

monatlich

Compute mit

COMMODORE & SCHNEIDER

2/86

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

Unabhängiges Magazin für Anwender von Commodore- und Schneider-Computern

2,80 DM
24 öS
2,80 sFr

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

News

MADGAMES

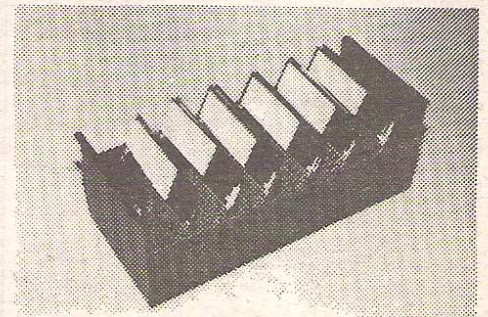
Wie die „Compute mit“-Redaktion bereits in Heft 12/85 berichtete hat jetzt Mastertronic ihre neuen M.A.D.-Games vorgestellt.

Mit diesen MADGAMES, Mastertronic Added Dimension, hat Mastertronic den Superhit gelandet. Die MADGAMES bieten für einen Preis von rund 15,-- DM Grafik, Sound und Action, wie es sonst nur Spiele in der Preisklasse zwischen 49,-- DM und 99,-- DM können. Zwei dieser MADGAMES wurden jetzt vorgestellt und sind ab sofort erhältlich. „Hero of the golden Talisman“ bietet in über 60 verschiedenen Räumen jeweils vier verschiedene Computerbildschirme. Dazu kann man nur sagen: „Action live“. Nicht weniger fantastisch sind die grafischen Leistungen des Spiels „The last V8“. Neben einer außergewöhnlichen Grafik besitzt „The last V8“ noch eine echte Sprachsynthese, einen Sound, der kaum überbietbar ist.



Beide Spiele stehen für den Commodore 64 und Commodore 128 zur Verfügung und kosten wie alle Computer-Spiele der MADGAMES-Serie nur ca. 15,-- DM.

Weitere Software-Neuerscheinungen auf S. 63 dieser Ausgabe.



BRANDNEU

In unserer heutigen Ausgabe stellen wir Ihnen zwei neue Diskettenboxen vor, die ab sofort überall im Handel erhältlich sind. Gleichzeitig veröffentlichen wir einen Bericht über den Kauf von Disketten:

- welche Kriterien müssen beim Kauf von Disketten berücksichtigt werden
- welche Disketten für welche Zwecke
- wie berechnet man die Speicherkapazität einer Diskette
- usw.

Lesen Sie mehr darüber auf Seite 10.

**Tips & Tricks – Assembler-Kurs Teil 9 – Software-Abo – Werkstatt – Report
Leser + Meckerecke – Software-Reviews – Kleinanzeigen – Software-Service
Robo – Raketenstart – Dreikampf – Defender – Alien 2001 – Diamond-Hunter**

REPORT

- Nützliche Tips zum Diskettenkauf **10**
- Rolli-Disk –
Diskettenkasten für Hängeregister **10**
- Misco Roll Top –
platzsparende Diskettenbox **11**
- Anschluß von C-64-Drucker
an Schneider-Computer **21**
- Neuer Monochrom-Datenmonitor
für Commodore-Homecomputer **21**
- Robcon – Module für den C-64/128 **22**
- Messevorschau: Mikrocomputer '86 **22**
- Anschluß von zwei Computern
an einen Drucker **22**
- Was gibt's Neues auf dem
Software-Markt **63**

BÜCHER

- Praktisches Computerwissen **11**

Leser + Meckerecke

Software-Service

Assembler-Kurs

- Teil 9 **43**

TIPS & TRICKS

- Schneider **52**

Bewerbung als Programmautor

- 53**

KLEINANZEIGEN

- 54**

Software

Commodore

- Checksummer (C-64) **3**
- Robo (C-64) **5**
- Basic-Befehlserweiterung (C-64) **14**
- Raketenstart (C-16) **23**
- Dreikampf (C-16) **29**
- Defender (VC-20) **34**
- Formel 1 (VC-20) **40**

Schneider

- Checksummer **45**
- Alien 2001 **46**
- Diamond Hunter **56**

WERKSTATT

- Headline für den C-16 **26**
- Memory-Display für den VC-20 **38**
- Directory für Schneider **50**

„Compute mit“

erscheint am

17. Februar

wieder überall

im Zeitschriftenhandel.

Impressum

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

Redaktion:

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

Freie Mitarbeiter:

Volker Becker,

Gesamtherstellung:

Druckhaus Dierichs Kassel, Frankfurter Str. 168, 3500 Kassel

Vertrieb:

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

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

Erscheinungsweise:

Erstverkaufstag von „Compute mit“ jeweils Mitte des Monats.

Urheberrecht:

Alle in „Compute mit“ veröffentlichten Beiträge sind urheber-
rechtlich geschützt. Alle Rechte, auch Übersetzungen, vor-
behalten.

Reproduktionen jeder Art (Fotokopien, Microfilm, Erfassung in
Datenverarbeitungsanlagen usw.) bedürfen der schriftlichen
Genehmigung des Verlags. Alle veröffentlichte Software wurde
von Mitarbeitern des Verlages oder von freien Mitarbeitern
erstellt.

Aus ihrer Veröffentlichung kann nicht geschlossen werden,
daß die beschriebenen Lösungen oder Bezeichnungen frei von
Schutzrechten sind.

Bezugspreis:

Einzelheft 2,80 DM
Abonnement: Inland 30,- DM im Jahr (12 Ausgaben)
Ausland: Europa 40,- DM
ohne Kassetten!

Autoren, Manuskripte:

Der Verlag nimmt Manuskripte und Software zur Veröffentli-
chung gerne entgegen.

Sollte keine andere Vereinbarung getroffen sein, so gehen wir
davon aus, daß Sie mit einem Honorar von 120,- DM pro abge-
druckter Seite im Heft einverstanden sind.

Bei Zusendung von Manuskripten und Software erteilt der Au-
tor dem Verlag die Genehmigung zum Abdruck und Versand
der veröffentlichten Programme auf Datenträger.

Rücksendung erfolgt nur gegen Erstattung der Kosten. Zusen-
dungen von Software zur Veröffentlichung sollten folgendes
enthalten:

Kopierfähige Kassette oder Diskette mit dem Programm (Com-
puter-Bezeichnung), von Drucker erstelltes Listing oder Serie
von Bildschirmfotos (keine Schreibmaschinenlistings), evtl.
Bildschirmfotos von einem Probelauf und ausführliche Pro-
grammbeschreibung (Erklärung der Variablenliste, Beschrei-
bung des Bildschirmaufbaues, Farbe, Grafik usw.). Für einge-
sandte Programmunterlagen kann keinerlei Haftung übernom-
men werden.

Anzeigenpreise:

Bitte Mediaunterlagen anfordern.

Anzeigenverwaltung:

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

Anzeigenleitung:

Inland: H. Wendt
Ausland: M. Kleimann



Neu: Sonderzeichen werden durch Klartext ersetzt!

Eingabe-Hinweise für Commodore C-64-Benutzer!

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

Tastenbezeichnungen für Steuercodes

(TRONIC-NORM)

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

Achtung: Einige Listings enthalten das Zeichen "-", dieses Zeichen entspricht dem Pfeil nach links (Taste ganz links oben)

Folgendes Beispiel demonstriert die Arbeitsweise unseres Systems:

STANDARD) AUSDRUCK:

```
10 A$="TEST":PRINT"HALLO LEUTE !":PRINT"TRONIC VERLAG"
20 PRINT"ENDE"
```

UNSER NEUER AUSDRUCK DER GLEICHEN ZEILEN:

```
10 A$="TEST":PRINT" {CLEAR RIGHT3 DOWN3 R
ED}HALLO {RIGHT2 BLUE}LEUTE {SPACE}!":PRIN
T"TRONIC {SPACE}VERLAG" <34>
20 PRINT" {CYAN PURPLE GREEN}ENDE" <117>
```

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

Um im Listing Tastenbezeichnungen von Grafik oder normalen Texten zu unterscheiden, werden alle Tastenkennzeichnungen in geschweifte Klammern gesetzt. Diese dürfen selbstverständlich nicht eingegeben werden. Auch Leerzeichen innerhalb geschweifeter Klammern

dienen nur zur Trennung einzelner Tastenfunktionen und dürfen ebenfalls nicht eingegeben werden. Um die Tastenfunktionen noch besser hervorzuheben, werden diese in unterstrichener Kursivschrift (Schrägschrift) abgedruckt.

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

zeichen innerhalb von Anführungszeichen werden auf diese Art gekennzeichnet. Das bislang übliche Abzählen einzelner Zeichen entfällt somit völlig.

Alle Zeichen außerhalb von den geschweiften Klammern werden normal abgedruckt und auch eingegeben. Auf den ersten Blick hört sich das sicher etwas kompliziert an, ist jedoch in der Praxis ganz einfach. Wenn man sich erst an die in Klartext geschriebenen Steuerzeichen gewöhnt hat, wird man den großen Vorteil dieser Schreibweise erkennen.

Checksummer C Version 1.0 für Commodore 64

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

Arbeitsweise und Aufbau unseres Checksummers:

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

Der Start erfolgt durch den Befehl „RUN“. Nach kurzer Zeit meldet sich der Rechner mit der Meldung „TRONIC...“. Der Checksummer ist nun aktiv und man kann ein beliebiges Tronic-Listing eingeben. Nachdem eine Zeile mit RETURN abgeschlossen wird, erscheint links oben auf dem Schirm eine Prüfzahl. Vergleichen Sie diese mit der Zahl im Heft hinter der Zeile. Stimmt die Zahl überein, ist die Zeile richtig eingegeben.

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

Interessant ist auch, daß bei der Eingabe von Zeilen die üblichen Abkürzungen benutzt werden können, ohne die Checksumme zu verändern. Leerzeichen außerhalb von Anführungszeichen werden ignoriert, da diese auf die

Ausführung der einzelnen Befehle keinen Einfluß haben. Sie können Ihr Programm auch starten, denn der Checksummer und Ihr Programm beeinflussen sich nicht gegenseitig. Wollen sie den Checksummer abschalten, geben Sie einfach POKE 1,55 ein oder betätigen Sie die Tasten-Kombination RUNSTOP und RESTORE.

Aktivieren können Sie den Checksummer jederzeit (auch nach RESET) mit POKE 1,53.

Wer den Checksummer nicht eingeben möchte kann diesen auch unter der folgenden Bestellnummer beziehen:

Bestellnummer:	
CV10K Kassette	10 DM
CV10D Diskette	15 DM

```

0 REM ***** TRONIC CHECKSUMMER ***** <116>
1 REM VERSION C 1.0 BEI FRANK BRALL <10>
2 REM <148>
3 FOR I=832 TO 1008:READ A:S=S+A <68>
4 POKE I,A:NEXT I:IF S<>16397 THEN PRINT
"FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE
}!{SPACE}(10-19)":STOP <32>
5 SYS 832:S=0:FOR I=58464 TO 58603:READ
A:S=S+A <21>
6 POKE I,A:NEXT I:IF S<>18919 THEN PRINT
"FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE
}!{SPACE}(29-36)":STOP <219>
7 S=0:FOR I=48979 TO 49005:READ A:S=S+A <33>
8 POKE I,A:NEXT I:IF S<>2888 THEN PRINT
"FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE
}!{SPACE}(46-47)":STOP <23>
9 PRINT"DOWN2AKTIVIEREN{SPACE}:{SPACE}
POKE{SPACE}1,53{SPACE18}AUSSCHALTEN:{SPA
CE}POKE{SPACE}1,55":NEW <26>
10 DATA169,0,133,254,162,1,189,137,3,133
,255,160,0,177,254,145,254,136 <176>
11 DATA208,249,230,255,165,255,221,139,3
,208,238,202,16,230,169,96,141,49 <147>
12 DATA165,169,228,141,50,165,169,53,133
,1,169,141,133,254,162,3,134,255 <225>
13 DATA160,0,177,254,240,7,32,202,241,20
0,76,120,3,169,2,141,32,208 <140>
14 DATA96,160,224,192,0,147,17,32,32,35,
35,35,35,32,84,82,79,78 <158>

```

```

15 DATA73,67,45,86,69,82,76,65,71,32,67,
72,69,67,75,83,85,77 <129>
16 DATA77,69,82,32,35,35,35,139,13,13
,32,32,32,32,32,32 <147>
17 DATA32,32,32,32,32,86,69,82,83,73,79,
78,32,67,32,49,46,48 <79>
18 DATA13,13,32,32,32,32,32,40,67,41,
32,70,82,65,78,75,32 <94>
19 DATA66,82,65,76,76,32,40,49,48,46,56,
53,41,13,0 <12>
29 DATA169,0,141,107,191,141,108,191,141
,109,191,160,2,24,177,95,200,113 <7>
30 DATA95,141,109,191,160,3,200,177,95,2
40,44,201,34,208,10,173,107,191 <91>
31 DATA73,1,141,107,191,177,95,174,107,1
91,208,4,201,32,240,228,238,108 <128>
32 DATA191,174,108,191,24,177,95,109,109
,191,141,109,191,202,208,244,76,120 <244>
33 DATA228,56,76,183,228,72,32,201,255,1
70,104,144,1,138,96,32,240,255 <101>
34 DATA142,105,191,140,106,191,162,0,160
,0,24,32,240,255,169,91,32,12 <89>
35 DATA225,169,0,174,109,191,32,205,189,
169,93,76,83,191,173,33,208,145 <239>
36 DATA193,96,105,2,164,145,200,208,4,19
7,161,208,247,96 <202>
46 DATA32,12,225,32,63,171,32,63,171,172
,106,191,174,105,191,24,32,240 <213>
47 DATA255,76,128,164,24,0,0,5,200 <65>

```

Software: Flugsimulatoren, Psycho, Lotto, Diagnose, Horoskop, Buchhalter, Assemblerkurs, Schach, Kopierschutz K37.

Hardware: Turbo Floppy, Zusatztastatur, Akustik-Koppler, Speichererweit., Moduladapter, 80 Zeichenkarten, Eprommer + Karten + Eproms, RS 232 und ... und ...

Zubehör: Staubschutzhauben, Reset-Taster, Stecker, Floppy-Kühler, Tastaturmasken, HiFi-Kabel und ... und ...

Über 500 Hard- und Software-Ideen!
Commodore-Katalog anfordern 2.50 DM (Briefmarken)

Jetzt auch alles für ... **SCHNEIDER**
Die neuesten Spiele, Geräte, Bücher, Zubehör und ... und ...
SCHNEIDER-Katalog anfordern 1.20 DM (Briefmarken)

mükra
DATEN-TECHNIK

Schöneberger Str. 5
1000 Berlin 42 (Tempelhof)
☎ 030-752 91 50/60

Öffnungszeiten:
Mo-Fr: 10-18 Uhr
Sa: 10-13 Uhr

Händler-
anfragen
erwünscht

SOFT-TEAM-BERLIN

Software und Zubehör für
C-64 Atari Schneider Spectrum MSX

Über 2000 Programme immer auf Lager.
Ständig die neuesten Programme aus England und den USA.

Gratiskatalog anfordern:
**Anton Peter & Partner, Kamminerstraße 9
1000 Berlin 10, Telefon 453 27 11**

TOP-GAME
Commodore 64

Robo

TOP-GAME
Commodore 64

Ziel des Spiels:

Das Spiel ist eine Mischung von „LODE-RUNNER“, „MR. ROBOT AND HIS FACTORY“, „HARD HAT MACK“ usw.

„ROBO“ ist ein Roboter, der sich in einem Labyrinth von Aufzügen, Steinen, Falltüren, Förderbändern usw. seine Nahrung in Form von Disketten suchen muß. Der Spieler erhält dabei für jede Diskette 10 Punkte. Nun die einzelnen Funktionen des Roboters:

GEHEN: (Joystick links/rechts)

Nach links und rechts: auf Förderbändern ist keine Änderung der Richtung bzw. kein Stillstand möglich. Auf allen anderen Feldern kann er sich in beide Richtungen bewegen.

AUF- UND ABBEWEGEN: (Joystick rauf/runter)

Aufwärts mit dem Aufzug; auf- und abwärts nur auf der Leiter.

GRABEN: (Feuerknopf)

Gegraben kann nur auf dem braunen Ziegel werden; „ROBO“ fällt dadurch um ein Feld nach unten.

ZERTRETEN:

Geht man über ein Feld Nr. 5, zerbricht dieses. Bleibt man darauf stehen, fällt man nach unten.

SPRINGEN: (Feuerknopf)

Springen kann man auf allen Feldern, außer Feld Nr. 1 (Diskette), Feld Nr. 3+4 (Förderbänder) und Feld Nr. 7 (Brauner Ziegel). Im höchsten Punkt des Sprunges kann man 1 Feld nach links bzw. rechts gleiten (durch entsprechende Bewegung des Joysticks).

ZERSTÖREN DES ROBOTERS:

Fällt „ROBO“ über eine Distanz von mehr als 2 Feldern frei nach unten, so geht er kaputt. Pro Spiel hat man eine Reserve von 5 Robotern. Verläßt der Roboter in irgendeiner Richtung den Bildschirm, ist er ebenfalls verloren.

NÄCHSTES LEVEL:

Hat man es geschafft alle Disketten zu erwischen, so kommt man ins nächste Level.

Will man dem Roboter ein vorzeitiges Ende bereiten, drückt man einfach die „Q“-Taste.

Feld 1 - DISKETTE



Nahrung für den Roboter. Sie wird durch Berühren des Roboters aufgenommen.

Feld 2 - AUFZUG



Der Roboter wird nach oben befördert. Nach links und rechts kann abgesprungen werden.

Feld 3 - FÖRDERBAND LINKS



Der Roboter wird nach links befördert; keine Bewegung möglich.

Feld 4 - FÖRDERBAND RECHTS



Wie Feld 3, nur rechts

FELD 5 - STEIN ZERBRECHLICH



Dieses Feld kann nur einmal betreten werden, da es sofort zerbricht. Alle Richtungen, außer graben, sind möglich.

Feld 6 - STEIN FEST



hellrot

Unzerbrechlich; sonst wie Feld 5.

Feld 7 - STEIN WEICH



braun

Wie Feld 5; wird jedoch nur zerstört, wenn der Feuerknopf gedrückt ist. Springen ist daher unmöglich.

Feld 8 - LEITER



Alle Richtungen sind möglich.

Starten des Spiels:

Nach dem Laden und Starten des Programms „ROBO“ erscheint vorerst das Titelbild. Nach der Aufforderung „PRESS ANY KEY“ drückt man eine Taste bzw. den Feuerknopf des Joysticks an Port 2. Jetzt erscheint Game-Menue. Man hat nun folgende Auswahl:

1.) PLAY GAME

Start des Spiels. Es ist darauf zu achten,

programme

daß sich die Diskette mit den Levels im Laufwerk befindet. Die Anzahl der Levels spielt keine Rolle. Wenn sich z. B. nur 5 Levels wie in diesem Fall auf der Diskette befinden, wird nach Level 05 wieder mit Level 01 begonnen.

Während des Spiels kann die Anzahl der Roboter mit der „Q“-Taste vermindert werden. Dies ist dann sinnvoll, wenn sich der Roboter in einer Sackgasse befindet, aus der er sich nicht mehr befreien kann.

2.) LEVEL EDITOR

Man gelangt in Editor-Menue des Level Editors. Eine genaue Beschreibung folgt nachstehend.

3.) SEE SCORES

Die besten 5 Spielergebnisse werden aufgelistet. Durch das Drücken einer Taste zurück ins Menue 1.

4.) SAVE SCORES

Die Bestenliste wird auf Disk geschrieben. Dann wird der Disk-Status und eventuelle Fehlermeldung der Diskettenstation angezeigt. Durch Druck einer Taste zurück ins Game-Menue.

Der Level Editor:

Durch Drücken der Taste „F3“ im Game-Menue gelangt man in das Editor-Menue. Nun hat man folgende Möglichkeiten:

1.) EDIT LEVEL

Wählt man diese Funktion, wird als erstes die Nummer des zu editierenden Levels erfragt. Diese kann zwischen 1 und 99 liegen. Man kann also bis zu 99 eigene Levels erstellen. Natürlich können auch bereits vorhandene Levels geändert bzw. gelöscht werden. Die Anzahl der Möglichkeiten ist beinahe unendlich. Hat man nun die Nummer des Levels eingegeben, wird dieser, falls vorhanden, von der Diskette geladen und kann editiert werden. Andernfalls wird er neu erstellt. Im oberen Drittel des Bildschirms erscheint nun ein Balken, der die verschiedenen Felder zur Editierung beinhaltet. Mit dem Joystick

wird nun ein Feld angewählt, indem man den weißen Rahmen (Cursor) auf das gewünschte Feld bewegt und die Feuertaste drückt. Nun kann man in den unteren beiden Dritteln des Bildschirms sein eigenes Level erstellen. Will man das Level löschen, drückt man die „C“-Taste. Hat man ein Level fertig erstellt, wird es mit der „Q“-Taste in den vorgesehenen Level-Speicher übernommen.

2.) PLAY LEVEL

Bevor man ein neu erstelltes Level auf Diskette speichert, sollte man ein Probespiel machen, da es nicht sinnvoll wäre, ein Level zu übernehmen, das man unmöglich schaffen kann, da es möglicherweise eine Sackgasse enthält.

3.) SAVE LEVEL

Hat man ein neues Level erstellt und getestet, kann man es mit dieser Funktion auf Diskette speichern. War ein Level mit dieser Nummer bereits auf Diskette vorhanden, wird es überschrieben.

4.) RETURN TO GAME

Mit dieser Funktion gelangt man wieder zurück ins Game-Menue.

BEMERKUNG:

Das Spiel „ROBO“ ist bereits mit einem Level lauffähig. Das Programm arbeitet folgendermaßen:

Bei Start des Spiel wird auf der Diskette nach Level 01 gesucht; wird dieses gefunden, startet das Spiel. Hat man das Level 01 geschafft, wird Level 02 geladen usw.; Wird nun kein fortlaufendes Level mehr auf der Diskette gefunden, wird wieder mit Level 01 begonnen.

Ladeanweisungen:

1.) „ROBO1“, „ROBO2“ und „ROBO3“ abtippen und speichern. Jetzt „ROBO LEVELMAKER“ abtippen und auf Disk ab speichern.

Nun laden Sie „ROBO1“ und starten das Programm. Dieses Programm erzeugt die neuen Zeichen, verschiebt den BASIC-Speicherbeginn, lädt „ROBO2“ von Diskette und startet das selbe. „ROBO2“ erzeugt den Maschinencode und die 8 Sprites. Dann verschiebt es den BASIC-Speicher wieder zurück, schreibt ein kleines Startprogramm, verschiebt den BASIC-Speicher abermals und lädt zum Abschluß „ROBO3“

2.) Während des Ladens von „ROBO3“ erscheint auf dem Bildschirm folgende Zeile:

POKE646,0:POKE44,8:SAVE"ROBO",8

Wenn „ROBO3“ fertig geladen ist, springt der Cursor in diese Zeile. Sie brauchen nun nur „RETURN“ zu drücken, und das fertige Programm „ROBO“ wird auf Diskette gespeichert.

Diese Vorgänge sind natürlich nur beim ersten Erstellen des Programms notwendig. Soll das Programm später wieder geladen werden, geschieht das einfach mit **LOAD"ROBO",8**.

Bevor allerdings gespielt werden kann, muß noch das Programm „ROBO LEVELMAKER“ geladen und gestartet werden. Dieses Programm schreibt nun die ersten 5 Spiel-Levels auf Diskette.

Wenn Sie alle diese Punkte nacheinander ausgeführt haben, kann das Spiel beginnen.

Teil 1:

C64 LISTING MIT CHECKSUMMEN (C V1.0)

```
100 REM***** <211>
101 REM* ROBO1 ZEICHEN * <136>
102 REM* (C)1985 BY MR-SOFT * <8>
103 REM* RONALD MAYER * <128>
104 REM* A-4062 THENING 24/AUSTRIA * <168>
105 REM***** <216>
106 POKE6,28:CLR:POKE3281,5:POKE53280,
S:PRINT"⟨CLEAR BLACK⟩PLEASE⟨SPACE⟩WAIT..
" <70>
107 S=0:FORI=6000TO6041:READA:S=S+A:POKE
I,A:NEXT <102>
108 IFS<>6785THENX=110:Y=112:GOTO160 <212>
109 SYS6000 <183>
110 DATA20,169,51,133,1,160,,132,254,13
```

Teil 1:

```

2,252,162,32,134,255,162,208,134,253 <227>
111 DATA177,252,145,254,200,208,249,230, <127>
253,230,255,165,255,201,48,144,239,162 <156>
112 DATA55,134,1,88,96 <224>
113 REM***** <45>
114 REM* CODES FUER ZIFFERN * <226>
115 REM***** <252>
116 S=0:FORI=8496TO8655:READA:S=S+A:POKE <205>
I,A:NEXT
117 IFS<>27624THENX=118:Y=126:GOTO160
118 DATA126,60,60,60,66,129,255,255,254, <81>
252,252,252,252,254,255,255,67,63,63,63
119 DATA67,129,255,255,194,252,252,252,1 <214>
94,129,255,255,194,252,252,252,252,254
120 DATA255,255,194,252,252,252,194,129, <212>
255,255,66,60,60,60,66,129,255,255,254
121 DATA252,252,252,252,254,255,255,66,6 <108>
0,60,60,66,129,255,255,194,252,252,252
122 DATA194,129,255,255,255,255,129,66,6 <221>
0,60,60,126,255,255,254,252,252,252,252
123 DATA254,255,255,129,194,252,252,252, <110>
194,255,255,129,194,252,252,252,194,255
124 DATA255,126,60,60,60,66,255,255,1 <212>
29,67,63,63,63,67,255,255,129,67,63,63
125 DATA63,67,255,255,129,66,60,60,60,12 <71>
6,255,255,129,66,60,60,66,255,255
126 DATA129,66,60,60,60,66 <197>
127 REM***** <238>
128 REM* CODES FUER ZEICHEN D-J * <119>
129 REM***** <240>
130 S=0:FORI=8736TO8791:READA:S=S+A:POKE <243>
I,A:NEXT
131 IFS<>9588THENX=132:Y=134:GOTO160 <138>
132 DATA223,223,223,0,251,251,251,0,223,
223,223,0,251,251,251,0,195,195,255 <54>
133 DATA195,195,195,255,195,255,255,255, <4>
255,0,0,0,0,15,15,15,15,15,15
134 DATA15,240,240,240,240,246,246,246,2 <106>
46,240,248,252,254,255,247,243,241
135 REM***** <246>
136 REM* CODES FUER ZEICHEN M,P,Q,? * <146>
137 REM***** <248>
138 S=0:FORI=1TO4:READA:S=S+A:FORJ=0TO7: <200>
READB:S=S+B:POKE8192+B*A+J,B:NEXTJ,I
139 IFS<>5089THENX=140:Y=143:GOTO160 <88>
140 DATA77,255,255,255,255,240,240,240,2 <198>
40
141 DATAB0,15,15,15,15,240,240,240,240 <56>
142 DATAB1,255,255,255,255,15,15,15,15 <144>
143 DATA63,0,126,126,102,102,126,126,0 <249>
144 REM***** <255>
145 REM* CODES FUER IRR * <157>
146 REM***** <1>
147 S=0:FORI=10240TO11008STEP256:FORJ=0T <207>
O31:READA:S=S+A:POKEI+J,A:NEXTJ,I
148 IFS<>20136THENX=149:Y=156:GOTO160 <46>
149 DATA255,255,0,0,255,255,0,0,204,204, <48>
255,231,231,255,0,0,204,204,255,231
150 DATA231,255,0,0,255,255,1,131,199,23 <105>
9,255,0
151 DATA255,0,0,255,255,0,0,255,153,153, <8>
255,231,231,255,0,0,102,102,255,231
152 DATA231,255,0,0,255,255,1,131,199,23 <107>
9,255,0
153 DATA0,0,255,255,0,0,255,255,51,51,25 <89>
5,231,231,255,0,0,51,51,255,231,231
154 DATA255,0,0,255,255,255,255,255,255, <149>
255,0
155 DATA0,255,255,0,0,255,255,0,102,102, <92>
255,231,231,255,0,0,153,153,255,231
156 DATA231,255,0,0,255,255,255,255,255, <212>
255,255,0
157 PRINT"⟨CLEAR GREEN⟩PI56,160:CLR:PI44

