werden können.

Z.B.

1ADRESSE EQU 34356

2 LDA #< ADRESSE

3 STA HIGHBYTE

4 LDA #> ADRESSE

5 STA LOWBYTE

Wie bereits angesprochen, werden weitere "Bequemlichkeiten", nützliche Befehle und Hilfsprogramme zum

NEU FÜR DEN COMMODORE C-64



**Die revolutionäre,
neue Lernmethode.**

SM SOFTWARE AG / MÜNCHEN

Programmieren in Assembler
(und somit in Maschinencode) folgen.

Editor



```

80 POKE45,32:POKE46,31
90 CLR
100 DIMA$(1000)
110 L$=" ":FOR I=1 TO 5:L$=LEFT$(L$+L$,21):NEXT:L$=L$+" "
120 PRINTTAB(6);"HOME COMPUTER
200 PRINT"HOME COMPUTER EDITOR
210 PRINT"DISAPPEND [A]
215 PRINT"DISSAVE [A] [-] [E]
220 PRINT"DISVERIFY [A] [-] [E]
225 PRINT"DISADD
230 PRINT"DISINSERT [A]
235 PRINT"DISEDIT [A] [-] [E]
240 PRINT"DISDELETE [A] [-] [E]
245 PRINT"DISLIST [A] [-] [E]
250 PRINT"DISCOPY
255 PRINT"DISQUIT -> ASS + DISASS
280 IFB=10RB=50RB=9THENA$="L":GOTO300
290 INPUT"BEFEHL":A$
300 PRINT:RESTORE:FORB=1 TO 10:READB$:IFLEFT$(A$,LEN(B$))<>B$THENNEXT
310 IFB<>2ANDB<9THENGOSUB5000
330 ONB GOTO400,600,500,700,800,1000,1200,900,1100,540,200
400 REM APPEND
410 GOSUB6000
420 INPUT#1,A$
430 IFN2<NTHENFORC=NTON2+1STEP-1:A$(C+1)=A$(C):NEXT
440 N2=N2+1:N=N+1:A$(N2)=A$:IF(64ANDST)=0THEN420
450 CLOSE1:GOTO280
500 REM SAVE
502 INPUT"FILENAME":F$:PRINT"CASSETTE / DISK 2C/2D":WAIT198,1:GETG$:IFG$="C"TH
EN510
505 IFG$<>"D"THEN502
507 OPEN1,8,2,F$+".S,W":GOTO520
510 OPEN1,1,1,F$
520 FORC=N1TON2:PRINT#1,CHR$(34)A$(C):NEXT:CLOSE1
540 PRINT"ASSEMBLER EINLADEN ?
550 WAIT198,3:GETG$
560 IFG$="J"THENCLEAR:LOAD"ASSEMBLER",8
570 PRINT"DISASSEMBLER ?
580 WAIT198,3:GETG$
590 IFG$="J"THENCLEAR:LOAD"DISASSEMBLER",8
595 GOTO280
600 REM ADD
610 C=N:PRINT"
620 GOSUB4000:IFBFTHEN300,XPRINTL$:GOSUB4500
640 A$(C)=A$:N=C:GOTO620
700 REM LIST
710 PRINT" ":FORC=N1TON2:A$=A$(C):GOSUB4500
720 GETG$:IFG$="S"THENWAIT198,1:GETG$
730 IFG$="Q"THEN280
740 NEXT:GOTO280
800 REM DELETE
810 D=N2-N1+1:IFD>10THENPRINT"ARE YOU SURE ? ":WAIT198,7:GETG$:IFG$="N"THEN28
0
820 N=N-D
830 FORC=N1TON:A$(C)=A$(C+D):NEXT:GOTO280
900 REM VERIFY
910 GOSUB6000
920 FORC=N1TON2:INPUT#1,A$
930 IFA$(C)A$(C)OR64ANDSTANDC<N2THENPRINT"?VERIFY ERROR IN"C:CLOSE1:GOTO280
940 NEXT:CLOSE1:PRINT"OK":GOTO280
1000 REM EDIT
1010 FORC=N1TON2:PRINT" ":S=PEEK(214):A$=A$(C):GOSUB4500
1015 PRINT" ":IFPEEK(214)>STHEN1015
1020 A$="":SYS50528