```

```

,48:PI12288,0:LI"CHR$(34)"ROB02"CHR$(34) <213>
",8"
158 PRINT"⟨DOWN4⟩RUN" <165>
159 POKE631,19:POKE632,13:POKE633,13:POK <34>
E198,3:STOP
160 REM***** <15>
161 REM* DATA ERROR * <103>
162 REM***** <17>
163 PRINT"⟨DOWN⟩DATA⟨SPACE⟩ERROR⟨SPACE⟩I <143>
N⟨SPACE⟩LINES"X"--"Y:END

```

Teil 2:

```

100 REM***** <211>
101 REM* ROB02 6510-CODE * <110>
102 REM* (C)1985 BY MR-SOFT * <8>
103 REM* RONALD MAYER * <128>
104 REM* A-4062 THENING 24/AUSTRIA * <168>
105 REM***** <216>
106 REM* MACHINE-CODE PART1 * <169>
107 REM***** <218>
108 POKES3281,5:POKES3280,5:PRINT"⟨CLEAR <29>
BLACK⟩CREATING⟨SPACE⟩MACHINE-CODE⟨SPACE
31"
109 S=0:FORI=4040TO4261:READA:S=S+A:POKE <160>
I,A:NEXT
110 IFS<>27514THENPRINT"DATA-ERROR1":END <91>
111 PRINT"DATAS⟨SPACE⟩1⟨SPACE⟩O.K." <98>
112 DATA169,0,133,252,169,160,133,253,16 <250>
2,32,160,0,177,252,145,252,200,208
113 DATA249,202,240,5,230,253,76,212,15, <115>
169,54,133,1,120,169,244,162,15,141
114 DATA20,3,142,21,3,88,96,230,139,165, <88>
139,201,3,208,39,169,0,133,139,230
115 DATA140,165,140,201,44,144,4,169,40, <103>
133,140,133,255,160,0,132,254,177
116 DATA254,153,0,34,200,192,32,208,246, <44>
165,141,240,3,32,38,16,76,49,234
117 DATA201,2,176,16,206,0,208,173,0,208 <43>
,201,255,208,3,206,16,208,76,133
118 DATA16,201,3,176,11,238,0,208,208,3, <161>
238,16,208,76,133,16,201,4,176,6
119 DATA206,1,208,76,133,16,201,5,176,9, <229>
238,1,208,238,1,208,76,133,16,201
120 DATAB,176,18,238,248,7,32,133,16,240 <135>
,1,96,169,8,162,1,133,142,134,141
121 DATAB6,206,248,7,32,133,16,208,16,16 <0>
9,8,162,2,76,113,16,198,142,165,142
122 DATA240,1,96,133,141,96,32,158,183,1 <157>
38,72,32,253,174,32,158,183,104,168
123 DATA24,32,240,255,32,253,174,76,164, <42>
170
124 REM***** <235>
125 REM* MACHINE-CODE PART2 * <207>
126 REM***** <237>
127 PRINT"CREATING⟨SPACE⟩MACHINE-CODE⟨SP <150>
ACE⟩2"
128 S=0:FORI=4315TO4848:READA:S=S+A:POKE <141>
I,A:NEXT
129 IFS<>67350THENPRINT"DATA-ERROR2":END <135>
130 PRINT"DATAS⟨SPACE⟩2⟨SPACE⟩O.K." <126>
131 DATA32,213,243,165,186,32,180,255,16 <237>
5,185,32,150,255,32,165,255,160,0
132 DATA145,252,230,252,208,2,230,253,16 <92>
5,144,240,239,76,66,246,169,192,160
133 DATA4,133,251,132,252,169,192,160,21 <70>
6,133,253,132,254,169,192,160,191
134 DATA133,167,132,168,160,88,96,120,32 <56>
,252,16,177,167,145,251,24,105,5
135 DATA145,253,200,208,244,230,168,230, <44>
254,230,252,165,252,201,7,144,232
136 DATAB8,96,169,0,141,22,192,120,32,25 <66>
2,16,177,251,145,167,201,63,208,3

```



```

148 GOSUB241: IFHH<H*(5) THEN153 <20>
149 GOSUB201: HH#=HH#+H# <76>
150 FORI=STO1STEP-1 <157>
151 IFHH#>H*(I) THENNEXT <235>
152 FORJ=STOI+1STEP-1: H*(J+1)=H*(J): NEXT <222>
: H*(I+1)=HH# <228>
153 GOSUB255: GOTO110 <95>
154 REM***** <123>
155 REM* GAME SET * <97>
156 REM*****
157 SYS4375: POKE141,0: POKE142,0: POKEV,31 <15>
: POKEV+1,224: POKE2040,41
158 POKEV+21,1: POKEV+16,0: POKE167,112: PO <135>
KE168,7: POKE2,0: POKE143,0
159 POKE49173,PEEK(49174): GOSUB241 <0>
160 POKES+24,15: SYS4440: RETURN <26>
161 REM***** <102>
162 REM* BEEP * <190>
163 REM***** <104>
164 POKES+1,50: POKES+5,0: POKES+6,240: POK <240>
ES+4,32: POKES+24,15: POKES+4,33
165 FORII=1TO50: NEXT: POKES+1,0: RETURN <187>
166 FORI=0TO24: POKES+I,0: NEXT: RETURN <170>
167 REM***** <108>
168 REM* SCREEN * <149>
169 REM***** <110>
170 PRINT "{CLEAR}": POKEBI,0: POKERA,11: PO <35>
KEHI,0
171 LETO,0,"{RVSON DGREY SPACE4}SCORE{SP <125>
ACES}LEVEL {SPACE2}MEN{SPACE3}HIGHSCORE {S
PACE2}"
172 LETO,1,"{RVSON SPACE} {SPA <226>
CE2} {SPACE2} {SPACE2} {
SPACE}"
173 FORI=2TO3 <4>
174 LETO,I,"{RVSON} {CYAN SPACE11 DGREY} <26>
I {KYAN SPACES DGREY}I {KYAN SPACE3 DBRE
Y}I {KYAN SPACE11 DGREY}I": NEXT
175 LETO,4,"{RVSON SPACE} {SPA <216>
CE2} {SPACE2} {SPACE2} {
SPACE}"
176 FORI=1106TO1114STEP2: POKEI,48: POKEI+ <165>
40,38: NEXT: RETURN
177 A=LE/10: POKE1119,48+A: POKE1159,38+A <17>
178 A=LE-INT(A)*10: POKE1121,48+A: POKE116 <120>
1,38+A
179 FORI=1TO5: A=VAL(MID$(H*(1),I,1)): POK <85>
E1129+I*2,48+A: POKE1169+I*2,38+A: NEXT
180 POKE1126,48+ME: POKE1166,38+ME <39>
181 SYS4375: X=10: Y=14: F=2 <20>
182 Z$="@LEVEL@"+RIGHT$(0+MID$(STR$(LL <10>
),2),2)+"@": GOSUB249
183 Z$="@<<<<<<<<<<@": X=10: Y=11: GOSUB249: Y <53>
=17: GOSUB249: POKEBI,23: RETURN
184 REM***** <125>
185 REM* SAVE/LOAD HIGHSCORES * <16>
186 REM***** <127>
187 OPEN2,8,2,"@:TOP5,S,W" <242>
188 GOSUB200: IFA<>OTHER190 <215>
189 FORI=1TO5: PRINT#2,H*(I): NEXT <99>
190 CLOSE1: CLOSE2: GOSUB196: RETURN <234>
191 OPEN2,8,2,"TOP5,S,R" <36>
192 GOSUB200: IFA<>OTHERGOSUB195: GOTO194 <210>
193 FORI=1TO5: INPUT#2,H*(I): NEXT <199>
194 CLOSE1: CLOSE2: RETURN <42>
195 FORI=1TO5: H*(I)="0010001RONNY": NEXT: <74>
RETURN
196 PRINT "{CLEAR}": X=8: Y=11: F=12: Z$="DIS <44>
K<STATUS": GOSUB249: Z$="": FORI=1TOLEN(F$)
197 A=ASC(MID$(F$,I,1)): IFA=32THENA=64 <56>
198 Z$=Z$+CHR$(A): NEXT: F=14: Y=15: X=19-1 E <161>
N(Z$)

```

```

199 GOSUB249: GOSUB241: RETURN <189>
200 OPEN1,8,15: INPUT#1,A,F$,C,D: RETURN <161>
201 REM***** <142>
202 REM* CONGRATULATIONS * <248>
203 REM***** <144>
204 PRINT "{CLEAR}": POKERA,0: X=0: F=5 <226>
205 Z$="<<<<<<<<<<<<<<<<<<": Y=0: GOSUB24 <120>
9: Y=6: GOSUB249: Y=21: GOSUB249
206 Y=3: A$="CONGRATULATIONS" <140>
207 FORI=1TOLEN(A$): X=2+I*2: F=I: Z$=MID$( <228>
A$,I,1): GOSUB249: GOSUB164: NEXT
208 X=1: Y=10: F=8: Z$="THIS@IS@ONE@OF@THE" <155>
: GOSUB249
209 X=2: Y=13: F=8: Z$="BEST@FIVE@SCORES?": <51>
GOSUB249
210 X=3: Y=18: F=3: Z$="YOUR@NAME:@:;:;:;:;: <187>
;:": GOSUB249
211 X=25: H$="" <221>
212 POKE198,0: WAIT198,1: GETZ$: IFZ$=""THE <242>
N212
213 IFZ$=CHR$(13)ANDX>25THENH$=LEFT$(H$+ <47>
"<<<<" ,5): RETURN
214 IFZ$<>CHR$(20) THEN216 <84>
215 IFX>25THENX=X-2: Z$=";:": GOSUB249: H$= <212>
LEFT$(H$,LEN(H$)-1): GOTO212
216 IFZ$<"A"ORZ$>"Z" THEN212 <206>
217 IFX<35THENGOSUB249: X=X+2: H$=H$+Z$: GO <227>
TO212
218 GOTO212 <34>
219 REM***** <160>
220 REM* TITLE * <126>
221 REM***** <162>
222 POKEHI,0: POKERA,9: PRINT "{CLEAR}": BOS <36>
UR166
223 POKES+4,128: POKES+1,5: POKES+5,9: POKE <13>
S+6,248
224 POKES+23,241: POKES+22,100: POKES+24,3 <207>
1
225 Y=2: F=5: X=3: Z$="MR<SOFT@PRESENTS: ": G <37>
OSUB249: POKES+4,129: POKEBI,23
226 FORI=0TO23: LETO,I,"{BROWN RVSON SPAC <173>
E RVSOFF LIG.BLUE}-": LET3,I,"-": NEXT
227 FORI=15TOOSTEP-1: POKES+24,16+I: NEXT <59>
228 FORI=6TO7: LET2,I,"{PURPLE} {SP <62>
ACE2} {SPACE2} {SPACE2}
-": NEXT
229 FORI=8TO9: LET2,I,"{LIG.RED} {SPACE3 <93>
} {SPACE2} {SPACE3} {SPACE2} {SPACE3
} {SPACE2} {SPACE3}": NEXT
230 LET2,10,"{ORANGE} {SPACE2} {S <76>
PACE3} {SPACE2} {SPACE2} {SPACE3
3}
231 LET2,11," {SPACE} {SPACE3} <150>
- {SPACE} {SPACE} {SPACE3}
232 FORI=12TO13: LET2,I,"{LIG.RED} {SPAC <111>
E3} {SPACE} {SPACE3} {SPACE} {SPAC
E3} {SPACE} {SPACE3}": NEXT
233 FORI=14TO15: LET2,I,"{PURPLE} {SPACE <198>
3} {SPACE} {SPACE} {SPACE}
E}": NEXT
234 LET14,20,"{RED}||||||| <224>
"
235 C$="{SPACE3}BY{SPACE}RONALD{SPACE}MA <75>
YER"
236 LET4,19,"{NGREY RVSON SPACE RVSOFF}1 <75>
{RVSON SPACE RVSOFF}9{RVSON SPACE RVSOFF
38{RVSON SPACE RVSOFF}5{RVSON SPACE RVSO
FF}"
237 LET4,20,"{NGREY RVSON SPACE RVSOFF} <19>
{RVSON SPACE RVSOFF}+{RVSON SPACE RVSOFF
3}{RVSON SPACE RVSOFF}+{RVSON SPACE RVSO
FF}"
238 FORI=1TO18: LET14,19,"{WHITE}"RIGHT$(

```

Fortsetzung Seite 12

Nützliche Tips zum Kauf von Disketten

Viele EDV-Anwender kennen die Probleme:

Zulange Lade- und Speicherzeiten bei den Datasetten.

Als Folge dieser etwas bedienerunfreundlichen Datasetten steigt eine Vielzahl von Anwendern auf Floppystationen um. Spätestens nach dem Kauf dieser Floppystationen, oder vielleicht auch schon vor dem Kauf, beginnt ein Problem:

Welches sind für meine Ansprüche und meinen Bedarf die richtigen Disketten?

Der Markt bietet derzeit eine Vielzahl von Disketten an, die alle für unterschiedliche Aufgaben angewandt werden. Da gibt es zunächst die großen 8" Disketten mit 48 tpi (tracks per inch). Danach die wohl z.Zt. noch am weitest verbreiteten 5.25" Disketten mit 48 oder 96 tpi und die 3.5" Micro-Flexy-Disks mit 67.5 oder 135 tpi. Die Aufzeichnungsdichte, ob doppelt oder

einfach erkennen Sie an den beiden Zeichen 2D (Flexy Disk 5.25 1D oder 2D). Jetzt beginnt das zweite entscheidende Kriterium. Von der Zahl der Aufzeichnungsseiten ist auch die Zahl der Gesamtkapazität meiner Diskette abhängig. Die meisten Anwender werden sagen, der Speicherplatz ist auf jeden Fall zu wenig. Mit Hilfe einer Kapazitätsberechnung kann man die theoretische Speicherkapazität einer Diskette berechnen.

Beispiel einer Kapazitätsberechnung für eine 5.25" Diskette:

Anzahl Sektoren	x	Bytes je Sektor	x	Zahl der Datenspuren	x	Aufzeichnungsseiten	=	Speicherkapazität
16	x	256	x	39	x	2	=	319.488 Bytes

Auf einer 5.25" Diskette mit 2 Aufzeichnungsseiten lassen sich also maximal 319.488 Bytes speichern. Auf diese Weise können Sie für jede Diskettengröße die Speicherkapazität errechnen.

Natürlich spielt aber auch die Länge der Datensätze eine wichtige Rolle. Bei festen oder variablen Satzlängen ändert sich nämlich auch die Sektorgröße einer Diskette.

Diese ausführlichen Berechnungen sprengen an dieser Stelle jedoch den Rah-

men unseres Berichtes. Da sollte jeder Anwender eine individuelle Berechnung der Daten vornehmen.

Wichtig für den Kauf ist aber neben den Kapazitäten auch die vorhandene oder die zu erwerbende Hardware. Nicht jede Disk paßt in jede Diskettenstation.

Genauso paßt nicht jede Diskettenstation an jeden Rechner usw.. Man ist hier also schon auf die Vorgaben der Hersteller beschränkt und sollte vor dem Kauf von Disketten genau die Angaben der Hersteller

in Punkto Größe und Typenbezeichnung beachten.

Als letzten entscheidenden Gesichtspunkt muß man auch noch die unterschiedlichen Preise der einzelnen Herstellerfirmen für Disketten berücksichtigen. Oftmals ist das Angebot so vielfältig, daß nur durch direkten Preisvergleich Geld gespart werden kann. Beachten Sie also bei dem Kauf Ihrer nächsten Disketten die angesprochenen Kriterien und Sie werden das für Sie günstigste Angebot finden.

Rolli Disk Ein Diskettenkasten für Hängeregister

Rolli Disk ist der Name für einen neuentwickelten Diskettenkasten - für 5 1/4" Disketten - der genau in ein Hängeregister eines jeden Schreibtisch-Unterschrankes paßt. Dieses neue Disketten-Aufbewahrungssystem zeichnet sich besonders durch 3 praktische Vorzüge aus:

*** platzsparend**

- Keine platzraubende Diskettenbox auf dem Schreibtisch.

*** sicher**

- Mit dem Rolli Disk Diskettenkasten

werden Ihre Disketten diebstahlsicher im Schreibtisch-Unterschrank oder



auch in jedem anderen Hängeregister verwahrt.

*** griffbereit**

- Die Disketten sind immer griffbereit neben dem Computer.

Jede Box hat unterteilte Fächer und ist für bis zu 80 Disketten ausgelegt. Der Preis liegt bei 79,- DM incl. MWSt. und nach Auskunft der Firma erfolgt die Lieferung frei Haus. Nähere Auskünfte erhalten Sie bei:

Computer Links EDV-Zubehör, München 81

Misco Roll Top – platzsparende

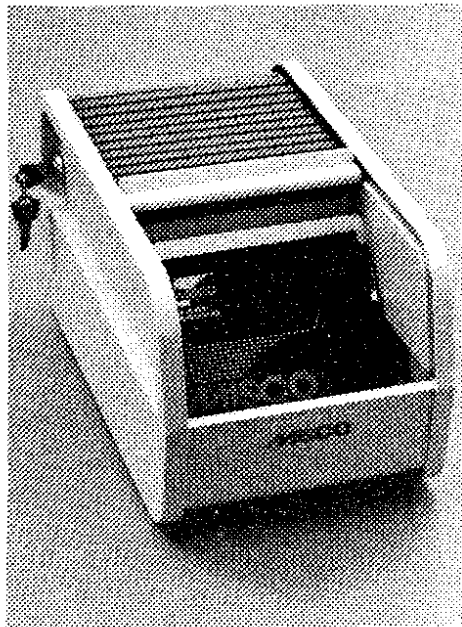
Archivieren leicht gemacht:

Mit dem Misco Roll Top, der platzsparenden Diskettenbox, lassen sich Disketten übersichtlich aufbewahren. Schutz vor Staub und Beschädigung gewährleistet der praktische Rolladen-Deckel im „Holz-Look“. Der rutschfeste Roll Top aus stabilem Kunststoff bietet – je nach Modell – Platz für 55 3 1/2“-, 120 5 1/4“- oder 100 8“-Disketten. Zum Schutz der Daten vor unbefugtem Zugriff liefert Misco die drei Versionen wahlweise auch abschließbar. Ein Plus in Sachen Übersicht sind die farbigen Registerstreifen.

Den Misco Roll Top gibt es im 3 1/2“-Format ab DM 79,-- (netto), für 5 1/4“-Disketten ab DM 129,-- (netto) und im 8“-Format für DM 149,-- (netto).

Misco
EDV-Zubehör GmbH
Nordendstraße 72-74
6082 Mörfelden-Walldorf

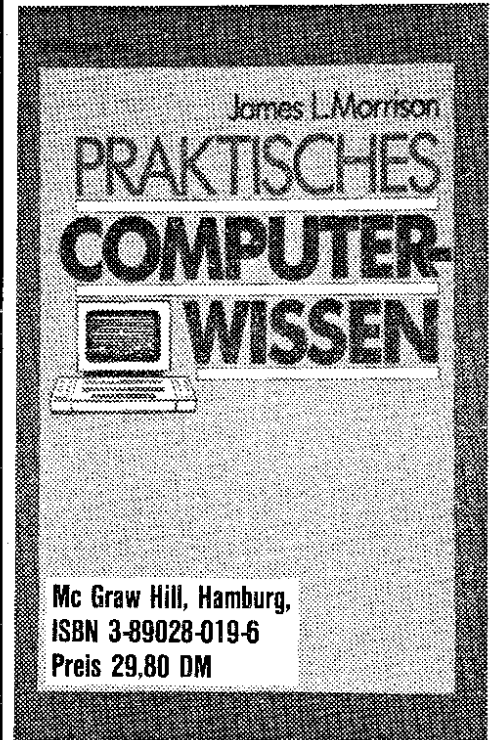
Diskettenbox



Praktisches Computerwissen

von James L. Morrison

Dieses Buch beschäftigt sich hauptsächlich mit dem Terminal und seinen Anwendungsmöglichkeiten im privaten und beruflichen Bereich. Die Ausführungen gelten für fast jedes Terminal und sollen dem Leser in seinem Bemühen unterstützen, sich mit der Benutzung von Terminals, und damit der breiten Palette von der Textverarbeitung bis hin zu allgemeinen Datenverarbeitungsprozessen, vertraut zu machen. Nach dem Durcharbeiten des



Glück für MARCOM-Fachhändler in Stade

In Stade, einem hübschen Städtchen zwischen Cuxhaven und Buxtehude an der Elbe, sorgt jetzt ein auffälliger Geländewagen im ELEPHANT-Look für Aufsehen: Es ist der im satten Gelb lackierte und mit dem Elefantenkopf des Markenzeichens ELEPHANT MEMORY SYSTEMS geschmückte Jeep, der während der Hannover-Messe '85 auf dem MARCOM-Messe-

stand in der CeBit-Halle als 1. Preis eines Fachhändler-Preisausschreibens bestaunt wurde.

Mitte September war es soweit: Als Gewinner wurde die Firma Streeck-Electronic-Shop ausgelost. Am 17.9.85 übergaben der Vertriebsleiter Klaus Heimers und der Productmanager Albrecht Niehof den gewonnenen Jeep den glücklichen neuen

Besitzern, Herrn Streeck und Herrn Noack, in ihrem Geschäftslokal.

Da im März der neue Termin der CeBit-Ausstellung ist, sind alle Besucher dieser Messe aufgefordert, ihr Glück bei den diesjährigen Preisausschreiben zu versuchen.



Textes sollten Sie sowohl Grundkenntnisse über den Gebrauch von Terminals als auch über die Bedienung von Tastaturen besitzen. Sie brauchen nur wenig oder keine Erfahrung im Umgang mit Computern zu haben, aber Sie sollten neugierig darauf sein, was man mit einem Terminal alles machen kann.

Das Buch will hauptsächlich die grundlegenden Fertigkeiten und Routinen beim Bedienen einer Tastatur vermitteln, damit der Benutzer diese versteht und in die Lage versetzt wird, leicht und problemlos damit umzugehen.


```

341 GOSUB304:POKEV+21,0:SYS4405:RETURN <1>
342 REM***** <28>
343 REM* SAVE LEVEL * <180>
344 REM***** <30>
345 OPEN2,8,2,"@:LEVEL"+RIGHT$("0"+MID$(
STR$(LL),2),2)+"",F,W":GOSUB200 <191>
346 IFA<>OTHEN348 <28>
347 FORI=0TO681:PRINT#2,CHR$(PEEK(49174+
I));:NEXT:PRINT#2,CHR$(0);CHR$(0) <156>
348 CLOSE1:CLOSE2:GOSUB196:RETURN <137>
349 REM***** <35>
350 REM* LOAD LEVEL * <111>
351 REM***** <37>
352 POKE252,22:POKE253,192:GOSUB353:GOSU
B200:GOTO354 <114>
353 CLOSE2:OPEN2,8,2,"LEVEL"+RIGHT$("0"+
MID$(STR$(LL),2),2)+"",F,R":RETURN <119>
354 IFA<>OTHEN356 <26>
355 GOSUB353:SYS4315 <103>
356 CLOSE1:CLOSE2:RETURN <205>
357 REM***** <43>
358 REM* DEMO PLAY * <166>
359 REM***** <45>
360 POKEBI,0:POKEHI,0:PRINT"<CLEAR>":POK
EHI,0:POKERA,11:Y=2:X=10:F=2 <172>
361 Z$="DEMO@MODE":GOSUB249:POKEBI,23:GO
SUB154 <189>
362 IFPEEK(2)=1THENFORI=1TO255STEP.5:POK
EV+39,I:NEXT <161>
363 POKE781,6:SYS59903:LETB,6,"<WHITE RV
SDN SPACE>ANOTHER<SPACE>TRY<SPACE>?<SPAC
E>(Y/N)<SPACE>":GOSUB332 <146>
364 IFA$="Y"THEN357 <206>
365 POKEV+21,0:GOTO288 <97>

```

Teil 4:

```

100 REM***** <211>
101 REM* ROBO-LEVEL MAKER * <27>
102 REM* (C)1985 BY MR-SOFT * <8>
103 REM* R O N A L D M A Y E R * <128>
104 REM* A-4062 THENING 24/AUSTRIA * <168>
105 REM***** <216>
106 POKES3281,0:POKES3280,0:PRINT"<CLEAR
DOWN4 RIGHTS PURPLE>THIS<SPACE>PROGRAM<
SPACE>IS<SPACE>CREATING" <199>
107 PRINT"<DOWN RIGHT?>S<SPACE>ROBO-LEVE
LS<SPACE>ON<SPACE>DISK." <221>
108 PRINT"<DOWN3 RIGHTS ORANGE>PLEASE<SP
ACE>WAIT<SPACE>ABOUT<SPACE>TWO<SPACE>MIN
UTES<DOWN2>" <86>
109 FORI=1TO5:PRINTTAB(10)"<CYAN>WRITING
<SPACE>LEVEL" I; <230>
110 OPEN2,8,2,"@:LEVEL"+RIGHT$("0"+MID$(
STR$(I),2),2)+"",F,W" <138>
111 X=0:READA:PRINT#2,CHR$(A);CHR$(I); <234>
112 READA#:A=ASC(LEFT$(A#,1)):IFA<63THEN
A=32:GOTO114 <20>
113 A#=MID$(A#,2) <62>
114 B=VAL(A#):FORJ=1TOB:PRINT#2,CHR$(A);
:X=X+1:NEXT:IFX<6BOTHEN112 <28>
115 PRINT#2,CHR$(0);CHR$(0):CLOSE2:PRINT
";OK":NEXT <76>
116 PRINT"<DOWN2 RIGHTS GREEN>THANK<SPAC
E>YOU!<SPACE>I'M<SPACE>READY...":END <60>
117 REM***** <228>
118 REM* LEVEL 01 * <142>
119 REM***** <230>
120 DATA34,41,?1,1,?1,1,?1,1,?1,C1,1,?1,
1,?1,1,?1,1,?1,C1,1,?1,1,?1,1,?1,1,?1 <145>
121 DATAC1,1,?1,1,?1,1,?1,1,?1,C1,5,@7,2
,@7,2,@7,2,@7,6,@1,D5,@1,1,C1,@1,D5,@1

```

```

122 DATA1,C1,@1,D5,@1,1,C1,@1,D5,@1,1,C1
,4,@1,D1,@3,D1,@1,2,@1,D1,@3,D1,@1,2,@1 <253>
123 DATAD1,@3,D1,@1,2,@1,D1,@3,D1,@1,6,@
1,D1,@1,1,@1,D1,@1,C1,1,@1,D1,@1,1,@1 <104>
124 DATAD1,@1,C1,1,@1,D1,@1,1,@1,D1,@1,C
1,1,@1,D1,@1,1,@1,D1,@1,C1,5,@1,D1,@1,1 <153>
125 DATA@1,D1,@1,2,@1,D1,@1,1,@1,D1,@1,2
,@1,D1,@1,1,@1,D1,@1,2,@1,D1,@1,1,@1,D1 <170>
126 DATA@1,6,@1,D1,@1,1,@1,D1,@1,1,C1,@1
,D1,@1,1,@1,D1,@1,1,C1,@1,D1,@1,1,@1,D1 <22>
127 DATA@1,1,C1,@1,D1,@1,1,@1,D1,@1,1,C1
,4,@1,D1,@3,@1,@2,1,@1,D1,@1,1,@1,D1,@2 <52>
128 DATA1,@1,D1,@3,D1,@2,1,@1,D1,@1,1,@1
,D1,@2,5,@1,?6,@1,1,@1,?1,@1,1,@1,?2,@1 <237>
129 DATA1,@1,?6,@1,1,@1,?1,@1,1,@1,?2,@1
,5,@1,D1,@3,D2,@1,1,@1,D1,@1,1,@1,D2,@1 <193>
130 DATA1,@1,D1,@3,D2,@1,1,@1,D1,@1,1,@1
,D2,@1,5,@1,D1,@1,1,@1,D2,@1,1,@1,D1,@1 <74>
131 DATA1,@1,D2,@1,1,@1,D1,@1,1,@1,D1,@1
,1,@1,D1,@1,1,@1,D2,@1,5,@1,D1,@1,1,@1 <71>
132 DATAD2,@1,F1,@1,D1,@3,D2,@1,F1,@1,D1
,@3,D2,@1,F1,@1,D1,@3,D2,@1,5,@1,D1,@1 <127>
133 DATA1,@1,D2,@1,F1,@1,D6,@1,F1,@1,D6,
@1,F1,@1,D6,@1,5,@3,1,@4,F1,@8,F1,@8,F1 <21>
134 DATA@8,13,F1,8,F1,8,F1,12,E38,2 <20>
135 REM***** <246>
136 REM* LEVEL 02 * <169>
137 REM***** <248>
138 DATA20,13,D3,C7,D3,2,C1,23,E6,?3,E6,
24,D5,C7,D5,1,C1,21,E6,5,E6,23,D5,7,D5 <32>
139 DATA2,C1,21,?3,7,E5,17,@1,B16,D6,B3,
F3,11,@1,15,E6,4,F3,11,@1,13,C3,D4,5,F3 <160>
140 DATA11,@1,11,E3,?1,E3,7,F3,11,@1,10,
D3,C3,D1,8,F3,11,@1,9,E5,?1,E1,9,F3,11 <43>
141 DATA@1,7,D6,C3,D7,2,F3,11,@1,6,E8,?1
,E8,2,F3,11,@1,6,D5,C7,D5,2,F3,11,@1,7 <17>
142 DATA?4,7,?4,3,F3,1,?3,3,D6,C4,D7,C5,
D7,C4,D5,2 <92>
143 REM***** <254>
144 REM* LEVEL 03 * <186>
145 REM***** <0>
146 DATA36,6,A30,@1,8,@1,30,@1,7,?1,@1,?
1,29,@1,7,@3,12,@1,4,A5,1,?4,2,@1,8,@1 <7>
147 DATA1,?3,1,?3,1,?2,2,?1,1,?3,5,D6,1,
@1,8,@1,1,@3,1,@3,1,@2,?1,1,@1,1,@3,4 <242>
148 DATAC1,D1,?4,D1,1,@1,8,@1,1,@1,3,@1,
1,@1,1,@1,1,@1,1,@1,7,D6,1,@1,8,@1 <176>
149 DATA1,@1,2,C1,@1,?1,@1,C1,@1,?1,@1,C
1,@1,C1,@1,?2,C1,2,C1,1,D1,6,@1,8,@1,1 <19>
150 DATA@1,3,@3,1,@1,?1,@1,1,@1,1,@3,5,D
1,6,@1,3,E27,88,D2,30,C1,1,D6,C1,6,?3,6 <36>
151 DATA?2,6,?3,6,B7,C1,4,E2,C3,E5,C4,E5
,C3,E5,C1,1,D1,4,D1,C1,32,B7,C1,31,C1,1 <53>
152 DATAD6,C1,17,C14,1,B7,C1,4,B12,C1,14
,C11,2 <29>
153 REM***** <8>
154 REM* LEVEL 04 * <205>
155 REM***** <10>
156 DATA20,1,D1,E1,D1,E3,5,E3,D1,E1,D1,3
,D1,E1,D1,E8,D1,8,D1,E1,D1,E4,3,E4,D1,E1 <30>
157 DATAD1,3,D1,E1,D1,E8,D1,E1,7,D1,E1,D
1,E5,1,E5,D1,E1,D1,B1,2,D1,E1,D1,E8,D1 <159>
158 DATAE1,D1,F2,4,D1,E1,D1,E11,D1,E1,D1
,1,B1,1,D1,E1,D1,E1,6,E1,D1,E1,D1,6,D1 <181>
159 DATAE1,D1,?3,E5,?3,D1,E1,D1,2,B1,D1,
E1,D1,E1,6,E1,D1,E1,D1,1,?1,4,D1,E1,D1 <31>
160 DATAE11,D1,E1,D1,1,A1,1,D1,E1,D1,E8,
D1,E1,D1,1,?1,4,D1,E1,D1,E2,2,E3,2,E2 <102>
161 DATAD1,E1,D1,A1,2,D1,E1,D1,E1,?6,E1,
D1,E1,7,D1,E1,D1,E1,4,E1,4,E1,D1,E1,D1 <196>
162 DATAB1,2,D1,E1,D1,E8,D1,3,?1,4,D1,E1
,D1,E1,4,E1,4,E1,D1,E1,D1,1,E1,1,D1,E1 <200>
163 DATAD1,E2,4,E2,D1,E1,2,?1,3,F1,D1,E1
,D1,E2,7,E2,D1,E1,D1,2,B1,D1,E1,D1,E1,6 <51>

```

programme

```

164 DATAE1,D1,E1,D1,2,F1,2,F1,D1,E1,D1,E
3,5,E3,D1,E1,D1,2,A1,D1,E1,D1,E1,6,E1 <224>
165 DATAD1,E1,D1,1,?1,F1,2,F1,D1,E1,D1,E
3,5,E3,D1,E1,D1,1,A1,1,D1,E1,D1,E2,4,E2 <106>
166 DATAD1,E1,D1,1,?1,F1,2,F1,D1,E1,D1,E
4,3,E4,D1,E1,D1,A1,2,D1,E1,D1,E2,4,E2 <153>
167 DATAD1,E1,D1,2,F1,2,F1,D1,E1,D1,ES,1
,ES,D1,E1,D1,3,D1,E1,D1,ES,2,E3,D1,E1 <139>
168 DATAD1,2,F1,2,F1,36,F1,2,F1,2,F1,11,
F1,1,C1,?2,C1,2,F1,B,F1,4,F1,2,F1,B13,4 <204>
169 DATAC2,5,A12,F1,2 <24>
170 REM***** <25>
171 REM* LEVEL 05 * <231>
172 REM***** <27>
173 DATA37,@3,B4,21,C2,2,?1,1,?1,5,@1,1,
@1,4,?1,7,?1,1,C2,11,F7,3,@1,1,@1,4,C1 <251>
174 DATA5,?1,1,C2,10,C1,2,F7,3,@1,1,@1,4

```

```

,?1,3,?1,1,C2,10,C2,3,F2,8,@1,?1,@1,4,C1 <115>
175 DATA1,?1,1,C2,3,C1,7,C1,1,?1,3,F2,?4
,4,@1,?1,@1,6,C2,12,C1,2,C1,@1,2,F6,?1 <182>
176 DATA3,@3,5,C1,7,C1,4,C2,1,?1,C1,1,@1
,2,F7,3,@3,13,?1,1,?1,1,?1,3,C1,2,@1,7 <15>
177 DATAF2,3,@1,1,@1,13,CS,1,?1,C1,3,@2,
1,F1,?4,F2,3,@1,1,@1,17,?1,1,C1,5,@1,1 <14>
178 DATAF7,3,@1,1,@1,17,C2,6,@1,1,F6,4,@
1,?1,@1,25,@1,?3,1,?2,5,@1,?1,@1,4,E1,1 <154>
179 DATAE3,1,E1,1,E3,3,E1,?1,E1,1,E3,@1,
E3,1,E2,?1,4,@3,4,E1,2,E1,2,E1,1,E1,5 <64>
180 DATAE1,?1,E1,1,E1,1,E1,1,E1,1,E1,1,E
1,1,E1,4,@1,1,@1,4,E1,2,E1,4,E3,3,E3,1 <113>
181 DATAE3,1,E3,1,E1,1,E1,4,@1,?1,@1,4,E
1,2,E1,6,E1,3,E1,1,E1,1,E1,1,E1,1,E2,2 <38>
182 DATAE1,1,E1,4,@3,4,E1,2,E1,4,E3,3,E1
,1,E1,1,E1,1,E1,1,E1,1,E1,1,E2,5 <220>

```

Extended Basic Befehlsenerweiterung

Der Anwendungsbereich des Toolkits umfaßt folgende vier Bereiche:

1. Die I/O Befehle für Diskette und Kassette
2. Die Befehle zum Debugging (Fehlerentfernung)
3. Befehle, die die Eingabe von Basicprogrammen vereinfachen
4. Befehle, die für die Strukturierung von Basicprogrammen sorgen

Anleitung zu den Toolkitbefehlen:

--- **DLOAD „Programmname“, Startadresse**
Der „DLOAD“ Befehl ersetzt auf komfortable Weise das übliche „LOAD Programmname“, 8(1)“ des Standardbasics.

„DLOAD“ nur mit Angabe eines Namens entspricht dem normalen „LOAD“ mit Angabe des Programmnamens und der Geräteadresse. Wird bei „DLOAD“ außer dem Namen noch ein Komma angegeben, wird das Programm als Maschinenprogramm wie bei „LOAD“ mit Angabe der Geräteadresse und einer Sekundäradresse geladen. Sollte hinter dem Komma noch eine Adresse angegeben worden sein, wird das Programm ab dieser Adresse in den Speicher geladen. Im normalen v2.0 Basic existiert hierzu keine Parallele.

--- **DVERIFY „Programmname“**
„DVERIFY“ vergleicht ein Programm im Speicher mit einem Programm auf einer Diskette.

--- **DMERGE „Programmname“**
Mit „DMERGE“ ist es endlich möglich, ein Programm von Diskette an ein Programm im Speicher anzuhängen.
Bei „DMERGE“ wird genauso wie bei

„DLOAD“ und „DVERIFY“ die Start- und Endadresse während bzw. nach dem Laden ausgegeben.

--- **DSAVE „Programmname“, Startadresse, Endadresse**
Der „DSAVE“ Befehl dient, falls nur der Programmname angegeben wurde, zum Abspeichern eines Basicprogramms. Sollten allerdings noch weitere Parameter mit angegeben worden sein, wird das Programm von der Startadresse bis zur Endadresse minus eins als Maschinenprogramm abgespeichert.

--- **CATALOG („\$:Parameter für Selektierung“)**
Das Anzeigen eines Directories ist mit dem „CATALOG“-Befehl möglich. Das Auslisten kann mit Shift oder Shift-Lock gehalten, mit Ctrl verlangsamt oder mit Stop abgebrochen werden.
Nach Angabe eines zusätzlichen Strings ist eine gezielte Ausgabe der Catalogeinträge möglich. Auskunft hierüber gibt das Manual zur 1541.

--- **DISK**
Der „DISK“-Befehl kann auf drei verschiedene Arten verwendet werden. Werden

keine weiteren Parameter angegeben, wird der Statusreport des Floppy's ausgegeben.

--- **DISK „Anweisung“**
„DISK“ mit Angabe eines Befehlsstrings sendet einen Befehl ans Floppy.

--- **DISK Geräteadresse**
Mit dieser Variation des „DISK“ Befehls ist eine Umschaltung der Geräteadresse (8-11), auf die sich alle Diskettenoperationen beziehen, möglich.

--- **MERGE „Programmname“, Geräteadresse**
Dieser Befehl hat dieselbe Funktion wie „DMERGE“, nur, daß er das Format des normalen „LOAD“ Befehl aufweist und somit auch für den Recorder zuständig ist.

--- **KEY Nr. „String“**
Der „KEY“-Befehl übernimmt die gesamte Programmierung der 16 (!) Funktionstasten, von denen jede mit einem 40 Zeichen langen String belegt werden kann. Die Belegung nach dem Programmstart oder einem Reset entnehmen Sie bitte Tabelle 2.
Die erste Anwendung des „KEY“-Befehls

dient zum Belegen der Funktionstaste „Nr.“ mit dem Text „String“.

--- KEY

„KEY“ ohne Parameter listet alle 16 Funktionstastenbelegungen aus.

--- KEY ON/OFF

Durch „KEY ON/OFF“ werden die Funktionstasten wahlweise an- oder ausgeschaltet.

--- DEL Start - Ende

„DEL“ löscht alle Basiczeilen im Bereich von „Start“ bis „Ende“. Die Parameterübergabe entspricht dem „LIST“ Befehl.

--- FIND Modus, Suchtext

Mit „FIND“ können Sie sich alle Zeilennummern ausgegeben lassen (Modus = 0) oder als Zeilen listen lassen (Modus = 1), in denen der Suchtext vorkommt.

--- AUTO Schrittweite

„AUTO“ mit Angabe einer Zahl schaltet die automatische Zeilennummernvorgabe mit der angegebenen Schrittweite ein.

--- AUTO OFF

„AUTO OFF“ schaltet die Zeilennummernvorgabe wieder ab.

--- OLD

Der „OLD“ Befehl rekonstruiert ein mit „NEW“ oder Reset gelöscht Basicprogramm, wobei alle Basiczeiger korrigiert werden.

--- CLS

Mit „CLS“ wird der Textbildschirm gelöscht.

--- VRES

Durch „VRES“ wird der Videocontroller neu initialisiert.

--- SRES

Durch „SRES“ wird der SID auf Null gesetzt.

--- TRACE x,y, Color, Modus

Der „TRACE“ Befehl dient zur komfortablen Fehlersuche und Programmverfol-

gung. Es werden jeweils die letzten sechs Zeilennummern während des Programmablaufs an der Stelle (x,y) mit der Farbe (Color) ausgegeben.

Über den Modus (0/1) kann bestimmt werden, ob eine Zeile über dem Tracefenster gelöscht werden soll oder nicht. Dies ist beim Aufwärtsscrollen sinnvoll.

--- TRACE

Nur „TRACE“ zeigt das letzte Tracefenster an.

--- TRACE OFF

Hiermit wird der Tracemodus abgeschaltet.

--- WALK ON/OFF

„WALK ON/OFF“ aktiviert bzw. deaktiviert den Einzelschrittmodus. Unter einem Einzelschrittmodus versteht man schrittweise Abarbeitung von (Basic)befehlen. Nach einem Befehl wird auf die CTRL-Taste gewartet.

Ein Abarbeiten mit normaler Geschwindigkeit ist durch folgende Tastenkombination im Einzelschrittmodus zu erreichen: Shiftlock/CTRL/CTRL.

Nach Loslassen der Shift(lock)-Taste wird das Programm wieder im Einzelschrittmodus abgearbeitet.

--- HELP

Nach dem Eingeben von „HELP“ oder dem Drücken von F7 wird der fehlerhafte Teil einer Zeile angezeigt (nur nach Programmabbruch durch einen Fehler).

--- ON ERROR GOTO Zeilennummer

Das Abfangen von Fehlern in einem Basicprogramm wird durch diesen Befehl ermöglicht. Sollte während des Programmablaufs ein Fehler auftreten, wird zu der in „ON ERROR GOTO“ angegebenen Zeilennummer gesprungen und das dort stehende Programm zur Fehlerauswertung abgearbeitet.

Achtung: In der Auswertungsroutine darf kein weiterer Fehler auftreten, da das Programm sonst in einer Endlosschleife bleiben würde.

--- ERROR OFF

Durch „ERROR OFF“ wird die Fehlerbehandlung wieder abgeschaltet.

--- RESUME

„RESUME“ dient zum Verlassen der Fehlerauswertungsroutine. Es wird zum nächsten hinter dem Fehler stehenden Doppelpunkt gesprungen.

--- ERNR/ERLIN

Diese beiden Funktionen dienen zum Erkennen des aufgetretenen Fehlers. Mit „ERNR“ kann die Fehlernummer und mit „ERLIN“ die Zeilennummer, in welcher der Fehler aufgetreten ist, ermittelt werden. Die Fehlermeldungen und die ihnen zugeordneten Fehlernummern entnehmen Sie bitte Tabelle 1.

--- LBL „String“

Mit „LBL“ wird eine Zeile symbolisch bezeichnet. d. h., außer durch die Zeilennummer, kann die Zeile auch noch durch die hinter „LBL“ stehende symbolische Bezeichnung genau bestimmt werden.

--- GOTO „String“

„GOTO“ mit Angabe eines Strings hat dieselbe Funktion, wie der normale „GOTO“ Befehl, nur daß zum nächsten folgenden Befehl, hinter dem mit „LBL“ definierten symbolischen Sprungziel gesprungen wird.

--- GOSUB „String“

„GOSUB“ entspricht dem „GOTO“ Befehl, nur daß er zum Aufruf eines mit „RETURN“ abgeschlossenen Unterprogrammes dient.

--- RESTORE Zeilennummer

Durch die Angabe einer Zeilennummer hinter „RESTORE“ wird der DATA-Zeiger nicht auf den Programmanfang, sondern auf das erste DATA-Element einer bestimmten Zeile gesetzt.

--- RESTORE „String“

Diese Anwendungsart hat dieselbe Aufgabe wie Restore Zeilennummer, nur daß die Zeile symbolisch angegeben wird.

programme

--- ELSE Zeilennummer oder Label oder „Programmtext“

Durch den „ELSE“-Befehl wird der normale „IF . . . THEN . . .“ Befehl wesentlich erweitert. Falls die hinter „IF“ angegebene Bedingung unwahr sein sollte, wird zum nach „ELSE“ angegebenen Sprungziel gesprungen oder der hinter dem Doppelpunkt folgende Programmtext abgearbeitet.

Nachdem der Basiclader eingegeben worden ist, sollte man ihn aus Sicherheitsgründen abspeichern. Nach dem Starten wird in der obersten Zeile die momentane DATA-Zeilennummer, das momentane Hexbyte und der Status ausgegeben. Sollte beim Einlesen ein Fehler auftreten, korrigieren Sie bitte die gesamte Zeile, bis die DATA's fehlerfrei eingelesen werden.

Sind alle DATA's eingegeben, verlangt das Programm die Adresse des Gerätes, auf dem das fertige Programm gespeichert werden soll.

Das Toolkit wird unter dem Namen „EXTENDED-BASIC“ abgelegt. Um das Toolkit zu aktivieren ist dieses Programm zu laden und mit „RUN“ zu starten.

Tabelle 1:

01 = too many files	02 = file open
03 = file not open	04 = file not found
05 = device not present	06 = not input file
07 = not output file	08 = missing filename
09 = illegal device number	10 = next without for
11 = syntax error	12 = return without gosub
13 = out of data	14 = illegal quantity
15 = overflow	16 = out of memory
17 = undef'd statement	18 = bad subscript
19 = redim'd array	20 = division by zero
21 = illegal direct	22 = type mismatch
23 = string too long	24 = formula too complex
25 = can't continue	26 = undef'd function
27 = verify	28 = load

Tabelle 2:

01 = „catalog“+chr\$(13)
02 = „disk“
03 = „list“+chr\$(13)
04 = „run“+chr\$(13)
05 = „dload“+chr\$(34)
06 = „dsave“+chr\$(34)
07 = „help“+chr\$(13)
08 = „find“
09 = „trace“
10 = „walk“
11 = „auto10“+chr\$(13)
12 = „dload“+(18*Cursor right)+”:“chr\$(13)
13 = „dverify“+chr\$(34)
14 = „dmerge“+chr\$(34)
15 = „del“
16 = „key“

```

10 REM ***** <121>
20 REM * --- EXTENDED BASIC V 2.0 --- * <20>
30 REM * * <127>
40 REM * WRITTEN AND COPYRIGHT 1985 * <95>
50 REM * * <147>
60 REM * BY DIETER VUELLERS * <248>
70 REM * * <167>
80 REM * 4780 LIPPSTADT 14 * <25>
90 REM * * <187>
95 REM ***** <206>
118 AD=11*4096:PRINT"⟨CLEAR⟩"; <59>
120 X=0:FS=0 <140>
130 READ H$:IFX<>THEN133 <109>
131 ZN=PEEK(63)+PEEK(64)*256 <133>
132 PRINT"⟨HOME⟩ ZEILE⟨SPACE⟩: "ZN"⟨SPACE⟩ <241>
BYE⟨SPACE⟩: $00⟨SPACE⟩STATUS⟨SPACE⟩:⟨SPA <13>
CE⟩OK⟨SPACE⟩! <84>
133 PRINT"⟨HOME⟩"TAB(21)H$ <27>
140 IF H$="-1" THEN 250 <219>
150 GOSUB 400 <27>
160 POKE AD,DC <133>
170 PS=PS+DC <62>
180 AD=AD+1 <31>
190 X=X+1 <124>
200 IF X<17 THEN 130 <85>
210 READ H$

```

```

220 GOSUB 310 <34>
230 IF DC=PS THEN 120 <199>
240 PRINT"⟨HOME⟩"TAB(33)"ERROR⟨SPACE⟩!": <35>
END
250 PRINT"PROGRAMM⟨SPACE⟩ IST⟨SPACE⟩FEHLE <241>
RFREI⟨SPACE⟩!" <183>
260 INPUT"GERAETE⟨SPACE⟩ADRESSE";DV <76>
270 POKE2,DV <223>
280 SYS 49933 <163>
290 END
300 : <103>
310 DC=0 <215>
320 HE$=H$:H$=RIGHT$(H$,2) <37>
330 GOSUB 400 <144>
340 D1=DC:H$=LEFT$(HE$,2) <244>
350 GOSUB 400 <164>
360 DC=D1+DC*256 <13>
370 RETURN <1>
380 : <183>
390 : <193>
400 L=ASC(H$):R=ASC(RIGHT$(H$,1)) <20>
410 IFL>64THENL=L-55:GOTO430 <52>
420 IFL<58THENL=L-48 <145>
430 IFR>64THENR=R-55:GOTO450 <212>
440 IFR<58THENR=R-48 <17>
450 DC=L*16+R <1>

```

460 RETURN	<91>	A0,81,8D,0A,03,8C,0B,064F	<18>
1000 DATA 00,23,0B,C1,07,9E,20,2B,36,3B,38,34,29,20,45,5B,54,03ED	<189>	1370 DATA 03,60,4F,46,C6,45,52,52,4F,D2,45,4C,53,C5,4C,42,CC,06CB	<200>
1010 DATA 80,45,44,AB,42,41,53,49,43,20,56,32,20,2B,44,56,29,04C9	<47>	1380 DATA 41,D4,4D,45,52,47,C5,44,49,53,CB,43,41,54,41,4C,4F,0664	<251>
1020 DATA 00,00,00,E1,82,3E,83,C3,C2,CD,3B,30,11,20,20,20,2A,0579	<67>	1390 DATA C7,4B,45,D9,44,45,CC,46,49,4E,C4,41,55,54,CF,4F,4C,077A	<0>
1030 DATA 2A,2A,2A,20,43,42,4D,20,45,5B,54,45,4E,44,44,20,0401	<238>	1400 DATA C4,54,52,41,43,C5,57,41,4C,CB,52,45,53,55,4D,C5,4B,06FB	<113>
1040 DATA 20,42,41,53,49,43,20,56,32,2E,30,20,2A,2A,2A,2A,20,0370	<134>	1410 DATA 45,4C,D0,43,4C,D3,56,52,45,D3,53,52,45,D3,45,52,4E,0725	<2>
1050 DATA 20,20,11,20,2B,43,29,20,31,39,3B,35,20,42,59,20,44,031B	<90>	1420 DATA D2,45,52,4C,49,CE,00,07,AF,80,8F,3A,A9,B4,90,07,AF,079E	<59>
1060 DATA 49,45,54,45,52,20,56,55,45,4C,4C,45,52,53,20,20,4C,0497	<100>	1430 DATA 7A,8C,D4,83,54,84,5F,85,81,8B,5C,89,64,8A,DA,8C,37,0892	<208>
1070 DATA 49,50,50,53,54,41,44,54,20,2D,2D,2D,2D,2D,2D,03F1	<243>	1440 DATA 8D,FE,8E,C4,8F,95,90,43,E5,1B,92,31,92,00,A9,92,AF,0913	<121>
1080 DATA 2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,02FD	<53>	1450 DATA 92,00,00,06,0C,00,00,01,0B,A2,FF,7B,9A,DB,A2,05,8E,056D	<49>
1090 DATA 2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,2D,00,A6,0349	<53>	1460 DATA 16,D0,20,AC,49,CE,20,50,FD,20,15,FD,20,5B,FF,5B,20,53,078A	<65>
1100 DATA 7A,A0,04,84,0F,8D,00,02,10,07,C9,FF,F0,3E,EB,D0,F4,0829	<103>	1470 DATA E4,20,BF,E3,A2,FB,9A,AD,DA,82,8D,86,02,A2,00,8D,09,0963	<84>
1110 DATA C9,20,F0,37,85,0B,C9,22,F0,55,24,0F,70,2D,C9,3F,D0,0775	<51>	1480 DATA 80,F0,09,20,D2,FF,EB,D0,F5,4C,72,FE,20,C1,87,AD,DF,0AC7	<38>
1120 DATA 04,A9,99,D0,25,C9,30,90,04,C9,3C,90,1D,84,71,A0,00,070F	<133>	1490 DATA 82,AC,E0,82,85,2B,84,2C,A2,00,A0,80,86,33,86,35,86,07AC	<43>
1130 DATA 84,0B,8B,86,7A,CA,CB,EB,8D,00,02,3B,F9,9E,A0,F0,F5,09A4	<205>	1500 DATA 37,84,34,84,36,84,3B,1B,20,25,FE,20,44,A6,4C,4E,83,05E7	<207>
1140 DATA C9,80,D0,2F,05,0B,A4,71,EB,CB,99,FB,01,C9,00,F0,3B,08A3	<121>	1510 DATA 4B,8A,4B,9B,4B,20,BC,F6,20,E1,FF,D0,C9,20,5B,FF,20,08FF	<144>
1150 DATA 3B,E9,3A,F0,04,C9,49,D0,02,85,0F,3B,E9,55,D0,A0,85,0B32	<17>	1520 DATA 5E,FF,20,29,82,AD,DA,82,8D,86,02,AD,DB,82,8D,20,D0,08CD	<103>
1160 DATA 0B,8D,00,02,F0,E0,C5,0B,F0,DC,CB,99,FB,01,EB,D0,F0,0A35	<11>	1530 DATA AD,DC,82,8D,21,D0,A9,80,8D,8A,02,A9,0B,8D,DD,82,A9,0911	<233>
1170 DATA A6,7A,E6,0B,CB,B9,9D,A0,10,FA,B9,9E,A0,D0,B5,F0,0F,0A54	<55>	1540 DATA 00,8D,DE,82,8D,34,8D,85,CF,8D,5B,8A,8D,59,8A,A9,E1,08FB	<239>
1180 DATA 8D,00,02,10,8D,99,FD,01,C6,7B,A9,FF,85,7A,60,A0,00,080B	<148>	1550 DATA A0,82,8D,00,80,BC,01,80,A9,3E,A0,83,8D,02,80,8D,1B,06FA	<98>
1190 DATA B9,52,82,D0,02,CB,EB,8D,00,02,3B,F9,52,82,F0,F5,C9,0981	<197>	1560 DATA 03,8C,03,80,8C,19,03,A9,4B,A0,83,8D,16,03,8C,17,03,051D	<189>
1200 DATA 80,D0,04,05,0B,D0,99,A6,7A,E6,0B,CB,B9,51,82,10,FA,0B3C	<77>	1570 DATA A2,05,8D,0F,FD,9D,03,80,CA,D0,F7,A9,CA,A0,85,8D,14,095A	<186>
1210 DATA B9,52,82,D0,E0,F0,C6,0B,4B,8A,4B,9B,4B,AD,8D,02,C9,08FA	<138>	1580 DATA 03,8C,15,03,A9,01,85,CC,A9,82,A0,90,8D,00,03,8C,01,061A	<187>
1220 DATA 01,F0,F9,6B,8A,6B,AA,AD,7C,90,F0,1E,AD,81,90,C5,60,09B6	<75>	1590 DATA 03,A9,8D,A0,8A,8D,02,03,8C,03,03,6C,02,A0,60,20,79,058E	<115>
1230 DATA F0,02,80,15,1B,9B,65,5F,CD,80,90,F0,02,90,0A,A9,12,074F	<116>	1600 DATA 00,90,65,D0,23,20,E2,83,4C,4E,84,AD,DD,82,85,8A,20,07F6	<73>
1240 DATA 20,D2,FF,A9,00,8D,7C,90,6B,2B,10,0F,24,0F,30,0B,C9,0619	<28>	1610 DATA B4,FF,A9,6F,85,B9,20,96,FF,20,A5,FF,20,D2,FF,C9,0D,0A49	<130>
1250 DATA FF,F0,07,C9,CC,80,06,4C,24,A7,4C,F3,A6,3B,E9,CB,AA,09D3	<244>	1620 DATA D0,F6,20,AB,FF,60,20,05,84,4C,4E,84,1B,20,9E,AD,24,075E	<70>
1260 DATA 84,49,A0,FF,CA,FO,0B,CB,B9,52,82,10,FA,30,F5,CB,B9,0A33	<190>	1630 DATA 0D,30,03,4C,0B,AF,20,A3,B6,C9,29,AA,90,03,4C,71,A5,064D	<9>
1270 DATA 52,82,30,05,20,47,AB,D0,F5,4C,EF,A6,20,73,00,A5,9D,0796	<174>	1640 DATA 20,79,00,AD,DD,82,85,8A,20,B1,FF,A9,6F,85,B9,20,93,08BD	<253>
1280 DATA 30,14,A5,39,A4,3A,8D,7E,90,8C,7F,90,A5,7A,A4,7B,8D,0801	<229>	1650 DATA FF,A0,00,B1,22,20,AB,FF,CB,CA,D0,F7,20,AE,FF,60,4C,0A0B	<73>
1290 DATA 80,90,8C,81,90,20,79,00,20,D4,91,20,D9,81,4C,AE,A7,07E6	<1>	1660 DATA 4B,82,20,9E,B7,E0,0B,90,F6,E0,0C,80,F2,8E,DD,82,60,09BB	<203>
1300 DATA C9,CC,90,04,C9,E0,90,06,20,79,00,4C,ED,A7,3B,E9,CC,08CE	<161>	1670 DATA 80,01,60,4C,D1,E1,60,20,5D,84,90,FA,4C,D1,E1,20,79,0891	<227>
1310 DATA 0A,AA,8D,AD,82,4B,8D,AC,82,4B,4C,73,00,A9,00,85,0D,0715	<254>	1680 DATA 00,F0,1E,1B,20,9E,AD,24,0D,30,03,4C,99,AD,20,A6,86,0603	<9>
1320 DATA 20,73,00,C9,E0,90,04,C9,E2,90,06,20,79,00,4C,8D,AE,0731	<79>	1690 DATA 85,B7,20,79,00,A5,22,A4,23,85,BB,84,BC,3B,80,10,A9,0784	<242>
1330 DATA 3B,E9,E0,AE,8A,20,73,00,20,F1,AE,6B,AB,B9,D5,82,85,084A	<213>	1700 DATA 24,85,02,A9,02,A0,00,85,BB,84,BC,A9,01,85,B7,AD,DD,07E6	<211>
1340 DATA 55,B9,D6,82,85,56,20,54,00,4C,8D,AD,A9,84,A0,80,8D,0815	<243>	1710 DATA 82,85,8A,A9,60,85,B9,20,D5,F3,90,03,4C,D1,E1,A5,BA,09E0	<2>
1350 DATA 04,03,8C,05,03,A9,47,A0,81,8D,06,03,BC,07,03,A9,AF,0530	<98>	1720 DATA 20,B4,FF,AS,B9,20,96,FF,A9,00,85,90,A0,03,84,FB,20,0BE6	<155>
1360 DATA A0,8D,8D,0B,03,8C,09,03,A9,F7,		1730 DATA A5,FF,85,FC,A4,90,D0,3C,20,A5,FF,A4,90,D0,35,A4,FB,0B01	<149>