```

```

1030 FOR I=515 TO 600: IF PEEK(I) THEN A$=A$+CHR$(PEEK(I)):NEXT
1040 GOSUB 4100: IF B THEN 300
1050 A$(C)=A$:NEXT C:GOTO 280
1100 REM COPY
1110 INPUT "COPY-BEREICH";A$
1120 GOSUB 5000: INPUT "WOHIN (ZEILENNUMMER)";C
1130 IF C > N THEN C=N
1140 D=N2-N1+1
1150 IF C < N THEN FOR I=NTOC+1 STEP -1: A$(I+D)=A$(I):NEXT
1160 N=N+D: IF N1 > C THEN N1=N1+D
1170 FOR I=C+1 TO C+D: A$(I)=A$(N1): IF N1=C THEN N1=N1+D
1180 N1=N1+1:NEXT
1190 GOTO 280
1200 REM INSERT
1210 C=N1
1220 GOSUB 4000: IF B THEN 300
1230 PRINT L$:GOSUB 4500
1240 FOR I=NTOC STEP -1: A$(I+1)=A$(I):NEXT: N=N+1: A$(C)=A$
1250 GOTO 1220
4000 A$="": I=1: C=C+1: C$=RIGHT$( " "+STR$(C),3)
4005 PRINT "00000"
4010 S=PEEK(214): PRINT C$ LEFT$(A$,I-1) " " LEFT$(MID$(A$,I,1)+" ",1) " " MID$(A$,I+1)
" "
4015 IF PEEK(214) > 0 THEN PRINT "0": GOTO 4015
4020 WAIT 198,15: GETG$: J=ASC(G$)
4030 IF J=23 THEN 4090
4040 IF J=157 THEN I=I+(I>1):GOTO 4010
4050 IF J=148 THEN A$=LEFT$(A$,I-1)+" "+MID$(A$,I):GOTO 4010
4060 IF J=20 THEN A$=LEFT$(A$,I-1+(I>1))+MID$(A$,I): I=I+(I>1):GOTO 4010
4070 IF J=13 THEN 4100
4080 A$=LEFT$(A$,I-1)+G$+MID$(A$,I+1)
4090 I=I+1+(I>LEN(A$)):GOTO 4010
4100 FOR I=1 TO LEN(A$): IF MID$(A$,I,1) <> " " THEN NEXT: BF=-1: RETURN
4110 IF MID$(A$,I+1,1)="" THEN A$=LEFT$(A$,I)+MID$(A$,I+2):GOTO 4110
4120 BF=0: RETURN
4500 PRINT RIGHT$( " "+STR$(C),3);
4510 FOR I=1 TO 8: I$=MID$(A$,I,1): PRINT I$: IF I$ <> " " THEN NEXT: I=I-1
4520 PRINT TAB(10) MID$(A$,I+1) " ": RETURN
5000 D=0: FOR I=1 TO 9: J=ASC(MID$(A$,I,1)+" "): IF J=45 THEN D=1
5010 IF J<48 OR J>57 THEN NEXT
5020 N1=VAL(MID$(A$,I))
5030 N2=VAL(MID$(A$,I+LEN(STR$(N1))))
5040 IF N1=0 THEN N1=1 AND B <> 7: N2=N
5050 IF 0 THEN N2=N1: N1=1
5060 IF RIGHT$(A$,1)="-" THEN N2=N
5070 IF N2<N1 THEN N2=N1
5080 IF N2 > N THEN N2=N
5090 IF N1 > N THEN N1=N
5100 RETURN
6000 INPUT "FILENAME";F$: PRINT "CASSETTE / DISK 00/00": WAIT 198,1: GETG$: IF G$="D" THEN 6100
6002 IF G$ <> "C" THEN 6000
6005 PRINT "CASSETTE ZURUECKSPULEN"
6010 PRINT "BIS ZUM ANFANG DES": PRINT "ASSEMBLER-FILES!"
6020 PRINT "DANN P DRUECKEN!"
6030 WAIT 198,1: GETG$: PRINT "OK": OPEN 1,1,0,F$: RETURN
6100 OPEN 1,8,2,F$+"$,R": RETURN
10000 DATA A,P,S,L,D,E,I,V,C,Q
READY.

```

Assembler

```

50 PRINT "HOME COMPUTER-ASSEMBLER"
100 READ O$: READ A$: O$=O$+A$: READ A$: O$=O$+A$: DIM A$(500),A(500)
110 FOR I=0 TO 7: READ A$(I): NEXT
120 DIM O$(58): FOR I=0 TO 55: READ O$(I): NEXT
130 FOR I=0 TO 7: READ S$(I): NEXT
140 DIM A$(21): FOR I=0 TO 21: READ A$(I): NEXT
200 PRINT "CASSETTE / DISK 00/00": WAIT 198,1: GETG$: IF G$ < "C" OR G$ > "D" THEN 200
210 DI=G$="D": IF NOT DI THEN GOSUB 5000
220 INPUT "FILENAME";F$: IF DI THEN F$=F$+"$,S,R"
500 OPEN 1,1+(7 AND DI),2 AND DI,F$: Z=0
510 Z=Z+1: INPUT #1,L$: FOR I=1 TO 9: IF MID$(L$,I,1) <> " " THEN NEXT

```



```

520 IF I-1 THEN A$(LC)=LEFT$(L$, I-1):A(LC)=PC:LC=LC+1
525 PRINTRIGHT$( " " +STR$(PC),5)LEFT$(L$, I)TAB(10)MID$(L$, I+1)
530 GOSUB 4000
540 PC=PC+L
900 IF (64 AND ST)=0 THEN 510
910 CLOSE 1
1000 IF NOT D1 THEN GOSUB 50000
1010 OPEN 1,1+(7 AND D1),2 AND D1,F#:F=-1:Z=0
1020 Z=Z+1:INPUT#1,L$:FOR I=1 TO 9:IF MID$(L$, I,1)<>" " THEN NEXT
1030 GOSUB 4000:IF OI>55 THEN 1150
1040 O=O%(OI):IF OI>21 THEN 1070
1050 IFAI=1 AND O AND 10 RL=3 THEN O=O+8
1060 O=O+S%(AI)
1070 IFAI<6 AND L>1 THEN GOSUB 5000:IF OI>46 AND OI<55 THEN S=S-PC-2:S=S+(256 AND S<0)
1080 S%=S/256:S=S-256*S%:PRINT PCTAB(7)OTAB(12)STAB(17)S%
1090 POKE PC,0:POKE PC+1,S:IF L=3 THEN POKE PC+2,S%
1100 PC=PC+L:IF (64 AND ST)=0 THEN 1020
1110 CLOSE 1
1120 GOTO 20000
1150 IF OI=57 THEN 1100
1160 K=0:FOR P=PCTOP+L-1:A$="":FOR K=K+1 TO LEN(L$):J$=MID$(L$,K,1)
1165 IF J$<>"", THEN A$=A$+J$:NEXT
1170 GOSUB 5000:POKE P,S:NEXT P:GOTO 1100
4000 A$=MID$(L$, I+1,3):L=0
4010 FOR J=1 TO 175 STEP 3:IFA$(J)>MID$(O$,J,3) THEN NEXT:PRINT "OPCODE ERROR IN"Z:GOTO 200
00
4020 OI=(J-1)/3:L$=MID$(L$, I+5)
4030 IF OI=56 AND NOT F THEN A$=L$:GOSUB 5000:A(LC-1)=S
4032 IF OI=57 THEN L=1:FOR J=1 TO LEN(L$):L=L-(MID$(L$,J,1)=""):NEXT
4035 IF OI=58 THEN A$=L$:GOSUB 5000:PC=S
4040 IF OI=55 THEN RETURN
4045 IF OI>21 THEN A$=L$:GOTO 4135
4050 IF LEN(L$)=0 OR L$="A" THEN B$="A":GOTO 4140
4060 A$="":B$="":FOR I=1 TO LEN(L$):J$=MID$(L$, I,1)
4070 IF J$="H" OR J$="C" THEN B$=B$+J$:GOTO 4130
4080 IF J$=" " OR J$=" " THEN 4130
4100 IF J$=" ", THEN I=I+1:B$=B$+MID$(L$, I,1):GOTO 4130
4110 IF J$=" "; THEN 4135
4120 A$=A$+J$
4130 NEXT
4135 AI=0:L=- (OI>21)-(OI>46)-(OI>54):IF L THEN RETURN
4140 FOR AI=0 TO 7:IF B$(AI) THEN NEXT:GOTO 19990
4150 L=1-(AI>0)-(AI>3):IF (OI=12 OR OI=15) AND AI=4 THEN AI=7
4152 IF (2+AI AND O%(OI))=0 THEN 19990
4155 IFAI<6 THEN RETURN
4160 GOSUB 5000:IFE=10 OR OI=7 THEN L=3:RETURN
4170 L=2-(S>255)
4175 IFAI=7 THEN IF L=3 AND (OI=10 OR OI=12) THEN 19990
4190 RETURN
5000 FOR I=1 TO LEN(A$):J=ASC(MID$(A$, I,1))
5010 IF J=43 THEN A$=MID$(A$, I+1):A$=LEFT$(A$, I-1):GOSUB 5040:SA=S:A$=SA$:GOSUB 5040
:S=S+SA:RETURN
5020 IF J=45 THEN A$=MID$(A$, I+1):A$=LEFT$(A$, I-1):GOSUB 5040:SA=S:A$=SA$:GOSUB 5040
:S=SA-S:RETURN
5030 NEXT
5040 IF ASC(A$)=62 THEN A$=MID$(A$,2):GOSUB 6000:S%=S/256:S=S-256*S%:RETURN
5050 IF ASC(A$)=60 THEN A$=MID$(A$,2):GOSUB 6000:S=INT(S/256):RETURN
6000 E=0:S=VAL(A$):IFS>0 OR ASC(A$)=48 THEN RETURN
6010 J=ASC(A$):IF J=36 THEN 6030
6020 FOR J=2 TO LEN(A$):J$=MID$(A$,J,1):S=16*S+VAL(J$)+(ASC(J$)-55 AND J$)"@":NEXT:R
ETURN
6030 IF J=37 THEN FOR J=2 TO LEN(A$):S=S+S+VAL(MID$(A$,J,1)):NEXT:RETURN
6035 IF J=42 THEN S=PC+L:RETURN
6040 FOR J=0 TO LC-1:IFA$(J) THEN S=A(J):RETURN
6050 NEXT:IF OI=7 THEN E=1:RETURN
6060 A$(LC)=A$:PRINT "DER LABEL "A$
6070 PRINT "WURDE NICHT GEFUNDEN."
6080 INPUT "ZUGEWIESENER WERT":A$:GOSUB 5000
6090 A(LC)=S:LC=LC+1:RETURN
10000 DATA ORASLBIT AND ROLE ORLSR JMP ADCROR STYSTASTXLDYLDALDXCPYCMPCDECCPXSBINC
10010 DATA BRKPHPCPLPSECRTPHACLIRTSPLASEIDEYTXATYATXSTAYTAXCLVTXININDEXCLDINXN
OP
10020 DATA SEDBPLBMBVBCBVBCCBCBNEBEQJSREQUDFBORG
10030 DATA A$,X,X,Y,Y,C,"",X
10040 DATA 5,6,36,37,38,69,73,63,101,102,132,133,134,164,165,166,196,197,198,228,