programme

1740 DATA 8B, D0, E9, A6, FC, 20, CD, BD, A9, 20, 20, 16, E7, 20, A5, FF, A6, 09DD	<62>	2110 DATA 02, 85, 01, A9, 0D, 20, D2, FF, 1B, A5, F7, 69, 2B, 85, F7, A9, 00, 0799	<250>
1750 DATA 90, D0, 1F, AA, FO, 06, 20, 16, E7, 3B, B0, FO, A9, 0D, 20, 16, E7, 07E7	<121>	2120 DATA 65, F8, B5, F8, E6, F9, A5, F9, C9, 11, FO, 03, 4C, BA, 86, A5, 02, 0A57	<151>
1760 DATA AD, 8D, 02, C9, 01, FO, F9, A5, 91, C9, 7F, FO, 04, A0, 02, D0, B9, 098C	<177>	2130 DATA 85, 01, 60, AA, 9B, 4B, BA, 4B, C0, 00, FO, 10, A5, FB, FO, 07, A9, 0B42	<9>
1770 DATA 20, 42, F6, 60, 20, 79, 00, D0, 03, 4C, 1D, AB, 20, C3, 90, 90, 0B, 0643	<236>	2140 DATA 22, 20, D2, FF, C6, FB, A9, 2B, 20, D2, FF, A5, 02, 85, 01, A2, 00, 0868	<115>
1780 DATA A5, 5F, A4, 60, 85, 41, B4, 42, 4C, 37, 85, 20, BA, AD, 20, F7, B7, 07C1	<17>	2150 DATA BD, BB, B7, FO, 06, 20, D2, FF, EB, D0, F5, 6B, AA, A9, 00, 20, CD, 0A3B	<136>
1790 DATA A5, 15, C9, FA, D0, 03, 4C, 4B, B2, 20, 13, A6, B0, 03, 4C, E3, AB, 07F9	<15>	2160 DATA BD, A9, 29, 20, D2, FF, A9, F6, B5, 01, 6B, AB, 4C, 1D, B7, 4B, 45, 0835	<89>
1800 DATA A5, 5F, E9, 01, 85, 41, A5, 60, E9, 00, 85, 42, A5, 14, A4, 15, 85, 0760	<23>	2170 DATA 59, 00, 90, 05, 1C, 9F, 9C, 1E, 1F, 9E, B1, 95, 96, 97, 9B, 99, 9A, 072E	<174>
1810 DATA 3F, B4, 40, 60, 20, 73, 00, A9, 31, A0, EA, 7B, 8D, 14, 03, 8C, 15, 0617	<90>	2180 DATA 9B, 12, 92, 13, 93, 11, 1D, 91, 9D, 94, 43, 4B, 52, 24, 2B, 00, A9, 05A7	<48>
1820 DATA 03, 5B, 60, 20, 73, 00, A9, CA, A0, 85, 7B, 8D, 14, 03, 8C, 15, 03, 05A6	<135>	2190 DATA 00, A0, A0, 85, F7, B4, FB, A9, 04, A0, 8B, 85, F9, 84, FA, A2, 01, 09AC	<66>
1830 DATA 5B, 60, 20, 79, 00, D0, 03, 4C, AA, B6, C9, CC, FO, D4, C9, 91, FO, 0943	<133>	2200 DATA A0, 00, B1, F9, 91, F7, FO, 06, CB, C0, 29, D0, F5, 8B, CB, 84, 02, 0A14	<77>
1840 DATA E0, 20, 9E, B7, E0, 11, 90, 03, 4C, 4B, B2, E0, 00, FO, F9, A9, DB, 0969	<122>	2210 DATA 1B, A5, F9, 65, 02, 85, F9, A9, 00, 65, FA, 85, FA, 1B, A5, F7, 69, 093F	<123>
1850 DATA A0, 9F, 85, F7, 84, FB, 1B, A5, F7, 69, 2B, 85, F7, A9, 00, 65, FB, 09FE	<206>	2220 DATA 2B, 85, F7, A9, 00, 65, FB, 85, FB, EB, E0, 11, D0, D0, 60, 43, 41, 0984	<22B>
1860 DATA 85, FB, CA, D0, FO, 20, FD, AE, 1B, 20, 9E, AD, 24, 0D, 30, 03, 4C, 0805	<42>	2230 DATA 54, 41, 4C, 4F, 47, 0D, 00, 44, 49, 53, 4B, 00, 4C, 49, 53, 54, 0D, 03FB	<174>
1870 DATA 99, AD, 20, A3, B6, C9, 29, 90, 03, 4C, 71, A5, AB, AA, B1, 22, 91, 085C	<123>	2240 DATA 00, 52, 55, 4E, 0D, 00, 44, 4C, 4F, 41, 44, 22, 00, 44, 53, 41, 56, 03B6	<246>
1880 DATA F7, 8B, C0, FF, D0, F7, 8A, AB, C0, 2B, B0, 04, A9, 00, 91, F7, 20, 0A24	<45>	2250 DATA 45, 22, 00, 4B, 45, 4C, 50, 0D, 00, 46, 49, 4E, 44, 00, 54, 52, 41, 03A5	<203>
1890 DATA 79, 00, 60, 00, 00, 00, AD, C7, 85, FO, 03, 4C, 61, 86, A6, C5, E0, 0743	<120>	2260 DATA 43, 45, 00, 57, 41, 4C, 4B, 00, 41, 55, 54, 4F, 31, 30, 0D, 00, 44, 03A2	<207>
1900 DATA 07, 90, 0E, AD, 8D, 02, BD, CB, 85, A5, C5, BD, C9, 85, 4C, 31, EA, 0867	<142>	2270 DATA 4C, 4F, 41, 44, 1D, 1D, 1D, 1D, 1D, 1D, 1D, 1D, 1D, 1D, 1D, 1D, 0299	<222>
1910 DATA E0, 03, 90, EE, AD, 8D, 02, CD, CB, 85, D0, 09, A5, C5, CD, C9, 85, 0A15	<103>	2280 DATA 1D, 1D, 1D, 1D, 1D, 3A, 0D, 00, 44, 56, 45, 52, 49, 46, 59, 22, 00, 0313	<68>
1920 DATA D0, 02, FO, DD, A9, 00, A0, A0, 85, A7, 84, AB, E0, 04, FO, 22, E0, 09B6	<49>	2290 DATA 44, 4D, 45, 52, 47, 45, 22, 00, 44, 45, 4C, 00, 4B, 45, 59, 00, 4C, 03E0	<174>
1930 DATA 05, D0, 02, A2, 02, E0, 06, D0, 02, A2, 04, E0, 03, D0, 02, A2, 06, 0636	<136>	2300 DATA 0B, AF, 4C, 4B, B2, 20, 4A, B9, 20, 79, 00, D0, 03, 4C, 42, A6, 90, 0620	<173>
1940 DATA 1B, A5, A7, 69, 2B, 85, A7, A9, 00, 65, AB, 85, AB, CA, D0, FO, AE, 093C	<245>	2310 DATA 06, FO, 04, C9, AB, D0, E7, 20, 6B, A9, A5, 14, 85, F7, A5, 15, 85, 08CD	<252>
1950 DATA 8D, 02, E0, 01, D0, 0D, 1B, A5, A7, 69, 2B, 85, A7, A9, 00, 65, AB, 0724	<220>	2320 DATA FB, 20, 79, 00, FO, 27, C9, AB, D0, D3, 20, 73, 00, 20, 6B, A9, A5, 082B	<80>
1960 DATA 85, AB, E0, 02, D0, 06, 20, 9C, B6, 20, 9C, 86, E0, 04, D0, 0D, 1B, 0742	<46>	2330 DATA 14, 05, 15, C9, 00, D0, 0B, A9, FF, 85, 14, A9, F9, 85, 15, 1B, A5, 0709	<194>
1970 DATA A5, A7, 69, 68, 85, A7, A9, 01, 65, AB, 85, AB, A9, 04, 8D, C7, 85, 08B3	<77>	2340 DATA 14, 69, 01, 85, F9, A9, 00, 65, 15, 85, FA, 20, 59, A6, 20, 33, A5, 06B5	<247>
1980 DATA A5, 01, 85, A9, A9, F6, B5, 01, A0, 00, B1, A7, FO, 23, 99, 77, 02, 0816	<254>	2350 DATA A5, F7, A4, FB, 85, 14, 84, 15, 20, 13, A6, A5, 5F, A4, 60, 85, F7, 08C7	<55>
1990 DATA CB, C0, 0A, D0, F4, 84, C6, CE, C7, 85, FO, 14, 1B, A5, A7, 69, 0A, 0995	<175>	2360 DATA B4, FB, A5, F9, A4, FA, 85, 14, 84, 15, 20, 13, A6, A5, 5F, A4, 60, 08CB	<182>
2000 DATA 85, A7, A9, 00, 65, AB, 85, AB, A5, A9, 85, 01, 4C, DB, 85, 84, C6, 08D6	<196>	2370 DATA 85, F9, B4, FA, A0, 00, 1B, A5, 2D, 69, 02, 85, 2D, A9, 00, 65, 2E, 06DF	<158>
2010 DATA A9, 00, BD, C7, 85, A5, A9, 85, 01, 4C, DB, 85, 1B, A5, A7, 69, A0, 086C	<60>	2380 DATA 85, 2E, A5, FA, C5, 2E, D0, 06, A5, F9, C5, 2D, FO, 13, B1, F9, 91, 09E9	<203>
2020 DATA 85, A7, A9, 00, 65, AB, 85, AB, 60, A9, 00, A0, A0, 85, F7, 84, FB, 0950	<199>	2390 DATA F7, E6, F7, D0, 02, E6, FB, E6, F9, D0, 02, E6, FA, 4C, 07, 89, 3B, 0B29	<169>
2030 DATA A9, 01, 85, F9, A9, 00, 85, FA, A9, 9E, A0, 87, 20, 1E, AB, A6, F9, 0946	<247>	2400 DATA A5, 2D, E9, 02, E0, 02, C6, 2E, 20, 59, A6, 20, 33, A5, A9, 00, 4C, 066F	<113>
2040 DATA E0, 0A, B0, 05, A9, 30, 20, D2, FF, A9, 00, 20, CD, BD, A9, 2C, 20, 07B1	<132>	2410 DATA DB, 8C, 44, 49, 52, 45, 43, 54, 20, 4D, 4F, 44, 45, 20, 4F, 4E, 4C, 0570	<71>
2050 DATA D2, FF, A9, 00, 85, FB, A5, 01, 85, 02, A9, F6, 85, 01, A0, 00, B1, 089D	<229>	2420 DATA D9, A5, 9D, 30, 0D, A9, 3A, 85, 22, A0, B9, B4, 23, 6B, 68, 4C, 47, 0715	<20>
2060 DATA F7, FO, 3A, A2, 00, DD, A2, 87, FO, 15, EB, E0, 1A, D0, F6, C9, 22, 0A61	<80>	2430 DATA A4, 60, 00, 20, 4A, B9, 20, 9E, B7, E0, 02, 90, 03, 4C, 4B, B2, 8E, 06B5	<193>
2070 DATA FO, 67, C9, 20, 90, 63, C9, 9F, B0, 04, C9, 7F, B0, 5B, 4B, A5, FB, 098A	<148>	2440 DATA 5C, 89, 20, FD, AE, A9, 34, A0, EA, 8D, 14, 03, BC, 15, 03, A5, 7A, 077E	<129>
2080 DATA D0, 10, C0, 00, FO, 05, A9, 2B, 20, D2, FF, A9, 22, 20, D2, FF, E6, 08FC	<44>	2450 DATA A4, 7B, 85, F7, 84, FB, A5, 2B, A4, 2C, 85, 7A, 84, 7B, A0, 00, B1, 0906	<203>
2090 DATA FB, 6B, 20, D2, FF, CB, C0, 2B, D0, C2, C0, 00, D0, 0B, A9, 22, 20, 0919	<143>	2460 DATA 7A, CB, 11, 7A, CB, C9, 00, D0, 03, 4C, 47, BA, B1, 7A, 85, F9, CB, 08BF	<58>
2100 DATA D2, FF, 20, D2, FF, A5, FB, FO, 07, A9, 22, 20, D2, FF, C6, FB, A5, 0B7B	<218>	2470 DATA B1, 7A, 85, FA, 1B, A5, 7A, 69, 04, 85, 7A, A9, 00, 65, 7B, 85, 7B, 07D6	<43>

2480 DATA A0,FF,C8,B1,F7,F0,27,48,B1,7A,DO,14,68,C8,B4,02,18,094B	<222>	2850 DATA B7,D0,06,20,10,F7,4C,13,8C,AD,DD,B2,85,BA,A5,2B,A4,085E	<166>
2490 DATA A5,7A,65,02,85,7A,A9,00,65,7B,85,7B,4C,BA,B9,68,D1,07A6	<96>	2860 DATA 2C,85,C1,B4,C2,A5,2D,A4,2E,85,AE,84,AF,20,79,00,F0,084B	<133>
2500 DATA 7A,F0,DD,E6,7A,D0,02,E6,7B,4C,AF,89,AD,5C,89,D0,17,09D7	<127>	2870 DATA 2A,20,FD,AE,20,79,00,D0,03,4C,48,B2,20,8A,AD,20,F7,0715	<172>
2510 DATA A6,F9,A5,FA,20,CD,BD,20,A6,B6,AB,C0,08,F0,3F,A9,20,09CC	<151>	2880 DATA B7,A5,14,A4,15,85,C1,B4,C2,20,FD,AE,20,8A,AD,20,F7,08EE	<182>
2520 DATA 20,D2,FF,C8,D0,F4,AD,00,03,AC,01,03,85,FB,B4,FC,A9,0986	<24>	2890 DATA B7,A5,14,A4,15,85,AE,84,AF,4C,EA,F5,00,00,00,00,A9,0763	<101>
2530 DATA 1E,A0,BA,8D,00,03,8C,01,03,BA,86,FD,A5,F9,A4,FA,85,0866	<47>	2900 DATA 01,2C,A9,00,85,02,20,C4,8C,A9,A7,A0,8C,8D,02,03,8C,0667	<26>
2540 DATA 14,B4,15,20,13,A6,4C,BD,A6,A5,FB,A4,FC,8D,00,03,8C,0791	<237>	2910 DATA 03,03,A5,2E,85,2C,38,A5,2D,E9,02,85,2B,B0,02,C6,2C,05D3	<99>
2550 DATA 01,03,A6,FD,9A,A9,91,20,D2,FF,20,09,A9,B4,02,E6,07AC	<187>	2920 DATA A5,02,D0,03,4C,68,E1,A6,02,4C,1A,8B,20,AD,8C,4C,83,06D0	<144>
2560 DATA 18,A5,7A,65,02,85,7A,A9,00,65,7B,85,7B,4C,BA,89,A9,072E	<71>	2930 DATA A4,AD,74,8C,AC,75,8C,8D,02,03,8C,03,03,AD,76,8C,AC,077D	<123>
2570 DATA CA,A0,85,8D,14,03,8C,15,03,4C,86,E3,00,00,00,00,04EC	<107>	2940 DATA 77,8C,85,2B,84,2C,60,AD,02,03,AC,03,03,8D,74,8C,8C,0660	<194>
2580 DATA 00,A9,00,8D,58,BA,8D,59,BA,4C,73,00,20,79,00,C9,CC,0675	<21>	2950 DATA 75,8C,A5,2B,A4,2C,8D,76,8C,8C,77,8C,60,D0,38,A5,2B,07FA	<94>
2590 DATA F0,EE,20,8A,AD,20,F7,B7,A5,14,05,15,F0,E2,A5,14,A4,0905	<126>	2960 DATA A4,2C,85,7A,84,7B,A0,04,20,FB,AB,20,09,A9,CB,20,FB,07EA	<120>
2600 DATA 15,8D,56,BA,8C,57,BA,A9,00,8D,58,BA,A9,01,8D,59,BA,0721	<81>	2970 DATA AB,A0,00,A5,7A,91,2B,CB,A5,7B,91,2B,A9,FF,85,14,85,088D	<54>
2610 DATA 60,AD,59,BA,D0,02,F0,41,AD,58,BA,F0,37,18,AD,54,BA,084C	<133>	2980 DATA 15,20,13,A6,18,A5,5F,69,02,85,2D,A9,00,65,60,85,2E,0548	<60>
2620 DATA 6D,56,BA,8D,54,BA,AD,55,BA,6D,57,BA,8D,55,BA,AE,54,0800	<143>	2990 DATA 20,59,A6,4C,86,E3,60,00,00,00,00,00,00,00,00,00,0334	<1>
2630 DATA 8A,AD,55,BA,20,CD,BD,3B,20,A6,B6,AB,CB,84,C6,A9,20,08F7	<147>	3000 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,0000	<230>
2640 DATA 99,76,02,8B,A9,9D,20,D2,FF,A9,1D,99,76,02,8B,D0,F3,08F2	<140>	3010 DATA 00,4C,48,B2,20,79,00,F0,5F,C9,CC,F0,66,20,9E,B7,E0,086E	<112>
2650 DATA A9,00,8D,58,BA,20,60,A5,B6,7A,84,7B,20,73,00,AA,F0,0769	<211>	3020 DATA 21,B0,ED,8E,30,8D,20,FD,AE,20,9E,B7,E0,14,B0,E0,8E,095B	<35>
2660 DATA F3,A2,FF,86,3A,90,03,4C,96,A4,20,6B,A9,A5,14,A4,15,0813	<220>	3030 DATA 31,8D,20,FD,AE,20,9E,B7,E0,10,B0,D3,8E,32,8D,A9,00,0867	<171>
2670 DATA 8D,54,BA,8C,55,BA,A9,00,8D,7D,90,20,79,00,C9,00,F0,076B	<64>	3040 DATA 8D,33,8D,20,79,00,F0,0D,20,FD,AE,20,9E,B7,E0,02,B0,07B5	<172>
2680 DATA 05,A9,01,8D,58,BA,4C,9F,A4,20,46,52,4F,4D,20,00,20,0541	<31>	3050 DATA BC,8E,33,8D,A2,00,A9,00,9D,24,8D,EB,E0,0C,D0,FB,A5,08E4	<152>
2690 DATA 54,4F,20,00,A2,00,A9,00,F0,07,A9,01,A2,01,20,C4,BC,05C2	<132>	3060 DATA 39,A4,3A,C0,FF,F0,06,8D,2E,8D,8C,2F,8D,A9,01,8D,34,07C7	<5>
2700 DATA 20,36,8B,90,FA,A6,02,48,F0,03,20,AD,8C,68,4C,D1,E1,080D	<207>	3070 DATA 8D,60,AD,34,8D,09,02,8D,34,8D,4C,0C,8E,A9,00,8D,34,0604	<16>
2710 DATA 85,93,85,0A,86,02,20,79,00,D0,0D,A5,02,F0,03,20,AD,060C	<59>	3080 DATA 8D,4C,73,00,A5,9D,30,34,AD,34,8D,F0,0E,A5,39,CD,2E,0737	<0>
2720 DATA 8C,20,10,F7,4C,13,8C,20,57,E2,A5,B7,D0,06,20,10,F7,0750	<51>	3090 DATA 8D,D0,2B,A5,3A,CD,2F,8D,D0,24,AD,DE,82,F0,15,AD,8D,0930	<33>
2730 DATA 4C,13,8C,AD,DD,82,85,BA,A5,2B,A4,2C,85,C3,84,C4,A9,090F	<193>	3100 DATA 02,29,01,D0,15,AD,8D,02,29,04,F0,F9,AD,8D,02,29,04,05CC	<18>
2740 DATA 00,85,B9,A5,02,F0,0B,20,79,00,F0,2B,20,AD,8C,4C,08,063E	<115>	3110 DATA D0,F9,AD,8D,02,C9,01,F0,F9,4C,B2,B1,A2,02,A0,00,8D,0938	<49>
2750 DATA AF,20,79,00,F0,1D,20,FD,AE,20,79,00,F0,11,20,BA,AD,0711	<147>	3120 DATA 24,8D,EB,99,24,8D,CB,8D,24,8D,EB,99,24,8D,CB,C0,0C,08DF	<245>
2760 DATA 20,F7,B7,A5,14,A4,15,85,C3,84,C4,4C,9D,8B,A9,01,85,0873	<41>	3130 DATA D0,EC,A5,39,8D,2E,8D,A5,3A,8D,2F,8D,AD,B6,02,8D,19,07E5	<250>
2770 DATA B9,A9,00,85,90,A6,B9,20,AF,F5,A9,60,85,B9,20,D5,F3,09C9	<17>	3140 DATA 8D,A5,C7,8D,1A,8D,A5,D1,A4,D2,8D,1B,8D,BC,1C,8D,A5,092B	<150>
2780 DATA A5,BA,20,09,ED,A5,B9,20,C7,ED,20,13,EE,85,AE,A5,90,0930	<45>	3150 DATA D3,8D,1D,8D,A5,D4,8D,1E,8D,A5,D5,8D,1F,8D,A5,D6,8D,0976	<226>
2790 DATA 4A,4A,90,06,20,30,F5,4C,13,8C,20,13,EE,85,AF,BA,D0,0709	<127>	3160 DATA 20,8D,A5,DB,8D,21,8D,A5,F3,A4,F4,8D,22,8D,8C,23,8D,090D	<238>
2800 DATA 08,A5,C3,85,AE,A5,C4,85,AF,20,D2,F5,A5,9D,F0,0E,A9,0A10	<245>	3170 DATA AD,33,8D,F0,19,AC,30,8D,F0,14,AE,31,8D,CA,1B,20,F0,0841	<28>
2810 DATA 0C,A0,8B,20,1E,AB,A6,AE,A5,AF,20,CD,BD,20,F3,F4,08,08B1	<114>	3180 DATA FF,A2,00,A9,20,20,D2,FF,EB,E0,08,D0,F6,AC,30,8D,AE,0A0B	<74>
2820 DATA 85,AE,B6,AF,84,B0,A5,9D,F0,0E,A9,13,A0,8B,20,1E,AB,0BAC	<209>	3190 DATA 31,8D,1B,20,F0,FF,AD,32,8D,8D,86,02,A0,00,A9,12,20,06E1	<138>
2830 DATA A6,AF,A5,B0,20,CD,BD,A5,AE,A6,AF,A4,B0,28,4C,7B,E1,0A1D	<153>	3200 DATA D2,FF,A9,23,20,D2,FF,B9,24,8D,AA,B9,25,8D,CB,84,0A21	<61>
2840 DATA AA,6C,00,03,60,20,20,8C,90,FA,4C,D1,E1,20,57,E2,A5,07CB	<95>	3210 DATA 02,20,CD,BD,AD,30,8D,18,69,07,3B,E5,D3,29,27,AB,A9,072F	<207>

programme

3220 DATA 20,20,D2,FF,8B,DO,FB,AS,02,4A,1B,6D,31,8D,AA,AC,30,0B1B	<184>	3590 DATA 00,B1,7A,F0,E3,C9,CF,F0,0B,A0,01,20,FB,AB,4C,0D,91,0BDC	<237>
3230 DATA 8D,20,F0,FF,A4,02,C0,0C,DO,CO,AD,1A,8D,85,C7,AD,1B,0906	<123>	3600 DATA 20,73,00,C9,22,F0,03,4C,4B,B2,A0,01,20,FB,AB,A0,00,06BB	<223>
3240 DATA 8D,AC,1C,8D,85,D1,84,D2,AD,1E,8D,85,D4,AD,1F,8D,85,091D	<162>	3610 DATA B1,7A,D1,22,DO,2B,CB,C4,F9,DO,F5,20,FB,AB,20,79,00,09BC	<59>
3250 DATA D5,AD,21,8D,85,DB,AD,22,8D,AC,23,8D,85,F3,84,F4,AD,09E2	<213>	3620 DATA C9,00,F0,07,C9,22,DO,C4,20,73,00,AS,7A,A4,7B,85,5F,07F4	<174>
3260 DATA 19,8D,8D,86,02,AC,1D,8D,AE,20,8D,84,D3,86,D6,AD,34,0B00	<168>	3630 DATA B4,60,AS,F7,A4,FB,85,7A,84,7B,3B,60,20,FB,AB,4C,0D,0BCE	<34>
3270 DATA 8D,29,02,DO,03,4C,C6,8D,AD,34,8D,29,01,8D,34,8D,60,0670	<163>	3640 DATA 91,A9,71,A0,91,85,22,B4,23,6B,6B,4C,47,A4,4C,41,42,0700	<166>
3280 DATA 20,79,00,DO,03,4C,0B,AF,C9,CC,DO,0B,A9,00,8D,DE,B2,0772	<230>	3650 DATA 45,4C,20,4E,4F,54,20,46,4F,55,4E,C4,20,C3,90,B0,03,05E4	<115>
3290 DATA 4C,73,00,C9,91,DO,ED,A9,01,8D,DE,B2,4C,73,00,00,0791	<94>	3660 DATA 4C,A0,AB,AS,5F,A4,60,85,7A,84,7B,AS,14,A4,15,85,39,079A	<213>
3300 DATA 00,00,00,00,00,20,79,00,C9,CD,F0,03,4C,4B,A9,AS,9D,05A4	<2>	3670 DATA B4,3A,60,20,79,00,0B,A9,00,20,90,FF,2B,DO,06,20,59,05BE	<9>
3310 DATA F0,03,4C,AB,B3,20,73,00,C9,89,F0,03,4C,0B,AF,20,73,070B	<40>	3680 DATA A6,4C,AE,A7,20,60,A6,20,79,00,2B,80,91,4C,AE,A7,00,20,79,00,D9,0052	<244>
3320 DATA 00,20,C3,90,B0,1A,20,6B,A9,20,13,A6,0B,3B,AS,5F,E9,0677	<138>	3690 DATA 03,20,FB,AS,AS,7B,4B,AS,7A,4B,AS,3A,4B,AS,39,4B,A9,0786	<144>
3330 DATA 01,85,5F,AS,60,E9,00,85,60,2B,B0,03,4C,E3,AB,AS,5F,06E	<208>	3700 DATA 8D,4B,20,79,00,20,80,91,4C,AE,A7,A0,00,20,79,00,06,20,80,91,4C,AE,A7,A0,00,20,79,00,D9,0052	<98>
3340 DATA A4,60,8D,1F,8F,8C,20,8F,A9,84,A0,8F,8D,00,03,8C,01,06F3	<24>	3710 DATA 04,92,F0,06,CB,CO,0B,DO,F3,60,9B,0A,AB,6B,6B,A9,A7,0BA9	<53>
3350 DATA 03,A9,FE,A0,8F,8D,02,03,8C,03,03,4C,AE,A7,8A,10,03,063B	<122>	3720 DATA 4B,A9,AD,4B,B9,0D,92,4B,B9,0C,92,4B,AS,7A,A4,7B,8D,07F0	<91>
3360 DATA 4C,91,E3,8D,21,8F,20,FB,AB,AS,39,A4,3A,8D,24,8F,8C,0B45	<171>	3730 DATA 70,92,8C,71,92,4C,73,00,91,8B,8C,89,8A,8D,99,44,25,079A	<159>
3370 DATA 25,8F,AS,7A,A4,7B,8D,22,8F,8C,23,8F,AD,1F,8F,AC,20,0795	<174>	3740 DATA 8F,35,90,FB,84,7F,91,9B,91,B6,91,3E,92,71,92,20,A0,0BE6	<204>
3380 DATA 8F,85,7A,84,7B,4C,AE,A7,C9,CC,F0,03,4C,0B,AF,A9,B2,0BE4	<120>	3750 DATA E5,AD,DB,82,8D,20,DO,AD,DC,82,8D,21,DO,AD,DA,82,8D,0ABB	<119>
3390 DATA A0,90,8D,00,03,8C,01,03,4C,73,00,AS,9D,F0,03,4C,AB,063B	<42>	3760 DATA B6,02,60,A0,00,A9,00,99,00,D4,CB,CO,19,DO,FB,60,C9,0B30	<250>
3400 DATA B3,AD,22,8F,0D,23,8F,DO,0D,A9,82,A0,90,8D,00,03,8C,0724	<245>	3770 DATA DO,F0,06,20,79,00,4C,A0,AA,20,73,00,20,9E,B7,E0,2B,0705	<9>
3410 DATA 01,03,4C,29,90,AD,22,8F,AC,23,8F,85,7A,84,7B,A9,00,066C	<106>	3780 DATA 90,03,4C,4B,82,8A,4B,20,FD,AE,20,9E,B7,E0,19,90,03,0777	<50>
3420 DATA 8D,22,8F,8D,23,8F,AD,24,8F,AC,25,8F,85,39,84,3A,60,0719	<111>	3790 DATA 4C,4B,82,6B,AB,20,F0,FF,20,79,00,4C,A0,AA,00,00,C9,075D	<206>
3430 DATA A9,82,A0,90,8D,00,03,8C,01,03,A9,83,A0,A4,8D,02,03,067D	<120>	3800 DATA 93,DO,06,20,73,00,4C,1B,8B,C9,94,DO,06,20,73,00,4C,05FD	<20>
3440 DATA 8C,03,03,4C,83,A4,52,45,53,55,4D,45,20,57,49,54,4B,0532	<142>	3810 DATA 1B,8C,C9,95,DO,06,20,73,00,4C,1E,8B,C9,D1,DO,06,20,06F0	<136>
3450 DATA 4F,55,54,20,45,52,52,4F,D2,A9,15,A0,90,85,22,84,23,065E	<26>	3820 DATA 73,00,4C,7B,8C,AD,70,92,AC,71,92,85,7A,84,7B,A9,44,0B0C	<73>
3460 DATA 6B,6B,4C,47,A4,20,9E,AD,A9,00,85,02,20,79,00,C9,89,068D	<244>	3830 DATA 4C,AS,A9,AC,21,8F,4C,A2,B3,AC,24,8F,AD,25,8F,84,63,0B3E	<70>
3470 DATA F0,09,A9,A7,20,FF,AE,A9,80,85,02,AS,61,DO,13,20,73,0B42	<108>	3840 DATA 85,62,A2,90,3B,4C,49,BC,A9,25,A0,0B,85,F7,84,FB,A9,0BB9	<196>
3480 DATA 00,C9,00,F0,23,C9,CE,DO,F5,20,73,00,F0,1A,4C,80,91,0B32	<46>	3850 DATA 00,A0,80,85,F9,84,FA,A2,00,A0,00,B1,F7,91,F9,CB,DO,0A2B	<65>
3490 DATA AS,02,DO,06,20,73,00,4C,80,91,20,79,00,90,EE,6B,6B,0654	<163>	3860 DATA F9,E6,FB,E6,FA,EB,E0,13,DO,EE,6B,6B,4C,E2,FC,AS,01,0BFO	<26>
3500 DATA 20,79,00,4C,85,B1,60,00,00,00,00,00,00,A9,00,8D,7C,042D	<218>	3870 DATA 4B,7B,A9,F6,85,01,A9,00,A0,80,85,F7,84,FB,A9,00,A0,091F	<125>
3510 DATA 90,8A,30,09,AS,9D,DO,05,A9,01,8D,7D,90,4C,8B,E3,20,078B	<211>	3880 DATA 0B,85,F9,84,FA,A2,00,A0,00,B1,F7,91,F9,CB,DO,F9,E6,0AEF	<203>
3520 DATA 4A,89,AD,7D,90,F0,15,A9,01,8D,7C,90,AD,7E,90,AC,7F,0BBB	<87>	3890 DATA FB,E6,FA,EB,E0,14,DO,EE,6B,85,01,5B,A9,0D,A0,1B,85,09AE	<237>
3530 DATA 90,85,14,84,15,20,13,A6,4C,C9,A6,60,1B,20,9E,AD,24,065D	<244>	3900 DATA 2D,B4,2E,A6,02,20,BA,FF,A9,0E,A2,5F,A0,C3,20,8D,FF,0B57	<143>
3540 DATA 0D,30,03,4C,99,AD,20,A3,B6,60,20,79,00,F0,02,80,02,05E8	<129>	3910 DATA A9,2B,A6,2D,A4,2E,20,DB,FF,4C,9D,E3,45,5B,54,45,4E,07C0	<222>
3550 DATA 1B,60,1B,20,9E,AD,24,0D,30,03,4C,99,AD,20,A3,B6,85,05EF	<25>	3920 DATA 44,45,44,2D,42,41,53,49,43,FF,00,00,00,00,00,00,035B	<5>
3560 DATA F9,AS,7A,A4,7B,85,F7,84,FB,AS,2B,A4,2C,85,7A,84,7B,09CD	<82>	3930 DATA -1	<217>
3570 DATA C6,7A,AS,7A,C9,FF,DO,02,C6,7B,A0,01,B1,7A,CB,11,7A,0959	<56>		
3580 DATA F0,65,CB,B1,7A,CB,85,14,B1,7A,CB,85,15,20,FB,AB,A0,0999	<209>		

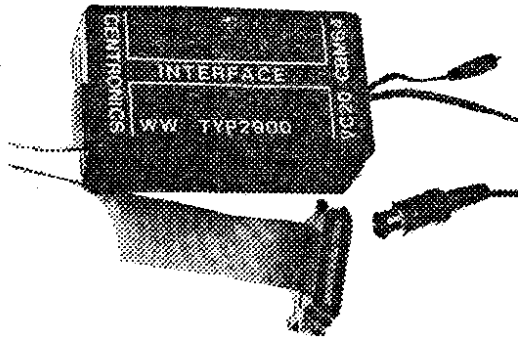


C64-Drucker an Schneider-Computer anschließen

Ja, Sie haben richtig gelesen!

Mit Hilfe des Interfaces Typ 29000 (ohne Pufferspeicher) oder Typ 29008 (mit 8KByte Pufferspeicher) kann man Drucker, die für Anschluß an den Commodore 64 gedacht sind, auch an Schneider-Computer anschließen.

Die Handhabung ist denkbar einfach: Der Adapter wird einfach zwischen Computer und Drucker geschaltet, so daß der Druck-



ker weiterhin für den Commodore 64 verwendbar ist.

Bei der Version mit dem 8KByte Pufferspeicher haben Sie den Vorteil, daß gleichzeitig gedruckt und weitergearbeitet werden kann.

Die beiden Interfaces werden von Wiesemann Mikrocomputertechnik, Wuppertal, zum Preis von 248,— DM (Typ 29000) und 298,— DM (Typ 29008) angeboten.

Neuer Monochrom-Datenmonitor für Commodore-Homecomputer

Eine neue Datenmonitor-Serie ist ab sofort von **MICROSCAN** lieferbar.

Mit einem großflächigen 31 cm Bildschirm, der mit einem Reflexionsschutz versehen ist, wird diese hochauflösende Monitor-Serie zu einem wichtigen Peripherie-Bestandteil.

Aufgrund einer Videobandbreite von 22 MHz präsentieren diese Datenmonitore alle Schriftzeichen und Grafiken flimmerfrei, gestochen scharf und in lesefreundlichem grün oder bernstein (amber). Der Refle-

xionsschutz verhindert die lästige Fremdlichtblendung.

Die Modelle

MS-1265/P42 (Bildschirm grün)

MS-1265/PLA (Bildschirm bernstein)

sind mit einem Composit-Video-Eingang versehen und damit an die meisten Commodore-Home-Computer über Cinch-Buchsen anschließbar. Über einen zusätzlichen Audio-Eingang lassen sich mit diesem Datenmonitor auch Tonsignale von z. B. Sprachsynthesizern wiedergeben.

Ein optimales Arbeiten mit dem Datenmonitor wird durch einen ergonomisch konstruierten Schwenkfuß gewährleistet.

MICROSCAN, 2000 Hamburg 60, empfiehlt folgende unverbindliche Endverkaufspreise inkl. MWSt:

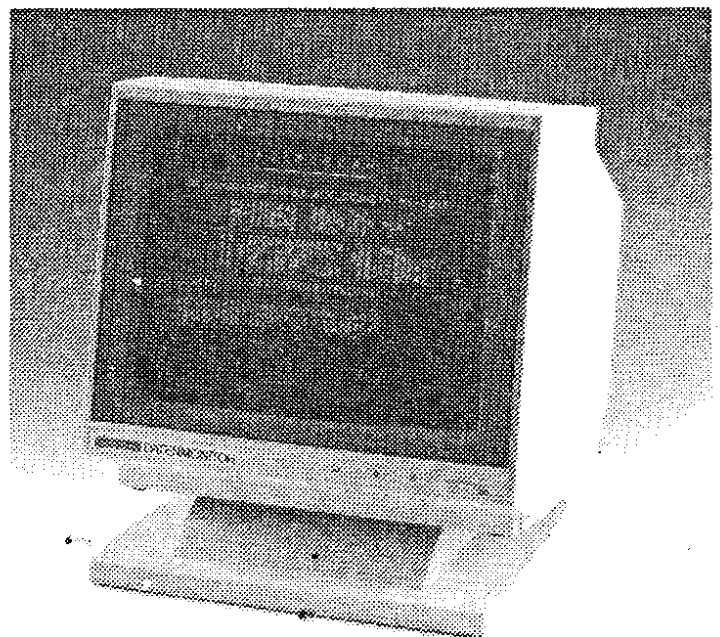
Datenmonitor MS-1265/P42 (Composit)

DM 399,—

Datenmonitor MS-1265/PLA (Composit)

DM 419,—

Techn. Daten		Modell	MS-1265 P42	MS-1265 PLA
Bildröhre	Typ		12", 90° P42 grün (entspiegelt)	12", 90° PLA bernstein (entspiegelt)
	Phosphor			
Eingangssignal Video	Art		composit	composit
	Pegel		1,0 ± 0,5 V _{ss}	1,0 ± 0,5 V _{ss}
	Impedanz Stecker		75 Ω Cinch	75 Ω Cinch
Eingangssignal Audio	Pegel		150 mV _{eff}	150 mV _{eff}
	Impedanz		10 kΩ	10 kΩ
	Stecker		Cinch	Cinch
Video	Bandbreite		22 MHz (− 3 dB)	22 MHz (− 3 dB)
Audio	Frequenzgang		300 Hz – 7 kHz (− 3 dB)	300 Hz – 7 kHz (− 3 dB)
	Ausgang		0,3 W/5% THD	0,3 W/5% THD
Abtastfrequenz	Horizontal		15,625 kHz ± 600 Hz	15,625 kHz ± 600 Hz
	Vertikal		47–63 Hz	47–63 Hz
Bildschirm	nutzbare Bildschirmfläche		242 x 182 mm	242 x 182 mm
	Darstellung		25 Zeilen x 80 Zeichen	25 Zeilen x 80 Zeichen
	Matrix		11 x 12	11 x 12
	Auflösung		920 x 300 Pixels	920 x 300 Pixels
Stromversorgung	Eingang		220 V (± 10%) 50/60 Hz	220 V (± 10%) 50/60 Hz
	Leistungsaufnahme		30 VA _{max}	30 VA _{max}
Umgebungstemperatur	Betrieb		10 bis 40 °C	10 bis 40 °C
	Lagerung		− 25 bis 55 °C	− 25 bis 55 °C
Luftfeuchte			20% bis 80% r.F.	20% bis 80% r.F.
Abmessungen	(B x H x T)		305 x 280 x 303 mm	305 x 280 x 303 mm
Gewicht			6 kg	6 kg



ROBCOM-MODULE FÜR DEN

Günstige Tool- und Utility-Steckmodule

Mit fünf ROBCOM-Modulen unterschiedlicher Leistung und unterschiedlicher Preisklasse, bietet Mastertronic GmbH eine neue Modulpalette für den C 64 an.

ROBCOM-Module sind Steckmodule, die eine Reihe Tools und Utilities für einen Preis zwischen 99,-- und 159,-- DM enthalten.

Das kleinste Modul, ROBCOM 10, bietet für einen Preis von nur 99,-- DM wichtige Zusatzfunktionen: Turbolader für Kassette, zusätzliche Diskettenfunktionen, Diskcopy, Tapecopy, Tapeazimuth mit freier Testkassette, jede Menge Basicerweiterungen, Funktionstastenbelegung, Centronicschnittstelle und einen Resettaster.

Das größte Modul, ROBCOM 50, welches für nur 159,-- DM erhältlich ist, bietet ein Kassetten- und Diskettenturbo. Dieser bringt eine 5-fache Geschwindigkeitssteigerung bei der Floppy und sogar eine 10-fache bei der Kassette. Der eingebaute Basicoolkit verfügt über 36 zusätzliche Befehle.

Neben dem Turbo und den Tools, verfügt ROBCOM 50 über Funktionstastenbelegung, Tapeazimuth mit Testkassette, Centronicsport und einem sehr komfortablen Assembler- und Disassembler.

ROBCOM 20, ROBCOM 30, ROBCOM 40 sind jeweils Module, die die Funktionen von ROBCOM 10 und ROBCOM 50 mischen.

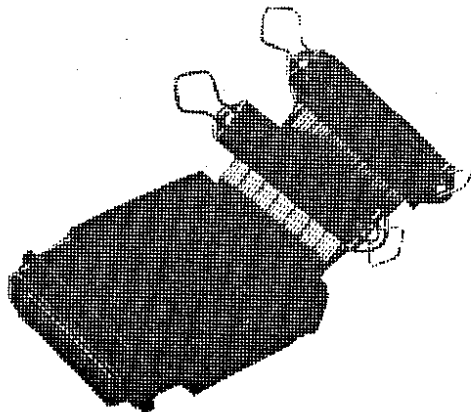
COMMODORE 64/128



Nähere Informationen:
Mastertronic GmbH
Kaiser Otto Weg 18
4770 Soest

Zwei Computer an einen Drucker anschließen

Als weitere Produktneuheit bietet die Fa. Wiesemann Mikrocomputertechnik, Wuppertal, den Schnittstellenvervielfacher Typ 2C/C/O an. Dieser bietet Ihnen die Möglichkeit, einen Drucker an zwei



verschiedene Computer (Centronics-schnittstellen) anzuschließen. Im Gegensatz zu den bisher erhältlichen, einfachen manuellen Umschaltern arbeitet dieses Gerät automatisch. D. h., zunächst können beide Computer Daten senden, und der Computer, der zuerst den Drucker belegt, blockiert den anderen Computer automatisch. Erst wenn der gerade druckende Computer nach einigen Sekunden keine Daten mehr sendet, beginnt der Drucker mit dem Ausdruck der Daten des anderen Computers. Auf diese Art wird eine Mischung der verschiedenen Daten vermieden. Der Preis liegt bei 248,-- DM und dürfte im Vergleich zu den manuellen Umschaltern als durchaus günstig angesehen werden.

Messevorschau: Micro-Computer '86

In der Zeit vom 29. Januar bis 2. Februar findet in Frankfurt die internationale Fachmesse „Mikrocomputer '86“ statt.

Der Schwerpunkt dieser Fachmesse liegt auf fünf Anwendergruppen: Handel, Handwerk, Industrie, Gesundheits- und Bildungswesen. Diese Messe legt besonderen Nachdruck auf die Orientierung der Anwender und wendet sich nicht nur an die EDV-Insider.

Hier soll der nötige Überblick über den Markt der dezentralen Informationsverarbeitung und praxisgerechten Problemlösung vermittelt werden.

So wird mit über sechzig Referaten, Seminaren, Diskussionen und Workshops das Thema allen Besuchern dieser Messe näher gebracht. Die Teilnahme an diesen Veranstaltungen ist im Eintrittspreis inbegriffen und kann von Jedermann besucht werden.

programme

```

270 PRINT"#####"
280 FORT=3271T03551STEP40:POKET,96:POKET-39,96:NEXT
290 COLOR1,3,3
300 PRINT"_____ "
310 PRINT"_____ "
320 PRINT"_____ "
330 COLOR1,11,4
340 PRINT"_____ "
350 PRINT"_____ "
360 PRINT"_____ "
370 PRINT"_____ "
380 COLOR1,10,2
390 PRINT"_____ "
400 PRINT"_____ "
410 PRINT"_____ " :POKE2902,119:POKE2903,119
420 COLOR1,10,0
430 PRINT"_____ "
440 PRINT"_____ "
450 CHAR1,0,24,"_____ " :POKE4071,64
460 RESTORE 2050
470 FORT=1T07:READA:POKEA,88:POKEA-1024,86:NEXT
480 REM *****
490 REM RAKETE
500 REM *****
510 COLOR1,2,6
520 CHAR1,20,6,"_":PRINT"⊗ _":COLOR1,15
530 CHAR1,20,7,"|":PRINT"⊗ _":COLOR1,15,4
540 CHAR1,20,8,"\":PRINT"⊗ _":COLOR1,15,3
550 CHAR1,20,9,"/":PRINT"⊗ _":COLOR1,15,1
560 CHAR1,20,10,"V":PRINT"⊗ _":COLOR1,2,3
570 CHAR1,20,11,"T":PRINT"⊗ _":COLOR1,2
580 GOTO670
590 REM *****
600 REM HAUPTPROGRAMM
610 REM *****
620 J=JOY(2)
630 DNJ+1GOTO750,650,660,670,680,690,700,710,720
640 GOTO1330
650 B=B-40: SOUND3,700,2: GOTO730
660 B=B-39: RI=65: SOUND3,500,2: GOTO730
670 B=B+1: RI=65: SOUND3,900,2: GOTO730
680 B=B+41: RI=65: SOUND3,500,2: GOTO730
690 B=B+40: SOUND3,700,2: GOTO730
700 B=B+39: RI=66: SOUND3,500,2: GOTO730
710 B=B-1: RI=66: SOUND3,900,2: GOTO730
720 B=B-41: RI=66: SOUND3,500,2: GOTO730
730 IFPEEK(B)<>32THENB40
740 POKEV,32:POKEB-1024,54:POKEB,RI:V=B
750 SOUND3,300,2
760 GOSUB990
770 CO=CO-.15
780 PRINT"#####"TAB(7);:PRINTUSING"###";CO
790 IFCO<0.5THEN1360
800 GOTO620
810 REM *****
820 REM GAME OVER
830 REM *****
840 IFPEEK(B)=81THENB=V:GOSUB1150:GOTO750
850 IFPEEK(B)>88THENB=V:GOTO750
860 IFPEEK(B)=88THENGOSUB1280:GOTO740
870 IFPEEK(B)=82THENB=V:GOSUB1190:GOTO750
880 POKEV,32:POKEB,85:POKEB-1024,113
890 FORT=800T0200STEP-30: SOUND3,T,2:NEXT
900 POKEB,86
910 FORT=500T01020STEP10: SOUND3,T,1:NEXT
920 POKEB,32
930 COLOR1,7
940 CHAR1,10,4,"##### GAME OVER #####"
950 FORT=1T03000:NEXT:CLR:GOTO110
960 REM ***
970 REM UFD
980 REM ***
990 DNJFBGOTO1000,1010,1020,1030,1040,
1050,1060,1070
1000 U=U-40: GOTO1080
1010 U=U-39: GOTO1080
1020 U=U+1: GOTO1080
1030 U=U+41: GOTO1080
1040 U=U+40: GOTO1080
1050 U=U+39: GOTO1080
1060 U=U-1: GOTO1080
1070 U=U-41: GOTO1080
1080 IFB=UTHENPOKER,32:GOTO880

```

**C 16/116 Software
wird ständig gesucht!
Senden Sie Ihre Unter-
lagen einfach an:**

Tronic-Verlag
Postfach 41
3444 Wehretal

**In dieser Ausgabe
finden Sie wieder
2 tolle Spiele
und
1 Anwendungs-
programm
unter der
Rubrik „Werkstatt“
für den C 16/116.**

```

1090 IFPEEK(U) <> 32 THEN U=R: UF=INT(RND(1)*8)+1: RETURN
1100 POKER, 32: POKEU-1024, 51: POKEU, 87: R=U
1110 RETURN
1120 REM *****
1130 REM AUFTANKEN
1140 REM *****
1150 IFTA<210 THEN TA=TA+30: SOUND1, 900, 2: ELSE RETURN
1160 SC=SC+20: PRINT "SC" TAB(24) SC
1170 PRINT "TA" TAB(7);: PRINT USING "###"; TA
1180 RETURN
1190 IFTA=0 THEN RETURN
1200 SC=SC+500: PRINT "SC" TAB(24) SC
1210 RT=RT+TA: FORT=800 TO 950 STEP 10: SOUND1, T, 2: SOUND2, T+50, 2: NEXT

```

1220 PRINT "RT" TAB(30) RT	1250 REM *****	1280 SC=SC+200: DI=DI+1
1230 TA=0: GOSUB 1170	1260 REM DIAMANT AUFNEHMEN	1290 SOUND1, 800, 20
1240 RETURN	1270 REM *****	1300 IF DI<7 THEN 1320

```

1310 FOR X=1 TO 4: FORT=700 TO 1020 STEP 50: SOUND1, Y, 2: NEXT: NEXT: SC=SC+1000

```

1320 PRINT "SC" TAB(24) SC: RETURN	1390 CHAR1, 19, 11, " * * "
1330 REM *****	1400 FORT=900 TO 700 STEP -8: SOUND3, T, 3: NEXT
1340 REM START DER RAKETE	1410 CHAR1, 19, 11, " * * "
1350 REM *****	1420 FORT=800 TO 100 STEP -10: SOUND3, T, 1: NEXT
1360 IF RT<1000 THEN 930	1430 RESTORE 2060
1370 SC=SC+2500: PRINT "SC" TAB(24) SC	1440 FORT=1 TO 27: READ A: SOUND1, A, 10: SOUND2, A, 5: NEXT
1380 COLOR1, 11, 5	

1450 GOTO 120	1590 CHAR1, 4, 10, " * * * * * * * * * * "
1460 REM *****	1600 FORT=2452 TO 2477 STEP 3: Y=Y+7: POKET, Y: NEXT
1470 REM ANFANGSBILD	1610 COLOR1, 15, X
1480 REM *****	1620 X=X+1: IF X=8 THEN X=0
1490 SCNCLR	1630 CHAR1, 8, 5, " R A K E T E N S T A R T "
1500 COLOR1, 16	
1510 CHAR1, 11, 22, " P R E S S F I R E B U T T O N "	1640 CHAR1, 4, 10, CHR\$(148): PRINT
1520 COLOR1, 2, 6	1650 POKE 3476, PEEK(3506): POKE 3506, 32
1530 CHAR1, 18, 12, " _ " : COLOR1, 15	1660 POKE 2452, PEEK(2482)
1540 CHAR1, 18, 13, " " : COLOR1, 15, 4	1670 IF JOY(2) > 127 THEN RETURN
1550 CHAR1, 18, 14, " \ " : COLOR1, 15, 3	1680 GOTO 1610
1560 CHAR1, 18, 15, " / " : COLOR1, 15, 1	1690 REM *****
1570 CHAR1, 18, 16, " v " : COLOR1, 2, 3	1700 REM DATAS FUER NEUEN ZEICHENSATZ
1580 CHAR1, 18, 17, " " "	1710 REM *****

```

1720 DATA 162, 0, 189, 0, 208, 157, 0, 48, 189, 0, 209, 157, 0, 49, 202, 208, 241, 96

```

1730 DATA 56, 116, 148, 114, 171, 204, 73, 29	1890 DATA 255, 255, 21, 117, 53, 117, 113, 255
1740 DATA 31, 4, 70, 253, 95, 31, 14, 0	1900 DATA 255, 255, 23, 119, 55, 119, 17, 255
1750 DATA 248, 32, 98, 191, 250, 248, 112, 0	1910 DATA 255, 255, 165, 153, 153, 165, 195, 129
1760 DATA 129, 195, 165, 153, 153, 165, 195, 129	1920 DATA 60, 178, 183, 215, 235, 171, 44, 40
1770 DATA 255, 66, 36, 24, 24, 36, 66, 255	1930 DATA 252, 172, 203, 215, 219, 199, 180, 120
1780 DATA 1, 1, 1, 3, 7, 15, 31, 63	1940 DATA 0, 8, 36, 60, 100, 40, 68, 1
1790 DATA 128, 128, 128, 192, 224, 240, 248, 252	1950 DATA 17, 68, 0, 18, 128, 10, 33, 145
1800 DATA 127, 255, 128, 255, 255, 128, 255, 255	1960 DATA 0, 24, 36, 126, 255, 231, 126
1810 DATA 254, 255, 1, 255, 255, 1, 255, 255	1970 DATA 0, 0, 16, 56, 68, 56, 16, 0
1820 DATA 247, 247, 247, 247, 247, 247, 240, 255	1980 DATA 0, 0, 255, 255, 255, 0, 0
1830 DATA 223, 223, 223, 223, 223, 223, 31, 255	1990 DATA 0, 0, 7, 15, 31, 60, 56
1840 DATA 255, 240, 247, 247, 240, 255, 255, 240	2000 DATA 0, 0, 192, 224, 240, 56, 56
1850 DATA 255, 31, 255, 255, 31, 223, 223, 31	2010 DATA 56, 56, 56, 56, 56, 56, 56, 56
1860 DATA 255, 255, 252, 253, 251, 240, 247, 247	2020 DATA 56, 56, 63, 31, 15, 0, 0
1870 DATA 255, 255, 63, 191, 223, 15, 239, 239	2030 DATA 56, 56, 56, 248, 240, 224, 0, 0
1880 DATA 255, 255, 56, 56, 124, 254, 254, 254	2040 REM

```

2050 DATA 3833, 3884, 3669, 3869, 4030, 3602, 3878
2060 DATA 516, 453, 516, 453, 516, 596, 643, 643, 596, 516, 453, 453, 345
2070 DATA 345, 453, 516, 516, 345, 770, 770, 643, 643, 770, 345, 345, 262, 262

```


Headline für C-16/116

HEADLINE C-16 erzeugt in den oberen beiden Zeilen des Bildschirms eine Kopfzeile, die ständig den Arbeitsmodus, den freien Speicherbereich, und die Systemuhr anzeigt.