```

```

229
10050 DATA230,0,8,24,40,56,64,72,88,96,104,120,136,138,152,154,168,170,184,186,2
00
10060 DATA202,218,232,234,248,16,48,80,112,144,176,208,240,32
10070 DATA4,-4,-4,12,12,32,0,16
10080 DATA222,193,64,222,193,222,193,36,222,193,192,222,194,194,222,194,66,222,2
04,66
10090 DATA222,204
19990 PRINT"ADR ERROR IN"Z
20000 PRINT"BEINLADEN VOM: 0
20005 PRINT"EDITOR ?":POKE198,0:WAIT198,1:GETG$:IFG$="J"THENCLR:LOAD"EDITOR",8
20010 PRINT"DISASSEMBLER ?":WAIT198,1:GETG$:IFG$="J"THENCLR:LOAD"DISASSEMBLER",
8
20020 END
20030 CLOSE1
50000 PRINT"KASSETTE ZURUECKSPULEN":PRINT"DANN P DRUECKEN!"
50005 GETWA$:IFWA$(">P"THEN50005
50010 PRINT:RETURN
60000 INPUTA$:A$="X"(A$:GOSUB6000:PRINTS:GOTO60000
S1000 OPEN1,8,2,"LC,S,R
61010 INPUT#1,A(LC):INPUT#1,A$(LC):LC=LC+1:IF(64ANDST)=0THEN61010
61020 CLOSE1
READY.

```

Leserbriefe

Wenn ich bei meinem Commodore 64 Zeichensätze programmieren will, packt mich immer wieder der Frust angesichts der Tatsache, daß auf diese Weise unglaublich viel Speicherplatz verbraucht wird. Warum reduziert ein Zeichensatz mit 2K Umfang den verfügbaren RAM von 38K auf "magere" 12K? Ich habe zwar den Verdacht, daß das Handbuch hier einige Auskünfte anbietet: verstehen tue ich die aber nicht. Können Sie mir helfen?