Im Modus unterscheidet HEADLINE C-16 zwischen „DIRECT“ (Direkt-/Eingabemodus), „PROGR“ (Basic-Programmmodus) und „TEDMON“ (Maschinensprachemonitor). Der freie Basic-Bereich zeigt die Differenz zwischen Variablen-ende und String-anfang an. Die Zeit wird aus dem aktuellem Stand von TI errechnet.

```

0 REM *****
1 REM ***
2 REM *** HEADLINE C16 (BASIC) ***
3 REM ***
4 REM *** (C) BY JOACHIM THEES ***
5 REM *** GUTENTHAL 32B ***
6 REM *** 5552 MORBACH ***
7 REM *** - 16.08.1985- ***
8 REM *****
80 :
90 REM DATAS UEBERPRUEFEN
100 PRINT "OK"
110 PRINT "OK" TAB(10) "OK HEADLINE C16/116 OK"
120 PRINT "OK BITTE WARTEN: DATAS WERDEN GEPRUEFT"
130 RESTORE
140 FOR I=1 TO 6: SU=0
150 FOR J=1 TO 85: READ A: SU=SU+A: NEXT J
160 READ A: IFSU=ATHENNEXT I: GOTO 180
170 PRINT "OK FEHLER IN DATA-BLOCK NR. "; I; "!!!!": STOP: RUN
180 I=7: SU=0
190 FOR J=1 TO 68: READ A: SU=SU+A: NEXT J
200 READ A: IFSU<>ATHEN170
210 PRINT "OK DATAS OK !!!!!!!!!!! OK"
220 PRINT "OK WOLLEN SIE JETZT DAS MASCHINENPROGRAMM"
230 PRINT "OK ERZEUGEN ? (J/N)"
240 GETKEY A$: IFA$<>"J" THEN PRINT "OK NEIN": END
250 :
260 REM *****
270 :
280 REM DATENBLOCK #1
290 DATA 000,013,016,193,007,158,052,049,049,050,058,162,000,000,000,000,173
300 DATA 059,005,141,022,255,162,096,160,062,134,051,132,052,134,053,132,054
310 DATA 134,055,132,056,162,001,134,208,134,212,202,032,105,016,169,010,133
320 DATA 202,169,006,133,205,032,105,016,169,004,133,202,169,010,133,205,032
330 DATA 105,016,162,000,189,168,016,157,096,062,232,208,247,189,168,017,157
340 DATA 8693
350 :
360 REM DATENBLOCK #2
370 DATA 096,063,232,224,158,208,245,162,096,160,062,120,142,020,003,140,021
380 DATA 003,088,096,169,032,032,210,255,232,189,116,016,208,247,096,019,019
390 DATA 147,000,018,032,072,069,065,068,076,073,078,069,032,067,049,054,032
400 DATA 146,000,040,067,041,032,048,056,047,049,057,056,053,032,066,089,032

```

Nach dem Start des Basic-Programms erzeugt dieses ein Maschinenprogramm, das das eigentliche HEADLINE C-16 darstellt. Das Basic-Programm wird nicht weiter benötigt und kann gelöscht werden.

Nach dem Start des Maschinenprogramms wird am Ende des Basic-Speichers ein Interruptprogramm erzeugt und gestartet, das zweimal in jeder Sekunde durchlaufen wird, und das dabei die Kopfzeile erzeugt. Das Mpr. löscht sich nach seinem Ende selbst.

Nach dem Abtippen (man sollte die Basic-Zeilen von 900 abwärts exakt übernehmen, um einen Absturz des Programms zu vermeiden) und Starten des Programms meldet es, daß die DATA-Zeilen überprüft werden. Sollte ein Fehler gefunden werden, stoppt das Programm und meldet die entsprechende Blocknummer. Wenn die DATA-zeilen korrekt sind, stoppt das Programm nochmals und fragt, ob es fortgesetzt werden soll. Sie sollten hier „N“ eingeben und das Basicprogramm sicherheitshalber abspeichern, da es im 2. Teil zerstört wird.

Wenn Sie „J“ drücken, wird das Programm fortgesetzt und durch ein Maschinenprogramm ersetzt, das in den DATA-Zeilen abgelegt ist.

Nach wenigen Sekunden wird das Basicprogramm beendet. Wenn ein Listing 1985 SYS 4112:NEW

ergibt, kann das Maschinenprogramm abgespeichert werden. Dieses Maschinenprogramm ist das eigentliche Programm. Es kann wie ein normales Basic-Programm geladen, gespeichert und gestartet werden. Nach dem Start des Maschinenprogramms sollte auf dem Bildschirm eine Kopfzeile und ein Titelbild erscheinen. Falls dies nicht passiert, oder die Systemuhr nicht korrekt angezeigt wird, sollte das Basicprogramm überprüft und das Maschinenprogramm neu erzeugt werden.

Wenn das Maschinenprogramm korrekt arbeitet, kann das Basicprogramm auf der Kassette oder der Diskette gelöscht werden, das es nicht wieder benötigt wird.

Die Schriftfarbe der Kopfzeile kann vom Basic aus mit dem Befehl COLOR 3, ... geändert werden. Die Kopfzeile kann nur durch POKE 212,0 und zweimaliges Drücken der HOME-Taste gelöscht werden. Durch POKE 212,1 wird sie wieder aktiviert.

Bei gestartetem Interruptprogramm sollte die Hi-res Grafik nicht benutzt werden, da das Interruptprogramm dabei beschädigt werden könnte und der Computer dadurch abstürzen würde. Sollte dies passieren, können Basic-Programme nur durch Drücken der RESET-Taste bei gedrückter RUN STOP-Taste gerettet werden.

Um dies zu vermeiden, kann man eventuell im Maschinensprache-Monitor > 0314 0e ce RETURN

eingeben. Dadurch wird das Interruptprogramm völlig abgeschaltet. Die Uhr kann mit TI\$ = „UHRZEIT“ gestellt werden.

```

410 DATA 074,079,065,067,072,073,077,032,084,072,069,069,083,017,000,120,166
420 DATA 7310
430 :
440 REM DATENBLOCK #3
450 DATA 212,240,024,174,230,007,224,002,176,013,169,017,032,210,255,032,210
460 DATA 255,162,002,142,230,007,198,208,240,003,076,014,206,169,030,133,208
470 DATA 169,160,141,000,012,170,173,025,255,041,127,141,000,008,160,041,153
480 DATA 038,008,072,138,153,038,012,104,136,208,244,173,022,255,041,127,160
490 DATA 038,153,000,008,136,208,250,240,056,185,189,062,240,007,157,001,012
500 DATA 10157
510 :
520 REM DATENBLOCK #4
530 DATA 200,232,208,244,200,096,032,013,015,004,005,058,000,148,133,132,141
540 DATA 143,142,000,132,137,146,133,131,148,000,144,146,143,135,146,174,000
550 DATA 032,006,018,005,005,058,000,032,020,009,013,005,058,000,162,000,032
560 DATA 175,062,165,154,201,160,176,010,165,129,208,004,160,014,208,002,160
570 DATA 021,032,175,062,160,028,032,175,062,134,003,132,004,165,051,056,229
580 DATA 7990
590 :
600 REM DATENBLOCK #5
610 DATA 049,008,133,209,165,052,040,229,050,133,210,160,000,162,000,165,209
620 DATA 056,249,086,063,008,133,209,165,210,040,249,087,063,008,133,210,232
630 DATA 040,176,234,165,209,024,121,086,063,008,133,209,165,210,040,121,087
640 DATA 063,133,210,202,138,166,003,024,105,176,157,001,012,230,003,200,200
650 DATA 192,010,144,197,176,010,016,039,232,003,100,000,010,000,001,000,166
660 DATA 9585
670 :
680 REM DATENBLOCK #6
690 DATA 003,164,004,032,175,062,134,003,166,165,134,209,166,164,134,210,166
700 DATA 163,134,211,160,000,162,000,165,209,056,249,232,063,008,133,209,165
710 DATA 210,040,249,233,063,008,133,210,165,211,040,249,234,063,008,133,211
720 DATA 232,040,176,225,165,209,024,121,232,063,008,133,209,165,210,040,121
730 DATA 233,063,008,133,210,165,211,040,121,234,063,133,211,202,138,166,003
740 DATA 11577
750 :
760 REM DATENBLOCK #7
770 DATA 024,105,176,157,001,012,230,003,169,010,224,030,240,240,224,033,240
780 DATA 236,200,200,200,192,017,144,168,162,000,165,163,230,165,208,006,230
790 DATA 164,208,002,230,163,076,014,206,074,046,032,084,072,069,069,083,102
800 DATA 245,032,193,075,003,160,140,000,016,014,000,088,002,000,060,000,000
810 DATA 7526
820 :
830 REM *****
840 :
850 REM MASCHINENPROGRAMM ERZEUGEN
860 RESTORE:PRINT"*****"TAB(10)" HEADLINE C16/116 "
870 PRINT"BITTE WARTEN: DAS MASCHINENPROGRAMM"
880 PRINT"WIRD ERZEUGT."
890 KEY1," SAVE "+CHR$(34)+"HEADLINE C16"+CHR$(34)+" ,1"
900 FORI=4096TO4673:READA:IFA<256THENPOKEI,A:NEXT:ELSEREDA:POKEI,A:NEXT
910 PRINT"OK."
920 PRINT"ABSPEICHERN DES MPR. DURCH F1 !*****"
930 REM ZEIGER FUER BASIC-ENDE KORRIGIEREN
940 POKE45,68:POKE46,18:CLR
950 :
960 REM *****
970 :
980 END

```

★ ★ ★ Leserecke + Meckerecke ★ ★ ★

Computer-Kommunikation auch für C-16/116

Beim Durchblättern der Ausgabe 1/86 stieß ich auf S. 6 auf das Programm „Computer-Kommunikation“ für den Commodore 64. Da auch ich als C-16 Besitzer an diesem Programm interessiert bin, möchte ich fragen, ob dieses Programm nicht für den C-16 erhältlich ist?

Stephan Scholl, Essen

Redaktion: Das Programm ist nur auf dem Commodore 64 lauffähig. Eine Version für den C-16/116 liegt uns hier leider nicht vor. Vielleicht schreibt jedoch einer unserer Leser dieses Programm für dieses System um und schickt es uns zur Veröffentlichung ein.

Fehlerhafte C-16/116 Listings

Ich habe ein Programm für den C-16/116 abgetippt und mußte dann leider feststellen, daß es nicht funktioniert. Muß man annehmen, daß das Spiel „Hueycobra“ absichtlich mit Fehlern abgedruckt wird? Ich finde es peinlich, wenn man immer für das selbe System (C-16/116) Reklamationen bekommt. Das gilt auch für das Programm „Helmut und die Zeitmaschine“.

Roswitha Patro, Essen

Redaktion: Um Ihnen gezielt weiterhelfen zu können, hätten Sie uns eine genaue Fehlermitteilung geben müssen. Da aber die Zuschriften mit den fehlerhaften Listings für den C-16/116 nicht abnehmen, möchten wir nochmals einige Hinweise weitergeben. Tippen Sie die einzelnen Programmzeilen sehr sorgfältig ein und vergleichen Sie diese vor dem Abspeichern nochmals ganz genau. Wir können aus Erfahrung sagen, daß die meisten Fehler bei der Eingabe der Datenzeilen für die

Grafikzeichen entstehen. Diese Fehler haben dann zur Folge, daß das Bild verzerrt und keine Fehlermeldung sichtbar wird. Achten Sie also bei der Eingabe der Datenzeilen auf evtl. Fehler.

Ein sicherer Beweis für die Funktionstüchtigkeit des Programms „Helmut und die Zeitmaschine“ ist die Einsendung des Herrn Berthold Bronisz aus Köln. Er sandte uns nämlich einen Verbesserungsvorschlag der es erlaubt, das Programm mit Joystick und über Tastatur zu steuern:

```
370 F=JOY(1):TA=PEEK(198)
371 IFF=3orTA=51THENC=C+1
380 IFF=7orTA=48THENC=C-1
390 IFF=128orTA=21THENGOSUB500:
    E-E-25
```

```
400 IFF=10RTA=43THENC=-80:E=E-5
```

Geben Sie diese Zeilen ein und die Steuerung funktioniert. Bei einem fehlerhaften Listing wäre ein Verbesserungsvorschlag nicht möglich gewesen. Wir können also nur sagen, alle bisher veröffentlichten Programme waren fehlerfrei und bedürfen deshalb auch keiner Korrektur. Vielen Dank auch an Herrn Bronisz!

Beantwortung der Leserbriefe

Werden auch solche Leserbriefe beantwortet, die nicht abgedruckt werden? Da ich keinen Drucker besitze, möchte ich wissen, ob es noch eine andere Möglichkeit gibt, ein Listing einzuschicken?

Klaus Konrad, Dortmund

Redaktion: Der Tronic-Verlag ist bemüht, sämtliche schriftlichen oder mündlichen Zuschriften oder Anfragen zu beantworten. Wir verstehen diesen Service als eine Selbstverständlichkeit. Bei der Einsendung der Software-Unterlagen ist ein Listing nicht unbe-

dingt erforderlich. Es genügt der Datenträger, eine Programmbeschreibung und ein ausreichend frankierter Rückumschlag.

VC-20 Zeilennummern-Problem

Wie bringt man Programme in den VC-20, wenn es mehrere Teile sind, die immer wieder mit der Zeilennummer 1 anfangen?

Walter Kessler, Hamburg

Redaktion: VC-20 Programme die aus mehreren Teilen bestehen enthalten in der Regel Maschinenroutinen. Die einzelnen Teile müssen getrennt eingegeben und hintereinander abgespeichert werden. Bevor man einen der Teile eingibt, sollte der Speicher mit dem Befehl NEW gelöscht werden, um sicher zu gehen, daß keine Restdaten im Speicher stehen.

Unvollständige ABO-Kassetten

Nach dem Erhalt der Nr. 1/86 incl. Kassette bin ich etwas sauer! Um die Eintipperei sicher zu machen druckt Ihr das „Checksummen-Programm“ ab, aber auf der Kassette fehlt es. Wenn kleine Programme mit zwei, drei Zeilen nicht auf der Kassette sind ist dies ja weiter nicht schlimm. Aber in diesem Fall ist das ärgerlich. Ich hoffe, daß dies ein einmaliger Ausrutscher war.

Michael Dröhsel, Berlin

Redaktion: Zunächst möchten wir uns bei allen Abonnenten für dieses Mißgeschick entschuldigen. Wir haben in der Tat vergessen das Programm „Checksummer“ auf die Kassetten zu kopieren. Bei allen weiteren Lieferungen werden wir bemüht sein, daß sich ein solches Mißgeschick nicht wiederholt und alle Datenträger vollständig Ihren Empfänger erreichen.

programme

```

500 PRINT" DISZIPLINEN: REKORDE: "
510 PRINT"1 - 50 METER SPURT - 3 SEKUNDEN"
520 PRINT"2 - 50 METER HUERDEN - 6 SEKUNDEN"
530 PRINT"3 - WEITSPRUNG - 6 METER"
540 PRINT"PUNKTEREKORD 85 PUNKTE - SWEN O."
550 PRINT"COPYRIGHT (C) 1985 BY SWEN OTTMANN."
560 F=INT(RND(2)*14)+2:COLOR1,F,4:NEXTZZ
570 PRINT"C":COLOR1,3,3
580 CLR:X=0:J=1:POKE1344,0
590 POKE65298,PEEK(65298)AND251
600 POKE65299,PEEK(65299)AND3OR48
610 FORA=832TO849:READB:C=C+B
620 POKEA,B:NEXTA
630 IF C > 2115 THEN PRINT"?DATA ERROR":END
640 SYS832:VOLB
650 FORD=12800TO12871:READE:POKED,E:NEXT
660 DATA162,000,189,000,208,157,000,048
670 DATA189,000,209,157,000,049,202,208
680 DATA241,096
690 :
700 REM AB HIER NEUE GRAFIKZEICHEN
710 :
720 DATA060,066,165,129,129,153,066,060
730 DATA000,000,024,024,024,024,024,024
740 DATA024,052,062,060,024,060,061,126
750 DATA188,060,024,024,040,068,068,103
760 DATA024,052,062,060,024,060,060,063
770 DATA060,060,024,056,088,144,144,092
780 DATA060,060,024,100,131,000,000,000
790 DATA060,060,056,025,008,000,000,000
800 DATA255,000,255,000,255,000,255,000
810 :
820 REM 50 METER SPURT (3 SEKUNDEN)
830 :
840 FORA=0TO38:FORB=20TO24
850 CHAR1,A,B,"█ █":NEXTB,A
860 COLOR1,2:CHAR1,38,19,"+"
870 FORA=0TO10:FORB=0TO6:COLOR1,2,2
880 CHAR1,A,B,"-":NEXTB,A
890 FORA=10TO20:FORB=0TO6:COLOR1,6,2
900 CHAR1,A,B,"-":NEXTB,A

```

Sport

Spiel

Spannung

```

910 FORA=20TO30:FORB=0TO6:COLOR1,7,2
920 CHAR1,A,B,"-":NEXTB,A
930 FORA=30TO38:FORB=0TO6:COLOR1,8,2
940 CHAR1,A,B,"-":NEXTB,A
950 COLOR1,2:CHAR1,13,22,"50 METER LAUF"
960 CHAR1,12,12,"AUF DIE PLAETZE"
970 FORA=0TO1000:NEXTA
980 CHAR1,12,12," FERTIG "
990 FORA=0TO1000:NEXTA
1000 CHAR1,12,12," PENG !!! "
1010 FORA=0TO100:NEXTA:SOUND3,999,5
1020 CHAR1,12,12," "
1030 TI$="000000":COLOR1,7,5
1040 CHAR1,X,18,"|"
1050 CHAR1,X,19,"-"
1060 CHAR1,X,18,"-"
1070 CHAR1,X,19,"-"
1080 IFJOY(1)=0THEN1040
1090 IFJ=2ANDJOY(1)=3THENJ=1:GOSUB1130
1100 IFJ=1ANDJOY(1)=7THENJ=2:GOSUB1130
1110 IFX>37THEN1170
1120 GOTO1040
1130 X=X+1:CHAR1,X,18,"|"
1140 CHAR1,X,19,"-":Y=X
1150 CHAR1,X-1,18," "
1160 CHAR1,X-1,19," ":X=Y:RETURN
1170 T$=TI$:PRINT"C":IFT$>"000006"
THEN1260
1180 FORA=0TO50
1190 CHAR1,8,5,"QUALIFIZIERT"
1200 CHAR1,13,10,"DEINE ZEIT IST"
1210 CHAR1,17,12,T$
1220 CHA 16,14,"SEKUNDEN"
1230 F=IN: (ND(2)*14)+2
1240 H=INT(RND(1)*6)+1:COLOR1,F,H

```

```

1250 NEXTA:COLOR1,2:SCNCLR:T=VAL(T$):S1=100-T:GOTO1380
1260 FORA=0TO50
1270 CHAR1,5,5,"DISQUALIFIZIERT"
1280 CHAR1,9,10,"TUT MIR WIRKLICH LEID"
1290 CHAR1,10,12,"ABER DEINE ZEIT HAT"
1300 CHAR1,10,14,"DIE 6 SEKUNDENGRENZE"
1310 CHAR1,11,16,"UEBERSCHRITTEN !!!"
1320 F=INT(RND(2)*14)+2
1330 H=INT(RND(1)*6)+1:COLOR1,F,H
1340 NEXTA:COLOR1,2:RUN
1350 :
1360 REM 50 METER HUERDEN (6 SEKUNDEN)
1370 :
1380 COLOR1,3,3:FORA=0TO38:FORB=20TO24
1390 CHAR1,A,B,"█ █":NEXTB,A:COLOR1,2
1400 FORA=5TO33STEP7:CHAR1,A,19,"+":NEXTA
1410 FORA=0TO10:FORB=0TO6:COLOR1,2,2

```

```

1420 CHAR1,A,B,"-":NEXTB,A
1430 FORA=10TO20:FORB=0TO6:COLOR1,6,2
1440 CHAR1,A,B,"-":NEXTB,A
1450 FORA=20TO30:FORB=0TO6:COLOR1,7,2
1460 CHAR1,A,B,"-":NEXTB,A
1470 FORA=30TO38:FORB=0TO6:COLOR1,8,2
1480 CHAR1,A,B,"-":NEXTB,A:COLOR1,2
1490 CHAR1,11,22,"50 METER HUERDEN"
1500 CHAR1,12,12,"AUF DIE PLAETZE"
1510 FORA=0TO1000:NEXTA
1520 CHAR1,12,12," FERTIG "
1530 FORA=0TO1000:NEXTA
1540 CHAR1,12,12," PENG !!! "
1550 FORA=0TO100:NEXTA:SOUND3,999,5

```

```

1560 CHAR1,12,12," "
1570 X=0;TI$="000000";COLOR1,7,5
1580 CHAR1,X,18,"|"
1590 CHAR1,X,19,"-"
1600 CHAR1,X,18,"-"
1610 CHAR1,X,19,"-"
1620 IFJOY(1)=0THEN1580
1630 IFJ=2ANDJOY(1)=3THENJ=1;GOSUB1730
1640 IFJ=1ANDJOY(1)=7THENJ=2;GOSUB1730
1650 IFJOY(1)=131THENGOSUB1770
1660 IFX=5THENR=R+2
1670 IFX=12THENR=R+2
1680 IFX=19THENR=R+2
1690 IFX=26THENR=R+2
1700 IFX=33THENR=R+2
1710 IFX>35THEN1870
1720 GOTO1580
1730 X=X+1;CHAR1,X,18,"|"
1740 CHAR1,X,19,"-" : Y=X
1750 CHAR1,X-1,18," "

```

```

1760 CHAR1,X-1,19," " : X=Y;RETURN
1770 Y=X;CHAR1,X,18," "
1780 CHAR1,X,19," " : X=Y
1790 X=X+1;CHAR1,X,16,"|"
1800 CHAR1,X,17,"-" : Y=X
1810 CHAR1,X,16," "
1820 CHAR1,X,17," " : X=Y
1830 X=X+1;CHAR1,X,16,"|"
1840 CHAR1,X,17,"-" : Y=X
1850 CHAR1,X,16," "
1860 CHAR1,X,17," " : X=Y;RETURN
1870 T$=TI$;PRINT"Q":IFT$>"000009"
    THEN1960
1880 FORA=0TO50
1890 CHAR1,8,5,"Q U A L I F I Z I E R T"
1900 CHAR1,13,10,"DEINE ZEIT IST"
1910 CHAR1,17,12,T$
1920 CHAR1,16,14,"SEKUNDEN"
1930 F=INT(RND(2)*14)+2
1940 H=INT(RND(1)*6)+1;COLOR1,F,H

```

```

1950 NEXTA;COLOR1,2;SCNCLR:T=VAL(T$);S2=B1-T-R;GOTO2080

```

```

1960 FORA=0TO50
1970 CHAR1,5,5,"D I S Q U A L I F I Z I E R T"
1980 CHAR1,9,10,"TUT MIR WIRKLICH LEID"
1990 CHAR1,10,12,"ABER DEINE ZEIT HAT"
2000 CHAR1,10,14,"DIE 9 SEKUNDENGRENZE"
2010 CHAR1,11,16,"UEBERSCHRITTEN !!!"
2020 F=INT(RND(2)*14)+2
2030 H=INT(RND(1)*6)+1;COLOR1,F,H
2040 NEXTA;COLOR1,2;RUN
2050 !
2060 REM WEITSPRINGEN (GESCHAFFT!)
2070 !
2080 COLOR1,3,3;SCNCLR;FORA=0TO38;FORB=20TO24
2090 CHAR1,A,B,"□ ■":NEXTB,A;COLOR1,8,4
2100 FORA=30TO36;CHAR1,A,20,"□ ■":NEXTA
2110 FORA=0TO10;FORB=0TO6;COLOR1,2,2
2120 CHAR1,A,B,"-":NEXTB,A
2130 FORA=10TO20;FORB=0TO6;COLOR1,6,2
2140 CHAR1,A,B,"-":NEXTB,A
2150 FORA=20TO30;FORB=0TO6;COLOR1,7,2
2160 CHAR1,A,B,"-":NEXTB,A
2170 FORA=30TO38;FORB=0TO6;COLOR1,8,2
2180 CHAR1,A,B,"-":NEXTB,A
2190 COLOR1,2;CHAR1,15,22,"WEITSPRUNG"
2200 SOUND1,0,30;FORA=0TO500;NEXTA
2210 TI$="000000";X=0;COLOR1,7,5
2220 CHAR1,X,18,"|"
2230 CHAR1,X,19,"-"
2240 CHAR1,X,18,"-"
2250 CHAR1,X,19,"-"
2260 IFJOY(1)=0THEN2220
2270 IFJ=2ANDJOY(1)=3THENJ=1;GOSUB2310
2280 IFJ=1ANDJOY(1)=7THENJ=2;GOSUB2310
2290 IFX>29THEN2350
2300 GOTO2220
2310 X=X+1;CHAR1,X,18,"|"
2320 CHAR1,X,19,"-" : Y=X

```

```

2330 CHAR1,X-1,18," "
2340 CHAR1,X-1,19," " : X=Y;RETURN
2350 T$=TI$;IFT$>"000007"THEN2610
2360 IFT$="000007"THENSS=7
2370 IFT$="000006"THENSS=6
2380 IFT$="000005"THENSS=5
2390 IFT$<"000004"THENSS=4
2400 WE=SS-10;FR=32+WE;CHAR1,X,18," "
2410 CHAR1,X,19," " : FORSP=29TOFR
2420 CHAR1,SP,17,"|"
2430 CHAR1,SP,18,"|"
2440 CHAR1,SP,17," "
2450 CHAR1,SP,18," " : NEXTSP
2460 CHAR1,SP+1,19,"|"
2470 CHAR1,SP+1,20,"□ ■":FORA=0TO2500;NEXTA
2480 SCNCLR;FORA=0TO100
2490 CHAR1,8,3,"Q U A L I F I Z I E R T"
2500 CHAR1,11,6,"*** WEITSPRUNG ***"
2510 IFSS=7THENCHAR1,15,8,"-3 METER-"
2520 IFSS=6THENCHAR1,15,8,"-4 METER-"
2530 IFSS=5THENCHAR1,15,8,"-5 METER-"
2540 IFSS=4THENCHAR1,15,8,"-6 METER-"
2550 CHAR1,12,10,"*** DU HAST ***"
2560 S3=B2-SS;S3$=STR$(S3);CHAR,18,12,S3$
2570 CHAR1,6,14,"PUNKTE IM DREIKAMPF EREICHT"
2580 F=INT(RND(2)*14)+2
2590 H=INT(RND(1)*6)+1;COLOR1,F,H
2600 NEXTA;COLOR1,2;SCNCLR;RUN
2610 SCNCLR;FORA=0TO100
2620 CHAR1,5,5,"D I S Q U A L I F I Z I E R T"
2630 CHAR1,9,10,"TUT MIR WIRKLICH LEID"
2640 CHAR1,8,12,"ABER DIE GESCHWINDIGKEIT"
2650 CHAR1,11,14,"WAR DEIN SCHICKSAL"
2660 F=INT(RND(2)*14)+2;H=INT(RND(1)*6)+1
2670 COLOR1,F,H;NEXTA;COLOR1,2;SCNCLR
2680 RUN

```



Programm-Angebote des Jahrgangs '85

Alle veröffentlichten Programme aus der Zeitschrift „Compute mit“ Jahrgang '85 können auch weiterhin noch auf Kassette bzw. Diskette nachbestellt werden. Jeder der noch nicht eine vollständige Programm-Sammlung hat, kann diese also durch Nachbestellung ergänzen.

Die notwendigen Angaben entnehmen Sie bitte der Bestellliste. Die angegebenen Bestellnummern gelten jeweils für eine Heft-Ausgabe. Beachten Sie bitte auch die Angaben zu den Bestell- und Lieferbedingungen.

Heft 5/85

Commodore	
Bestell-Nr. COM C 1/1	16,- DM
Bestell-Nr. COM D 1/1	20,- DM
Schneider	
Bestell-Nr. COM S 1/1	16,- DM

Heft 7/85

Commodore	
Bestell-Nr. COM CK 3/1	16,- DM
Bestell-Nr. COM CD 3/1	20,- DM
Schneider	
Bestell-Nr. COM SK 3/1	16,- DM
Bestell-Nr. COM SD 3/1	39,- DM

Heft 9/85

Commodore	
Bestell-Nr. COM CK 5/1	16,- DM
Bestell-Nr. COM CD 5/1	20,- DM
Schneider	
Bestell-Nr. COM SK 5/1	16,- DM
Bestell-Nr. COM SD 5/1	39,- DM

Heft 11/85

Commodore	
Bestell-Nr. COM CK 7/1	16,- DM
Bestell-Nr. COM CD 7/1	20,- DM
Schneider	
Bestell-Nr. COM SK 7/1	16,- DM
Bestell-Nr. COM SD 7/1	39,- DM

Heft 6/85

Commodore	
Bestell-Nr. COM C 2/1	16,- DM
Bestell-Nr. COM D 2/1	20,- DM
Schneider	
Bestell-Nr. COM S 2/1	16,- DM

Heft 8/85

Commodore	
Bestell-Nr. COM CK 4/1	16,- DM
Bestell-Nr. COM CD 4/1	20,- DM
Schneider	
Bestell-Nr. COM SK 4/1	16,- DM
Bestell-Nr. COM SD 4/1	39,- DM

Heft 10/85

Commodore	
Bestell-Nr. COM CK 6/1	16,- DM
Bestell-Nr. COM CD 6/1	20,- DM
Schneider	
Bestell-Nr. COM SK 6/1	16,- DM
Bestell-Nr. COM SD 6/1	39,- DM

Heft 12/85

Commodore	
Bestell-Nr. COM CK 8/1	16,- DM
Bestell-Nr. COM CD 8/1	20,- DM
Schneider	
Bestell-Nr. COM SK 8/1	16,- DM
Bestell-Nr. COM SD 8/1	39,- DM

Unser Software-Service ist in der Lage, Spiel- und Anwendungsprogramme für alle Commodore- und Schneidersysteme innerhalb

nur 1 Woche zu liefern.

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

In unserer Bestellliste finden Sie Software für alle Anwendungsgebiete. Schauen Sie einmal unverbindlich auf unser Angebot. Vielleicht können wir auch Sie von unse-

ren hervorragenden Angeboten überzeugen, und Sie senden uns den ausgefüllten Bestellcoupon ein oder greifen einfach zum Telefon und bestellen telefonisch unter den Nummern:

0 56 51 / 4 06 43

oder 4 06 93

Bitte beachten:

Der Preis für den Datenträger gilt nur in Verbindung mit der Einsendung des Bestellcoupons. Ohne Coupon erhöht sich der Preis entsprechend (24,50 DM pro Kassette, 29,50 Commodore-Diskette und 44,90 DM Schneider-Diskette).

Heft 1/86

Commodore

Bestell-Nr. COM CK 1/2 16,- DM
Bestell-Nr. COM CD 1/2 20,- DM

Schneider

Bestell-Nr. COM SK 1/2 16,- DM
Bestell-Nr. COM SD 1/2 39,- DM

Aus diesem Heft:

Commodore

Bestell-Nr. COM CK 2/2 16,- DM
Bestell-Nr. COM CD 2/2 20,- DM

Schneider

Bestell-Nr. COM SK 2/2 16,- DM
Bestell-Nr. COM SD 2/2 39,- DM

Weitere Programmangebote für Ihr System aus dem Tronic-Verlag entnehmen Sie bitte der nachfolgenden Bestellliste.

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

C64

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

VIC-20

Bestellschein/Glucky	8,50 DM	15,- DM	V 61
Multigraph/All Rammer	11,- DM	15,50 DM	V 71
Zyklo/Meteorit	11,- DM	19,50 DM	V 81

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

Schneider

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

programme

Feindliche Invasoren sind in der Sahara gelandet. Um zu den wichtigsten Hauptstädten der Erde zu gelangen, haben sie einen langen Tunnel unter dem Atlantik bis nach Amerika gegraben. Durch einen Zufall ist der Tunnelausgang entdeckt worden und man hat Sie hineingeschickt, um die feindlichen Invasoren im Tunnel zu vernichten.

Aber Achtung: Ihre Gegner schießen scharf und der Tunnel ist durch Minen und Mauern gesichert. Versuchen Sie möglichst viele Ihrer Gegner zu erledigen, bevor diese Sie erwischen.

DEFENDER

für VC-20 ohne Erweiterung

Das Programm ist völlig in Maschinensprache geschrieben und belegt 3 K des vorhandenen Speichers. Darum ist es unmöglich, das Programm in DATAzeilen einzugeben.

Um das Programm dennoch abtippen zu können, habe ich einen kleinen Monitor geschrieben, der lediglich 384 Bytes belegt und so zusätzlich zum Programm in den Speicher paßt.

Der Monitor ist in DATAzeilen abgelegt. Um ihn in den Speicher zu bringen sind folgende Operationen notwendig:

NEW
POKE 44,18:POKE 4608,0:NEW

Eingabe und Start des DATA-Programms:

POKE 44,16:NEW
10 SYS 4324
POKE 46,18:POKE 45,0:CLR

Abspeichern des Monitors:

Mit M aaaa kann der Speicherinhalt an der Stelle aaaa ausgegeben werden, mit M aaaa bbbb der Speicherbereich von aaaa bis bbbb.

Jede Zeile besteht aus vier Datenbytes und einem Kontrollbyte. Passen Datenbytes und Kontrollbyte nicht zusammen, wird die Zeile nicht angenommen.

Ist das Programm fertig im Speicher, wird der Monitor mit STOP/RESTORE verlassen.

Danach NEW und 10 SYS 6656, sowie POKE 46,29:POKE 45,240:CLR eingeben und das Programm abspeichern.

Auf der Kassette befindet sich das fertige Programm. Der fertige Monitor und der Monitor in DATAzeilen sind ebenfalls gespeichert.

Zum Ausdruck des HEXDUMPS:

DEFENDER laden, Monitor laden
OPEN 4,4:CMD4 eingeben, sowie
RUN und M 1200 1DFF eintippen.

/.1200 53 00 00 00 53	/.1274 C9 06 B0 02 7D	/.12E8 EA EA EA EA 00	/.135C A2 15 A0 15 02
/.1204 00 00 00 44 44	/.1278 A9 20 88 91 90	/.12EC EA EA EA EA 00	/.1360 A9 01 B5 00 2D
/.1208 00 00 00 00 00	/.127C 01 C8 B1 03 7B	/.12F0 EA EA EA EA 00	/.1364 E4 01 B0 04 51
/.120C 00 00 AD 20 8D	/.1280 88 91 03 E6 FC	/.12F4 EA EA EA EA 00	/.1368 A9 09 D0 14 64
/.1210 91 48 A9 FF 8F	/.1284 01 E6 03 D0 34	/.12FB EA EA EA EA 00	/.136C D0 04 A5 03 72
/.1214 8D 22 91 68 56	/.1288 E7 E6 02 E6 E5	/.12FC EA EA EA EA 00	/.1370 D0 0E A9 20 57
/.1218 0A 2A 29 01 08	/.128C 04 CA D0 E0 FE	/.1300 E6 F7 20 60 51	/.1374 E4 02 D0 04 32
/.121C 0D FF 1F 60 8D	/.1290 EA A9 1F 85 D9	/.1304 12 A5 F7 18 58	/.1378 A5 04 D0 04 75
/.1220 48 A9 0F 85 6B	/.1294 D2 A9 97 85 69	/.1308 69 15 AA BD 6B	/.137C 90 02 A9 09 32
/.1224 D2 BA 85 D1 0C	/.1298 F4 A9 E4 85 3C	/.130C 00 1B 48 29 7A	/.1380 20 A0 12 CA 58
/.1228 0A 18 65 D1 A6	/.129C D1 85 F3 60 C7	/.1310 0F B5 01 68 E3	/.1384 10 DE 60 EA 44
/.122C B5 D1 8A 0A D4	/.12A0 91 D1 A5 00 E5	/.1314 4A 4A 4A 4A 00	/.1388 EA EA EA EA 00
/.1230 0A 0A 18 65 7D	/.12A4 91 F3 A5 D1 16	/.1318 B5 02 CA BD F0	/.138C EA EA EA EA 00
/.1234 D1 85 D1 06 83	/.12A8 38 E9 16 85 42	/.131C 00 1B 48 29 7A	/.1390 A9 09 85 F7 D2
/.1238 D1 26 D2 A5 80	/.12AC D1 85 F3 B0 17	/.1320 0F 85 03 68 E1	/.1394 A9 16 85 F8 C2
/.123C D1 85 F3 A5 02	/.12B0 04 C6 D2 C6 D6	/.1324 4A 4A 4A 4A 00	/.1398 20 00 13 C6 F5
/.1240 D2 18 69 7B DB	/.12B4 F4 60 EA EA 94	/.1328 B5 04 A9 15 3D	/.139C F8 D0 F9 A9 78
/.1244 B5 F4 68 60 79	/.12B8 EA EA EA EA 00	/.132C 38 E5 02 85 5A	/.13A0 0A 85 F8 A9 DE
/.1248 20 20 12 B1 A3	/.12BC EA EA EA EA 00	/.1330 02 A9 15 38 B6	/.13A4 01 85 F9 A9 D4
/.124C D1 60 EA EA B1	/.12C0 20 00 12 48 7A	/.1334 E5 04 B5 04 60	/.13A8 00 85 FA A9 D6
/.1250 20 20 12 91 83	/.12C4 29 09 47 09 60	/.1338 A2 0B A5 01 0D	/.13AC 0F BD 0E 90 1C
/.1254 D1 A5 00 91 E5	/.12C8 F0 02 A9 F0 AB	/.133C C5 03 F0 0B 3E	/.13B0 A9 00 85 FB D7
/.1258 F3 60 EA EA 93	/.12CC 8D 0D 90 68 7B	/.1340 90 03 CA D0 89	/.13B4 B5 FC A9 01 D1
/.125C EA EA EA EA 00	/.12D0 60 EA EA EA BA	/.1344 03 E6 01 E8 0C	/.13B8 BD 14 15 8D 01
/.1260 A9 1E 85 02 30	/.12D4 EA EA EA EA 00	/.1348 B6 03 A2 07 20	/.13BC 5F 15 60 EA C0
/.1264 A9 16 85 01 3B	/.12D8 EA EA EA EA 00	/.134C A5 02 C5 04 66	/.13C0 EA EA EA EA 00
/.1268 85 03 A9 96 89	/.12DC EA EA EA EA 00	/.1350 F0 0B B0 03 4B	/.13C4 EA EA EA EA 00
/.126C 85 04 A2 02 21	/.12E0 EA EA EA EA 00	/.1354 CA D0 03 EB F1	/.13C8 EA EA EA EA 00
/.1270 A0 01 B1 01 11	/.12E4 EA EA EA EA 00	/.1358 C6 02 B6 04 46	/.13CC EA EA EA EA 00

programme

/.13D0 EA EA EA EA 00	/.14E0 EA EA EA EA 00	/.15F0 11 B9 80 11 39	/.1700 A6 FB D0 03 BE
/.13D4 EA EA EA EA 00	/.14E4 EA EA EA EA 00	/.15F4 9D 80 11 B9 B5	/.1704 4C 00 18 CA 9E
/.13DB EA EA EA EA 00	/.14EB EA EA EA EA 00	/.15F8 C0 11 9D C0 8C	/.1708 DE 40 11 10 9F
/.13DC EA EA EA EA 00	/.14EC EA EA EA EA 00	/.15FC 11 A5 FB 60 2F	/.170C 05 20 E0 15 D0
/.13E0 EA EA EA EA 00	/.14F0 EA EA EA EA 00	/.1600 A9 DF BD 0C F7	/.1710 F0 F2 CA 10 D8
/.13E4 EA EA EA EA 00	/.14F4 EA EA EA EA 00	/.1604 90 A9 FE 8D 4A	/.1714 F3 EA A5 FB 47
/.13EB EA EA EA EA 00	/.14F8 EA EA EA EA 00	/.1608 0D 90 A5 FA C2	/.1718 85 64 C6 64 43
/.13EC EA EA EA EA 00	/.14FC EA EA EA EA 00	/.160C 0A 3B E9 01 DA	/.171C A6 64 30 E4 16
/.13F0 EA EA EA EA 00	/.1500 20 00 13 A6 95	/.1610 B5 01 A6 FB DA	/.1720 BD 00 11 29 85
/.13F4 EA EA EA EA 00	/.1504 F8 A4 F9 20 85	/.1614 A4 F9 20 20 5D	/.1724 07 18 69 0E 78
/.13FB EA EA EA EA 00	/.1508 48 12 C9 20 B3	/.1618 12 98 3B E5 57	/.1728 85 61 BD 00 59
/.13FC EA EA EA EA 00	/.150C F0 01 60 A5 34	/.161C 01 AB C0 16 7F	/.172C 11 29 FB 85 45
/.1400 A9 08 BD 0F 23	/.1510 FA 91 D1 A9 13	/.1620 B0 08 B1 D1 DB	/.1730 65 BD 40 11 89
/.1404 90 A9 09 85 B5	/.1514 01 91 F3 20 43	/.1624 C9 20 F0 F1 EB	/.1734 85 62 BD 80 DA
/.1408 F7 A9 FF 8D 2C	/.1518 C0 12 4A B0 28	/.1628 D0 02 A9 00 7B	/.1738 11 85 63 BD 4A
/.140C 05 90 A9 07 3B	/.151C 01 C8 4A B0 33	/.162C 85 02 A2 30 15	/.173C C0 11 85 66 32
/.1410 85 00 A2 00 27	/.1520 01 CA 4A B0 31	/.1630 B4 03 A4 F9 DA	/.1740 EA A6 63 A4 8B
/.1414 A0 15 20 20 B5	/.1524 01 E8 4A B0 13	/.1634 98 3B E5 01 44	/.1744 62 20 20 12 70
/.1418 12 A9 A0 91 BA	/.1528 01 8B C0 FF B6	/.1638 AB C4 03 F0 9F	/.1748 A5 65 29 E0 09
/.141C D1 A9 07 91 EE	/.152C D0 01 C8 C0 D9	/.163C 15 AD 28 91 01	/.174C F0 CC A9 20 B5
/.1420 F3 8B 10 FB 90	/.1530 16 D0 01 BB 4F	/.1640 6D 04 90 29 D0	/.1750 91 D1 A6 64 82
/.1424 98 18 69 16 FF	/.1534 84 01 A4 F9 DB	/.1644 03 18 69 02 70	/.1754 A5 65 10 0A DA
/.1428 91 D1 C8 C0 48	/.1538 A9 20 91 D1 C9	/.1648 91 D1 A9 07 EE	/.1758 C6 62 10 06 B2
/.142C 04 D0 F5 A9 88	/.153C A4 01 A5 FA FA	/.164C 91 F3 18 90 EA	/.175C 20 E0 15 18 CD
/.1430 30 91 D1 C8 BB	/.1540 C4 F9 F0 08 C5	/.1650 E3 EA EE 0C EB	/.1760 90 B8 A5 65 EB
/.1434 C0 0A D0 F9 E3	/.1544 90 04 A9 00 3D	/.1654 90 CA D0 DA 50	/.1764 29 40 F0 07 9E
/.1438 A9 16 85 F8 C2	/.1548 F0 02 A9 01 5A	/.1658 A4 F9 98 38 FD	/.1768 A5 63 18 65 BB
/.143C 20 00 13 C6 F5	/.154C 85 FA 86 FB 01	/.165C E5 01 AB C4 88	/.176C 66 85 63 A5 25
/.1440 FB D0 F9 A9 78	/.1550 84 F9 20 48 15	/.1660 03 F0 06 A9 5C	/.1770 65 29 20 F0 9C
/.1444 0A 85 FB A9 DE	/.1554 12 C9 20 F0 0B	/.1664 20 91 D1 D0 B0	/.1774 0E A5 FB C5 96
/.1448 01 85 F9 A9 D4	/.1558 01 60 A5 FA 3E	/.1668 F1 A9 00 8D D5	/.1778 63 F0 08 90 0B
/.144C 00 85 FA A9 D6	/.155C 91 D1 A9 01 E8	/.166C 0C 90 8D 0D 1C	/.177C 04 E6 63 10 91
/.1450 0F 8D 0E 90 1C	/.1560 91 F3 20 C0 B2	/.1670 90 A5 02 C9 FE	/.1780 02 C6 63 A5 02
/.1454 A9 00 85 FB D7	/.1564 12 29 10 D0 FB	/.1674 0D B0 01 60 DC	/.1784 65 4A 4A EA BF
/.1458 85 FC A0 03 DA	/.1568 06 20 00 16 30	/.1678 D0 07 A9 20 5E	/.1788 29 04 85 00 AB
/.145C A9 1A 99 11 3B	/.156C 4C 77 15 A2 8C	/.167C 91 D1 4C E0 EC	/.178C E6 00 A6 63 23
/.1460 1E 8B D0 FB BE	/.1570 10 8B D0 FD B5	/.1680 16 C9 12 F0 3D	/.1790 A4 62 20 48 AE
/.1464 60 EA EA EA 8A	/.1574 CA D0 FA AD 4D	/.1684 F5 C9 11 D0 FD	/.1794 12 C9 20 F0 0B
/.1468 EA EA EA EA 00	/.1578 04 90 29 03 BE	/.1688 04 A9 12 D0 6F	/.1798 2F C9 02 B0 54
/.146C EA EA EA EA 00	/.157C F0 03 4C 9F 20	/.168C EF 38 E9 0E 30	/.179C 01 60 A5 65 A1
/.1470 EA EA EA EA 00	/.1580 15 AD 28 91 01	/.1690 AB B9 E0 17 E6	/.17A0 29 08 D0 04 F5
/.1474 EA EA EA EA 00	/.1584 29 1F C9 16 E9	/.1694 AB 20 E0 16 7E	/.17A4 A6 64 10 B4 66
/.1478 EA EA EA EA 00	/.1588 B0 F4 AA A0 4E	/.1698 8B D0 FA A6 04	/.17AB A6 64 BD C0 BF
/.147C EA EA EA EA 00	/.158C 15 20 48 12 6F	/.169C FB E6 03 CA D4	/.17AC 11 49 FF 18 BF
/.1480 20 00 15 A9 9C	/.1590 C9 20 D0 EA D3	/.16A0 30 21 BD 40 EC	/.17B0 69 01 9D C0 35
/.1484 80 8D 0D 90 90	/.1594 A9 0D 91 D1 E4	/.16A4 11 C5 03 D0 07	/.17B4 11 BD 40 11 FD
/.1488 A6 FB A4 F9 03	/.1598 A9 04 91 F3 CF	/.16A8 F6 BD 80 11 DA	/.17B8 85 62 BD 80 DA
/.148C 20 20 12 A9 8B	/.159C 4C 00 17 AD F6	/.16AC C5 FB D0 EF 02	/.17BC 11 85 63 A6 51
/.1490 01 91 F3 A2 C1	/.15A0 04 90 49 FF 22	/.16B0 BD 00 11 29 85	/.17C0 63 EA EA A4 C7
/.1494 1B BA E0 21 50	/.15A4 29 3F D0 F4 32	/.16B4 10 F0 09 A9 40	/.17C4 62 20 20 12 70
/.1498 B0 02 91 D1 F2	/.15A8 A2 01 A0 15 16	/.16B8 EF 3D 00 11 C3	/.17C8 A5 61 91 D1 84
/.149C A5 A2 29 07 29	/.15AC A9 04 85 00 28	/.16BC 9D 00 11 60 EC	/.17CC A5 00 91 F3 C7
/.14A0 D0 FA A5 A2 2D	/.15B0 20 4B 12 C9 B3	/.16C0 20 E0 15 A6 73	/.17D0 A6 64 A5 62 05
/.14A4 29 07 F0 FA 24	/.15B4 20 F0 03 EB 3B	/.16C4 FB A4 03 8B D7	/.17D4 9D 40 11 A5 69
/.14A8 EB CE 0E 90 8B	/.15B8 D0 F6 A9 11 9E	/.16C8 20 20 12 A9 8B	/.17D8 63 9D 80 11 6F
/.14AC 10 E7 A9 00 5E	/.15BC 91 D1 A9 04 ED	/.16CC 20 91 D1 60 00	/.17DC 4C 1A 17 EA AB
/.14B0 8D 0D 90 A0 B0	/.15C0 91 F3 EB 20 AA	/.16D0 EA EA EA EA 00	/.17E0 01 05 0A 00 0E
/.14B4 04 B9 11 1E B2	/.15C4 48 12 C9 20 B3	/.16D4 EA EA EA EA 00	/.17E4 00 14 32 64 42
/.14B8 C9 A0 D0 04 BD	/.15C8 F0 F0 4C 00 4C	/.16D8 EA EA EA EA 00	/.17EB 82 DF 08 A9 FC
/.14BC 8B D0 F6 60 CE	/.15CC 17 EA EA EA FD	/.16DC EA EA EA EA 00	/.17EC 00 21 E1 CB 0B
/.14C0 A9 A0 99 11 81	/.15D0 EA EA EA EA 00	/.16E0 A2 04 FE 04 5C	/.17F0 01 01 01 01 00
/.14C4 1E 20 90 13 BD	/.15D4 EA EA EA EA 00	/.16E4 1E BD 04 1E B9	/.17F4 01 4A 4A 4A 4B
/.14C8 4C 80 14 EA 32	/.15DB EA EA EA EA 00	/.16EB C9 3A F0 01 02	/.17FB 4A AD AD AD E7
/.14CC EA EA EA EA 00	/.15DC EA EA EA EA 00	/.16EC 60 A9 30 9D 64	/.17FC 3E 3E 3E 5F 61
/.14D0 EA EA EA EA 00	/.15E0 C6 FB A4 FB 62	/.16F0 04 1E CA 10 CQ	/.1800 AD 18 91 29 0D
/.14D4 EA EA EA EA 00	/.15E4 EA B9 00 11 42	/.16F4 ED 60 EA EA BD	/.1804 0F F0 03 4C 80
/.14DB EA EA EA EA 00	/.15EB 9D 00 11 B9 35	/.16FB EA EA EA EA 00	/.1808 00 19 AD 04 80
/.14DC EA EA EA EA 00	/.15EC 40 11 9D 40 8C	/.16FC EA EA EA EA 00	/.180C 90 29 1F C9 6F

programme

/.1B10	16	B0	F4	AA	FB	/.1920	61	BD	40	11	8D	/.1A30	E0	0E	D0	DB	E6	/.1B40	27	38	39	49	6F
/.1B14	A0	15	20	48	DD	/.1924	38	E9	01	99	49	/.1A34	78	AD	0F	90	4A	/.1B44	58	67	77	88	CO
/.1B18	12	C9	20	D0	2B	/.1928	40	11	30	29	48	/.1A38	18	29	17	69	4F	/.1B48	89	79	69	58	C1
/.1B1C	EA	8A	A6	FB	3D	/.192C	AB	A6	61	20	4F	/.1A3C	11	09	08	8D	9D	/.1B4C	48	49	59	68	30
/.1B20	E0	3F	B0	E3	BC	/.1930	48	12	C9	20	B3	/.1A40	0F	90	A0	B2	8D	/.1B50	67	57	46	36	40
/.1B24	9D	B0	11	98	94	/.1934	D0	1F	A4	FB	90	/.1A44	88	D0	FD	AD	08	/.1B54	35	34	23	12	30
/.1B28	9D	40	11	AD	61	/.1938	A9	C0	99	00	F0	/.1A48	1F	91	29	20	B7	/.1B58	01	00	00	00	01
/.1B2C	14	91	29	0F	A3	/.193C	11	8A	A2	00	39	/.1A4C	D0	E7	58	A9	C6	/.1B5C	00	00	00	00	00
/.1B30	AB	B9	F0	17	F6	/.1940	C5	FB	F0	06	CB	/.1A50	0F	8D	04	18	9E	/.1B60	00	10	10	00	00
/.1B34	9D	00	11	A9	25	/.1944	90	03	CA	D0	89	/.1A54	A9	01	8D	61	44	/.1B64	01	02	12	12	03
/.1B38	FF	9D	C0	11	B3	/.1948	01	EB	8A	99	FA	/.1A58	13	8D	14	15	9F	/.1B68	22	32	42	51	03
/.1B3C	E6	FB	BD	00	A0	/.194C	C0	11	E6	FB	CC	/.1A5C	8D	5F	15	20	E7	/.1B6C	60	60	51	42	13
/.1B40	11	29	07	18	27	/.1950	A9	7F	99	41	0E	/.1A60	00	14	20	80	B4	/.1B70	33	23	14	05	01
/.1B44	69	0E	4B	BD	92	/.1954	11	AD	1F	91	32	/.1A64	14	A0	00	B9	0D	/.1B74	06	07	08	08	01
/.1B48	00	11	4A	4A	11	/.1958	29	20	D0	04	DD	/.1A68	0B	1E	D9	04	CB	/.1B78	17	26	25	34	20
/.1B4C	EA	29	04	18	DF	/.195C	E6	FC	D0	06	CC	/.1A6C	1E	90	09	D0	57	/.1B7C	43	53	53	42	01
/.1B50	69	01	4B	BD	9D	/.1960	A5	FC	F0	02	AB	/.1A70	12	C8	C0	05	1F	/.1B80	31	21	11	00	01
/.1B54	40	11	A8	BD	44	/.1964	C6	FC	A5	FC	63	/.1A74	D0	F1	F0	0B	DA	/.1B84	11	22	33	44	44
/.1B58	B0	11	AA	20	1B	/.1968	C9	10	B0	04	6D	/.1A78	A0	06	B9	04	1B	/.1B88	44	43	42	41	04
/.1B5C	20	12	6B	91	CB	/.196C	A9	01	D0	02	7A	/.1A7C	1E	99	0B	1E	92	/.1B8C	40	40	40	40	00
/.1B60	F3	68	91	D1	DB	/.1970	A9	04	BD	14	34	/.1A80	88	10	F7	AD	C2	/.1B90	50	60	70	80	C0
/.1B64	A9	B0	A6	FB	74	/.1974	15	8D	5F	15	D2	/.1A84	8E	02	F0	FB	B7	/.1B94	81	71	62	52	C0
/.1B68	9D	40	11	4C	80	/.1978	A5	A2	29	07	29	/.1A88	4C	4F	1A	EA	F3	/.1B98	43	44	45	46	04
/.1B6C	00	19	EA	EA	19	/.197C	D0	FA	A5	FC	73	/.1A8C	EA	EA	EA	EA	00	/.1B9C	46	45	34	23	14
/.1B70	EA	EA	EA	EA	00	/.1980	C9	18	D0	01	00	/.1A90	EA	EA	EA	EA	00	/.1BA0	23	24	15	16	04
/.1B74	EA	EA	EA	EA	00	/.1984	60	A5	F7	F0	C2	/.1A94	EA	EA	EA	EA	00	/.1BA4	07	08	09	0A	0C
/.1B78	EA	EA	EA	EA	00	/.1988	03	4C	00	15	5A	/.1A98	EA	EA	EA	EA	00	/.1BA8	0B	0A	09	08	00
/.1B7C	EA	EA	EA	EA	00	/.198C	4E	04	18	AE	FC	/.1A9C	EA	EA	EA	EA	00	/.1BAC	18	28	37	47	40
/.1B80	EA	EA	EA	EA	00	/.1990	61	13	EB	EB	72	/.1AA0	EA	EA	EA	EA	00	/.1BB0	56	65	74	73	34
/.1B84	EA	EA	EA	EA	00	/.1994	8A	29	07	BD	29	/.1AA4	EA	EA	EA	EA	00	/.1BB4	62	52	42	42	30
/.1B88	EA	EA	EA	EA	00	/.1998	61	13	A2	01	D1	/.1AA8	EA	EA	EA	EA	00	/.1BB8	51	61	71	81	C0
/.1B8C	EA	EA	EA	EA	00	/.199C	FE	04	1E	BD	59	/.1AAC	EA	EA	EA	EA	00	/.1BBC	81	71	70	80	00
/.1B90	EA	EA	EA	EA	00	/.19A0	04	1E	C9	3A	E9	/.1AB0	EA	EA	EA	EA	00	/.1BC0	90	A0	B0	C0	40
/.1B94	EA	EA	EA	EA	00	/.19A4	D0	08	A9	30	41	/.1AB4	EA	EA	EA	EA	00	/.1BC4	D0	E0	F0	F0	30
/.1B98	EA	EA	EA	EA	00	/.19AC	9D	04	1E	CA	4D	/.1AB8	EA	EA	EA	EA	00	/.1BC8	F0	E0	D0	C0	00
/.1B9C	EA	EA	EA	EA	00	/.19A8	10	EE	A0	00	5E	/.1ABC	EA	EA	EA	EA	00	/.1BCC	B0	A0	90	80	00
/.1BA0	EA	EA	EA	EA	00	/.19B0	B9	12	1E	C9	7C	/.1AC0	EA	EA	EA	EA	00	/.1BD0	70	61	51	41	01
/.1BA4	EA	EA	EA	EA	00	/.19B4	1A	D0	08	CB	0A	/.1AC4	EA	EA	EA	EA	00	/.1BD4	32	32	33	24	17
/.1BA8	EA	EA	EA	EA	00	/.19B8	C0	04	D0	F4	E0	/.1AC8	EA	EA	EA	EA	00	/.1BDB	15	16	17	18	0C
/.1BAC	EA	EA	EA	EA	00	/.19BC	4C	00	15	A9	F0	/.1ACC	EA	EA	EA	EA	00	/.1BDC	0B	07	06	05	0C
/.1BB0	EA	EA	EA	EA	00	/.19C0	1A	99	12	1E	BF	/.1AD0	EA	EA	EA	EA	00	/.1BE0	05	06	07	08	0C
/.1BB4	EA	EA	EA	EA	00	/.19C4	4C	00	15	EA	B3	/.1AD4	EA	EA	EA	EA	00	/.1BE4	09	1A	2A	39	00
/.1BB8	EA	EA	EA	EA	00	/.19C8	EA	EA	EA	EA	00	/.1ADB	EA	EA	EA	EA	00	/.1BEB	38	37	26	25	0C
/.1BBC	EA	EA	EA	EA	00	/.19CC	EA	EA	EA	EA	00	/.1ADC	EA	EA	EA	EA	00	/.1BEC	35	45	54	63	47
/.1BC0	EA	EA	EA	EA	00	/.19D0	EA	EA	EA	EA	00	/.1AE0	EA	EA	EA	EA	00	/.1BF0	72	B1	71	61	E3
/.1BC4	EA	EA	EA	EA	00	/.19D4	EA	EA	EA	EA	00	/.1AE4	EA	EA	EA	EA	00	/.1BF4	62	73	83	92	00
/.1BC8	EA	EA	EA	EA	00	/.19D8	EA	EA	EA	EA	00	/.1AEB	EA	EA	EA	EA	00	/.1BF8	A1	A0	90	80	11
/.1BCC	EA	EA	EA	EA	00	/.19DC	EA	EA	EA	EA	00	/.1AEC	EA	EA	EA	EA	00	/.1BFC	70	70	70	80	F0
/.1BD0	EA	EA	EA	EA	00	/.19E0	07	23	24	25	25	/.1AF0	EA	EA	EA	EA	00	/.1C00	F8	60	98	F7	F7
/.1BD4	EA	EA	EA	EA	00	/.19E4	24	26	23	24	05	/.1AF4	EA	EA	EA	EA	00	/.1C04	F7	98	60	F8	F7
/.1BD8	EA	EA	EA	EA	00	/.19E8	27	00	0A	2B	05	/.1AF8	EA	EA	EA	EA	00	/.1C08	1F	06	19	EF	EF
/.1BDC	EA	EA	EA	EA	00	/.19EC	29	00	05	2A	06	/.1AFC	EA	EA	EA	EA	00	/.1C0C	EF	19	06	1F	EF
/.1BE0	EA	EA	EA	EA	00	/.19F0	24	2B	2C	2D	0E	/.1B00	81	72	63	53	C3	/.1C10	00	00	00	A9	A9
/.1BE4	EA	EA	EA	EA	00	/.19F4	2B	20	20	2A	01	/.1B04	42	32	31	40	01	/.1C14	CB	00	00	00	CB
/.1BE8	EA	EA	EA	EA	00	/.19F8	2E	25	2F	00	24	/.1B08	50	60	70	80	C0	/.1C18	00	00	00	5D	5D
/.1BEC	EA	EA	EA	EA	00	/.19FC	AA	AA	AA	AA	00	/.1B0C	90	A0	B0	C0	40	/.1C1C	B6	00	00	00	B6
/.1BF0	EA	EA	EA	EA	00	/.1A00	20	5F	E5	20	BA	/.1B10	D0	E0	D0	D1	31	/.1C20	00	00	00	DA	DA
/.1BF4	EA	EA	EA	EA	00	/.1A04	00	14	A9	00	BD	/.1B14	D2	D3	D3	C3	11	/.1C24	6C	00	00	00	6C
/.1BF8	EA	EA	EA	EA	00	/.1A08	85	61	A2	0B	4E	/.1B18	B3	A3	94	85	01	/.1C28	00	00	00	45	45
/.1BFC	EA	EA	EA	EA	00	/.1A0C	86	62	A0	00	44	/.1B1C	76	67	58	48	01	/.1C2C	A4	00	00	00	A4
/.1900	A4	FB	F0	51	FE	/.1A10	20	20	12	A6	B4	/.1B20	37	26	16	05	02	/.1C30	01	01	03	03	00
/.1904	C0	3E	B0	4D	03	/.1A14	61	BD	E0	19	25	/.1B24	04	13	23	22	16	/.1C34	07	0F	3F	FF	CB
/.1908	AD	24	91	29	31	/.1A18	AB	EB	BD	E0	1D	/.1B28	31	41	50	50	70	/.1C38	81	C3	FF	FF	42
/.190C	07	AA	E4	FB	B2	/.1A1C	19	F0	0A	91	72	/.1B2C	40	40	30	20	10	/.1C3C	FF	FF	FF	FF	00
/.1910	B0	43	BD	00	4E	/.1A20	D1	A9	00	91	E9	/.1B30	10	00	00	00	10	/.1C40	80	B0	C0	C0	00
/.1914	11	C9	C0	F0	EB	/.1A24	F3	CB	EB	D0	03	/.1B34	11	22	33	34	34	/.1C44	E0	F0	FC	FF	13
/.1918	3C	BD	B0	11	10	/.1A28	F1	EB	B6	61	FE	/.1B38	35	25	14	04	00	/.1C48	FF	FF	FF	FF	00
/.191C	99	B0	11	85	BD	/.1A2C	A6	62	EB	EB	C4	/.1B3C	05	05	16	16	00	/.1C4C	FF	FF	FF	FF	00