M. Kettwig

Redaktion:

Nun, die Ursache für Ihr Problem liegt wohl darin, daß Sie Ihren Zeichensatz in den ersten 16K-Bereich Ihres C-64 stellen. Was Sie in Angriff nehmen sollten, nennen die Fachleute bankswitching. Der Begriff besagt, daß es durchaus möglich ist, einzelne Speicherbereiche so zu schalten, daß bestimmte Teile ausgeklammert, andere wieder gezielt genutzt werden.

Sie schreiben Ihren Zeichensatz in die "Bank" 0 (\$0000-\$3FFF), wodurch die Adressen des Commodore-Basic verschoben werden. Wenn Sie "Bank" 3 benutzen, werden Sie wesentlich mehr Platz zur Verfügung haben. Benutzen Sie doch einmal folgendes Programm:

```

10 POKE 56576,4
20 POKE 53272,32
30 POKE 648,200
40 FOR I = 0 TO 2047
50 POKE 56333,127
60 POKE 1,51
70 J = PEEK (53248 + I)
80 POKE 1,55
90 POKE 56333,129
100 POKE 49152 + I,J
110 NEXT I
120 END

```

Das Programm ist zugegebenermaßen etwas langsam, weil in BASIC geschrieben, für Ihre Belange allerdings goldrichtig.

Ihr Zeichensatz liegt jetzt im Bereich zwischen 49152 und 51199. Den bestehenden Satz können Sie nun ändern, weil er durch die oben abgedruckte Routine in den RAM kopiert wurde.

Ich besitze einen VC-20 und eine 16K-Erweiterung. Dazu habe ich einen "Dr. Watson"-Assembler geschenkt bekommen.

Diese Software verfügt über einen nützlichen Befehl (CHR GET), mit dem sich Satzkommandos erzeugen lassen. Ich würde gern, wie ich das machen kann, da

SPONTAN; WIE UNSERE LESER ...

haben Sie in Ihrer neuen "Compute mit..." nicht nur sofort nach Erscheinen der Erstausgabe eine Menge Lösungen für das Suchrätsel eingesandt, nein, sie sorgten auch für regen Posteingang in der Redaktion.

So können wir die allerersten Leserbriefe heute veröffentlichen und hoffen natürlich, daß Sie, liebe Leser, es dabei nicht bewenden lassen.

das Begleithandbuch recht unübersichtlich ist.

B. Melthaus

Redaktion:

Die CHR GET-Routine, die Sie ansprechen, liegt im RAM. Daher ist es durchaus möglich, Zusatzbefehle zu implementieren.

Wichtig ist allerdings, daß Sie mit SEI die Interrupts außer Kraft setzen, nach der Ergänzung durch CLI allerdings wieder aufrufen.

Aus der beiliegenden Fotokopie ersehen Sie einen Mini-Industrieroboter.

Da Sie bestimmt laufend Informationen über neue Peripheriegeräte erhalten, sollten Sie mit sicher Auskunft geben können, wo ich mir Unterlagen über solche Roboter beschaffen könnte.

Redaktion:

Da die Redaktion das vorliegende Modell nicht exakt identifizieren kann, gibt sie die Frage an unsere Leser weiter. Wir merken nur folgendes an:

Der Markt für Kleinstroboter (z.B. Mitsubishi MF101) ist inzwischen beachtlich gewachsen. Bezugsquellen nennt vielfach der Fachhandel.

ZACK

39/
84

>CM-RÄTSEL<

Einsendeschl.
ist
der

29.09.
84

In der
richtigen (!)
Reihenfolge
ergeben die folgenden Silben
eine Devise des erfolgreichen
Commodore-Users ...



AL - COM - DET - ERST - FIN - GE - IN - KLAS -
LEM - MAN - MIT - PU - SI - SOFT - TE - VOR - WARE

□ □ □ □

□ □

□ □

□

□

□ □

□

□ □ □

□

17 Silben



Wie bekannt, gibt's für die Sieger tolle Spielprogramme auf Bändern aus unserem Kassetten-Service. Mitmachen darf jeder mit Ausnahme der Angestellten des Roeske-Verlages sowie deren Angehörigen. Rechtsweg wie üblich ausgeschlossen. Lösungen an: ROESKE-VERLAG, Redaktion CM, Postfach 620, D-3440 Eschwege.

1. PREIS

5

KASS.

2. PREIS

3

KASS.

3. PREIS

2

KASS.

4. - 25. PREIS

1

KASS.