/.1C50 FF 3F 0F 07 C8	/.1CD0 C0 08 08 55 55	/.1D50 7C FE C0 FC BE	/.1DD0 00 00 08 20 28
/.1C54 03 03 01 01 00	/.1CD4 5D 6B 6B 5D 00	/.1D54 7E 06 FE 7C FA	/.1DD4 00 08 00 00 08
/.1C58 FF FF FF FF 00	/.1CD8 00 00 18 3C 24	/.1D58 18 18 00 18 18	/.1DD8 00 00 08 00 08
/.1C5C FF FF C3 B1 42	/.1CDE 3C 18 00 00 24	/.1D5C 18 18 18 18 00	/.1DDC 00 08 08 10 10
/.1C60 FF FC F0 E0 13	/.1CE0 00 42 3C 3C 42	/.1D60 7C FE C6 C0 84	/.1DE0 0E 18 30 60 46
/.1C64 C0 C0 80 80 00	/.1CE4 3C 3C 42 00 42	/.1D64 C0 C6 FE 7C 84	/.1DE4 30 18 0E 00 26
/.1C68 00 00 00 1C 1C	/.1CE8 B2 00 38 1C A6	/.1D68 C6 C6 C6 FE 38	/.1DE8 00 20 7E 00 5E
/.1C6C 08 1C 00 00 14	/.1CEC 3C 30 41 00 4D	/.1D6C FE C6 C6 C6 38	/.1DEC 7E 00 00 00 7E
/.1C70 00 00 00 18 18	/.1CF0 01 40 00 28 69	/.1D70 7C FE C6 C6 82	/.1DF0 FD 9B BB 3F E2
/.1C74 18 00 00 00 18	/.1CF4 04 10 00 80 94	/.1D74 C6 C6 FE 7C 82	/.1DF4 F7 57 6F 37 F8
/.1C78 00 18 24 66 5A	/.1CF8 00 00 20 04 24	/.1D78 FC FC 30 20 10	/.1DF8 BF 87 DF F7 10
/.1C7C BD 42 3C 00 C3	/.1CFC 08 00 00 80 88	/.1D7C 30 30 30 30 00	/.1DFC DF 27 93 78 13
/.1C80 3C 7E DB FF 66	/.1D00 00 00 00 00 00	/.1D80 00 3C 66 66 3C	/?
/.1C84 66 C3 42 24 C3	/.1D04 00 00 00 00 00	/.1D84 66 66 3C 00 3C	
/.1C88 20 70 F8 74 DC	/.1D08 08 08 08 08 00	/.1D88 00 18 38 78 58	
/.1C8C 2E 1F 0E 04 3B	/.1D0C 00 00 08 00 08	/.1D8C 18 18 7E 00 7E	
/.1C90 00 00 00 04 04	/.1D10 24 24 24 00 24	/.1D90 00 3C 66 0C 56	
/.1C94 0E 1F 0E 04 1B	/.1D14 00 00 00 00 00	/.1D94 38 60 7E 00 26	
/.1C98 00 00 7E AB D5	/.1D18 FB FC C6 C6 04	/.1D98 00 3C 66 0C 56	
/.1C9C D5 7E 00 00 AB	/.1D1C C6 C6 FC F8 04	/.1D9C 06 66 3C 00 5C	
/.1CA0 00 7E 18 24 42	/.1D20 FE FE C0 FE 38	/.1DA0 00 0C 1C 2C 3C	
/.1CA4 DB 24 18 7E 99	/.1D24 FB C0 FE FE 38	/.1DA4 7E 0C 0C 00 7E	
/.1CAB 20 60 FC 7C C0	/.1D28 FE FE C0 F8 38	/.1DAB 00 7E 60 7C 62	
/.1CAC 3C 1F 0E 04 29	/.1D2C FB C0 C0 C0 38	/.1DAC 06 66 3C 00 5C	
/.1CB0 00 78 EC E0 74	/.1D30 C6 E6 F6 FE 28	/.1DB0 00 38 60 7C 24	
/.1CB4 79 1D DD 78 C1	/.1D34 FE DE CE C6 28	/.1DB4 66 66 3C 00 3C	
/.1CB8 00 00 00 E3 E3	/.1D38 FC FE C6 FE 3A	/.1DB8 00 7E 66 0C 14	
/.1CBC B6 86 B6 E3 65	/.1D3C FC CC C6 C6 30	/.1DBC 18 18 18 00 18	
/.1CC0 00 00 00 96 96	/.1D40 FC FE C6 FC 38	/.1DC0 00 3C 66 3C 66	
/.1CC4 DA DB D8 98 42	/.1D44 FC C6 FE FC 38	/.1DC4 66 66 3C 00 3C	
/.1CC8 00 00 03 73 70	/.1D48 C6 EE 7C 38 6C	/.1DC8 00 3C 66 3E 64	
/.1CCC DB FB C3 70 90	/.1D4C 30 60 E0 C0 70	/.1DCC 06 0C 38 00 32	

**Tippen Sie
zuerst das
Monitor-Ein-
gabeprogramm
ab, damit Sie
das Datazeilen-
Programm
eingeben kön-
nen!**

```

100 FORI=4224TD4607STEP16:CS=0
110 FORJ=OTD15:READA:CS=CS+A:POKEI+J,A:NEXT
120 READA:IFCS=ATHENNEXT:END
130 PRINT"DATA ERROR IN "1000+(I-4224)/16*10:END
1000 DATA201,63,8,41,15,40,144,2,105,8,96,32,207,255,32,128, 1377
1010 DATA16,10,10,10,10,133,255,32,207,255,32,128,16,5,255,96, 1470
1020 DATA32,139,16,133,250,32,139,16,133,249,96,72,74,74,74, 1603
1030 DATA32,182,16,104,41,15,24,105,246,144,2,105,6,105,58,76, 1261
1040 DATA210,255,165,248,32,171,16,165,247,76,171,16,234,234,234,234, 2708
1050 DATA169,47,32,210,255,32,207,255,201,47,240,249,201,46,208,3, 2402
1060 DATA76,128,17,201,77,240,13,169,63,32,210,255,169,13,32,210, 1905
1070 DATA255,76,208,16,32,0,17,76,208,16,234,234,234,234,234,234, 2308
1080 DATA32,207,255,32,160,16,165,249,133,247,165,250,133,248,32,207, 2531
1090 DATA255,201,13,240,3,32,160,16,169,13,32,210,255,169,47,32, 1847
1100 DATA210,255,169,46,32,210,255,32,194,16,169,32,32,210,255,160, 2277
1110 DATA0,132,251,177,247,69,251,133,251,177,247,32,171,16,169,32, 2355
1120 DATA32,210,255,200,192,4,208,235,165,251,32,171,16,169,13,32, 2185
1130 DATA210,255,165,247,24,105,4,133,247,144,2,230,248,165,248,197, 2624
1140 DATA250,144,186,208,6,165,249,197,247,176,178,96,234,234,234,234, 3038
1150 DATA32,160,16,165,249,133,247,165,250,133,248,96,234,234,234,234, 2830
1160 DATA32,112,17,160,0,132,251,32,207,255,32,139,16,145,247,69, 1846
1170 DATA251,133,251,200,192,4,208,239,32,207,255,32,139,16,197,251, 2607
1180 DATA240,13,169,145,32,210,255,169,13,32,210,255,76,208,17,169, 2213
1190 DATA13,32,210,255,165,247,24,105,4,133,247,144,2,230,248,165, 2224
1200 DATA247,133,249,165,248,133,250,32,29,17,169,145,32,210,255,234, 2548
1210 DATA160,6,132,198,169,29,153,119,2,136,16,250,76,208,16,234, 1904
1220 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234, 3744
1230 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234, 3744

```

MEMORY-DISPLAY - ein kurzes Maschinenprogramm für den VC-20, mit dem man sich auf Tastendruck den aktuellen Stand der Speicherbelegung ansehen kann.

Der VC-20 ist mit Speicherkapazität nicht gerade gesegnet und selbst wenn man sein Gedächtnis mit einer Speichererweiterung um einiges ausgebaut hat, ist es bei größeren Programmiervorhaben in Basic unerlässlich, den verbleibenden Speicherraum ständig zu kontrollieren. Mit der FRE-Funktion ist dies zwar möglich, doch ist es zugleich recht aufwendig, jedesmal die Anweisung neu einzugeben. Hier schafft die vorliegende Routine Abhilfe. Auf Tastendruck wird angezeigt, wieviel Speicherraum Basicprogramm, numerische Variablen, Arrays und Stringvariablen jeweils belegen, so daß man genau abschätzen kann, was noch im Speicher Platz hat.

Eingabehinweise:

Geben Sie den abgedruckten Basic-Loader ein und speichern Sie ihn zur Sicherheit ab. Die REM-Anweisungen können dabei weggelassen werden, da sie für den Programmablauf völlig irrelevant sind. Im Listing tragen sie sehr zur Übersichtlichkeit bei, in eingegebener Form verwirren sie eher. Da das Programm auf einem VC-20 mit 16kByte-Erweiterung entstand, ist es möglich, daß Anwender mit kleinerer oder ganz ohne Erweiterung die REM-Anweisungen sogar weglassen müssen. Davon unabhängig ist die eigentliche Maschinenroutine, die relativ kurz ist und in jeder Ausbaustufe lauffähig ist.

Nach dem Start mit RUN dauert es ein wenig, bis die DATA-Werte eingelesen sind. Dann meldet das Programm entweder einen Prüfsummenfehler in den Datenzeilen, Sie müssen dann die entsprechenden Zeilen auf Fehler untersuchen, oder Sie haben alles richtig eingegeben, dann erscheint die Abfrage „Abspeichern (J/N)“. Kassettenanwender müssen hier leider nein (N) eingeben. Da die Routine im

Kassettenpuffer angesiedelt ist, damit sie in allen Ausbaustufen läuft, kann sie nicht ohne weiteres auf Band abgespeichert werden. Kassettenanwender müssen also jedesmal den Basic-Loader laden und starten, um mit der Routine arbeiten zu können. Auch jede Kassettenoperation zerstört das Programm; dies hat, ist die Routine aktiviert, fatale Folgen, da der IRO-Vektor auf die Routine weist (dies ist zur Abfrage der Taste notwendig). Der Computer hängt sich nämlich auf und kann nur durch Aus-/Anschalten oder einen RESET wieder belebt werden. Vor jeder Kassettenoperation muß die Routine also deaktiviert werden; dies geschieht einfach durch gleichzeitiges Drücken der RUN/STOP- und der RESTORE-Taste auf Ihrem VC-20. Besser haben es da die Diskettenanwender. Diskettenoperationen greifen nicht auf den Kassettenpuffer zu, folglich wird das Programm davon nicht beeinflusst und kann sogar als reine Maschinenroutine auf Diskette abgespeichert werden. In der Form belegt es nur einen Block und das lästige Einlesen der Daten entfällt dann bei späterem Gebrauch. Geben Sie dazu einfach ja (J) ein, nachdem Sie eine Diskette eingelegt haben. Die Routine wird dann automatisch unter dem Namen „MEMDISPLAY.CODE“ abgespeichert. Eingeladen werden muß die Routine dann absolut, also mit LOAD „MEMDISPLAY.CODE“,8,1. Nach dem Laden muß ein NEW eingegeben werden, um alle Basic-Pointer zu normalisieren.

Haben Sie die Routine dann ordnungsgemäß im Speicher, wird sie mit SYS828 initialisiert. Abhängig davon, was zufällig gerade in Speicherstelle 2, die als Puffer genutzt wird, steht, kann es einige Sekunden dauern, bis das Programm zum ersten Mal reagiert. Ist die Speicherstelle jedoch erst einmal heruntergezählt, genügt ein

Druck auf die F1-Taste und augenblicklich wird die Speicherverteilung angezeigt. Aus programmtechnischen Gründen erscheint dazu zuerst ein SYS873 auf dem Bildschirm, was Sie aber nicht zu kümmern braucht. Darunter dann in der nächsten Zeile: „BAS: xxxx-yyyy“. xxxx-yyyy gibt dabei dann den Speicherbereich an, der vom Basic-Programm belegt wird. Nach demselben Schema werden in den folgenden Zeilen die Speicherbereiche für numerische Variablen (VAR), Arrays (ARR) und Strings (STR) angezeigt. Beim VC-20 sind die einzelnen Speicherbereiche so angelegt, daß BAS, VAR und ARR in dieser Reihenfolge am Beginn des RAM-Speichers liegen, während STR am Ende angesiedelt ist. Folglich ergibt sich aus dem Beginn des STR-Bereichs und dem Ende des ARR-Bereichs der verbleibende Speicherraum.

Für Interessierte ist ein kommentiertes Assemblerlisting in den Basic-Loader integriert. Die F1-Tasten-Abfrage ist in der Routine durch Erweiterung der IRQ-Routine realisiert. Ist F1 gedrückt, wird einfach SYS873 in den Tastaturpuffer geschrieben. Dadurch wird die eigentliche Ausgaberroutine gestartet, die für ein Einbinden in die IRQ-Routine einfach zu lang ist.

Wichtig:

VC-20 Anwender ohne Speichererweiterung müssen die REM-Zeilen des Programms weglassen, da sonst der Speicherbereich nicht ausreicht!


```

0 REM *****
1 REM *
2 REM *      M E M O R Y - D I S P L A Y      F U E R   D E N   C O M M O D O R E   V C 2 0      *
3 REM *
4 REM *      B E L E G T   W I R D   D E R   S P E I C H E R B E R E I C H   V O M   8 2 8   B I S   9 7 9      *
5 REM *      I N I T I A L I S I E R U N G   E R F O L G T   D U R C H :      *
6 REM *      S Y S 8 2 8      *
7 REM *
8 REM *      W R I T T E N   1 9 8 5   B Y   W A L D E M A R   R A A Z      *
9 REM *****
10 :
11 :
10000 DATA 78      :REM SEI      ; INTERRUPT SPERREN
10002 DATA A9,49   :REM LDA #$49   ; INTERRUPTVEKTOR AUF EIGENE
10004 DATA A0,03   :REM LDY #$03   ; ROUTINE UMBIEGEN
10006 DATA 8D,14,03 :REM STA $0314  ;
10008 DATA 8C,15,03 :REM STY $0315  ;
10010 DATA 58      :REM CLI      ; INTERRUPT WIEDER FREIGEBEN
10012 DATA 60      :REM RTS      ; ZURUECK IN DIE BASIC-BEFEHLS EBENE
10014 DATA C6,FB   :REM DEC $FB   ; PUFFERSPEICHER ERNIEDRIGEN
10016 DATA F0,03   :REM BEQ $0350 ; SCHON LEER ??
10018 DATA 4C,BF,EA :REM JMP $EABF ; NEIN, DANN NORMALE INTERRUPTSCHLEIFE
10020 DATA A9,0A   :REM LDA #$0A   ; JA, DANN WIEDER AUFFUELLEN
10022 DATA B5,FB   :REM STA $FB   ;
10024 DATA A5,CB   :REM LDA $CB   ; F1-TASTE GEDRUECKT ??
10026 DATA C9,27   :REM CMP #$27   ;
10028 DATA D0,F3   :REM BNE $034D  ; NEIN, DANN NORMALE INTERRUPTSCHLEIFE
10030 DATA A2,06   :REM LDX #$06   ; JA, DANN 'SYS873' + RETURN IN DEN
10032 DATA B6,C6   :REM STX $C6   ; TASTATURSPEICHER SCHREIBEN UND
10034 DATA BD,A9,03 :REM LDA $03A9,X ; POINTER ENTSPRECHEND SETZEN
10036 DATA 9D,76,02 :REM STA $0276,X ;
10038 DATA CA      :REM DEX      ;
10040 DATA D0,F7   :REM BNE $035E  ;
10042 DATA F0,E4   :REM BEQ $034D  ; ZUR NORMALEN INTERRUPTSCHLEIFE
10044 DATA A2,00   :REM LDX #$00   ; BEGINN DER EIGENTLICHEN ROUTINE ZUR
10046 DATA B6,FC   :REM STX $FC   ; SPEICHERBELEGUNGSANZEIGE
10048 DATA BD,B0,03 :REM LDA $03B0,X ; ZEIGER AUF NAMEN DES SPEICHERBEREICHS
10050 DATA BC,B1,03 :REM LDY $03B1,X ; LADEN
10052 DATA 20,1E,CB :REM JSR $CB1E  ; UND NAMEN AUSGEBEN
10054 DATA A9,20   :REM LDA #$20   ; LEERZEICHEN AUSGEBEN
10056 DATA 20,D2,FF :REM JSR $FFD2  ;
10058 DATA A6,FC   :REM LDX $FC   ;
10060 DATA BD,B2,03 :REM LDA $03B2,X ; ZEIGER AUF BEGINN DES SPEICHERBEREICHS
10062 DATA AB      :REM TAY      ; LADEN
10064 DATA B6,01   :REM LDX $01,Y ; ZAHLENWERT> LADEN
10066 DATA BA      :REM TXA      ;
10068 DATA B6,00   :REM LDX $00,Y ; ZAHLENWERT< LADEN
10070 DATA 20,CD,DD :REM JSR $DDCD  ; UND AUSGEBEN
10072 DATA A9,2D   :REM LDA #$2D   ; BINDESTRICH AUSGEBEN
10074 DATA 20,D2,FF :REM JSR $FFD2  ;
10076 DATA A6,FC   :REM LDX $FC   ; ZEIGER AUF ENDE DES SPEICHERBEREICHS
10078 DATA BD,B3,03 :REM LDA $03B3,X ; LADEN
10080 DATA AB      :REM TAY      ;
10082 DATA B6,01   :REM LDX $01,Y ; ZAHLENWERT> LADEN
10084 DATA BA      :REM TXA      ;
10086 DATA B6,00   :REM LDX $00,Y ; ZAHLENWERT< LADEN
10088 DATA 20,CD,DD :REM JSR $DDCD  ; UND AUSGEBEN
10090 DATA 20,D7,CA :REM JSR $CAD7  ; 'CR' UND 'LF' AUSGEBEN
10092 DATA A6,FC   :REM LDX $FC   ; ZAEHLREGISTER HOCHZAEHLEN
10094 DATA EB      :REM INX      ;
10096 DATA EB      :REM INX      ;
10098 DATA EB      :REM INX      ;

```

```

10100 DATA EB      :REM INX      ;
10102 DATA EO,10   :REM CPX  #$10   ; ALLE SPEICHERBEREICHE AUSGEGEBEN ??
10104 DATA 90,C2   :REM BCC  $036B  ; NEIN, DANN DEN NAECHSTEN
10106 DATA 60      :REM RTS      ; JA, DANN FERTIG UND ZURUECK INS BASIC
10108 DATA 53,D9,38,37,33,0D :REM .BY SY873.
10110 DATA C0,03,2B,2D,C5,03,2D,2F :REM
10112 DATA CA,03,2F,31,CF,03,33,37 :REM
10114 DATA 42,41,53,3A,00,56,41,52 :REM .BY BAS:.VAR
10116 DATA 3A,00,41,52,52,3A,00,53 :REM .BY :.ARR:.-S
10118 DATA 54,52,3A,00 :REM .BY TR:..

10200 :
10210 : RESTORE : P=0
10220 : FOR I = 828 TO 979
10230 : READ A$
10240 : GOSUB 10500
10250 : P = P + A
10260 : POKE I , A
10270 : NEXT I
10280 :
10290 : IF P = 18201 THEN 10400
10299 :
10300 PRINT"LIHNEN IST BEIM ABTIP-";
10305 PRINT"PEN DER DATAZEILEN EIN";
10310 PRINT"FEHLER UNTERLAUFEN!"
10315 PRINT"UEBERPRUEFEN SIE BITTE";
10320 PRINT"NOCH EINMAL DIE ENT-";
10325 PRINT"SPRECHENDEN PROGRAMM-";
10330 PRINT"ZEILEN!":END
10399 :
10400 PRINT"LSIE HABEN ALLE DATA-";
10405 PRINT"WERTE RICHTIG EINGEGE-";
10410 PRINT"BEN. WENN SIE MIT DIS-";
10415 PRINT"KETTE ARBEITEN,KOENNEN";

10420 PRINT"SIE DAS PROGRAMM ALS";
10425 PRINT"REINE MASCHINENROUTINE";
10430 PRINT"ABSPEICHERN. ES BELEGT";
10435 PRINT"IN DER FORM NUR EINEN";
10440 PRINT"BLOCK. FUER KASSETTEN-";
10445 PRINT"ANWENDER BESTEHT DIESE";
10450 PRINT"MOEGLICHKEIT NICHT!"
10455 PRINT"ABSPEICHERN (J/N) ?"
10460 GETG$:IFG$=""THEN10460
10465 IFG$<>"J"THEN10485
10470 SYS57809 "MEMDISPLAY.CODE",8,1
10475 POKE251,60:POKE252,3:POKE781,215
10480 POKE782,3:POKE780,251:SYS65496
10485 PRINT"AKTIVIIERT WIRD DURCH:"
10490 PRINT"SYS828"
10495 END
10499 :
10500 A$=LEFT$(A$,2)
10510 A1=ASC(LEFT$(A$,1)):A2=ASC(RIGHT$(A$,1))
10520 A1=A1-48:IFA1>9THENA1=A1-7
10530 A2=A2-48:IFA2>9THENA2=A2-7
10540 A=16*A1+A2
10560 RETURN

```

Formel 1

Geschicklichkeitsspiel
für VC-20
Grundversion

Dieses Geschicklichkeitsspiel ist eine tolle Autorennen-Spielvariante auf dem VC-20, bei der Sie die Aufgabe haben, einen Rennwagen sicher über die Rennstrecke zu steuern.

Ihr Gegenspieler ist der VC-20, der ebenfalls einen Rennwagen steuert.

Liegeengebliebene Fahrzeuge, plötzlich auftretende Ölflecken auf der Fahrbahn und Schikanen die ab der dritten Runde auftreten sorgen dafür, daß die Fahrt über den Parcours zu einer heißen Runde wird. Insgesamt gibt es fünf Runden pro Spiel.

Steuern können Sie das Spiel sowohl über die Tastatur, als auch über Joystick!

```

1 REM VC-20 GRUNDVERSION
2 REM DIRK TRABHARDT
85
10 POKES6,29:POKE52,29
20 FORX=7544T07679:READA:POKEX,A:NEXT:GOTOB00
200 REM
210 PRINT"LM":PRINT"MPRESS PLAY ON TAPE"
220 WAIT37151,64,64
230 PRINT"MLDADING PART 2"
240 POKE198,7:POKE631,76:POKE632,207:POKE633,13:POKE634,82:POKE635,213:POKE636,1
3:END
400 REM
410 FORX=7430T07498:READA:POKEX,A:NEXT:GOTO2000
600 REM
610 FORX=46T008STEP-2
620 POKE36867,X:GOSUB700:NEXT:PRINT"L"
630 FORX=2T0468STEP2
640 POKE36867,X:GOSUB700:NEXT
650 GOTO200
700 FORJ=0T015:NEXTJ:RETURN
800 PRINT"L":FO=30720:FU=30962:Y=242

```

programme

```
810 FORX=7775T07782:READA:POKEX,A:POKEX+FO,6:POKEX+Y,A:POKEX+FU,6:NEXT
820 FORX=7799T07802:READA:POKEX,A:POKEX+FO,0:POKEX+Y,A:POKEX+FU,0:NEXT
830 FORX=7820T07825:READA:POKEX,A:POKEX+FO,2:POKEX+Y,A:POKEX+FU,2:NEXT
840 FORX=7839T07850:READA:POKEX,A:POKEX+FO,0:POKEX+Y,A:POKEX+FU,0:NEXT
850 FORX=7861T07872:READA:POKEX,A:POKEX+FO,0:POKEX+Y,A:POKEX+FU,0:NEXT
860 FORX=7883T07894:READA:POKEX,A:POKEX+FO,0:POKEX+Y,A:POKEX+FU,0:NEXT
870 FORX=7944T07965:POKEX,64:POKEX+44,64:NEXT:GOTO400
900 DATA102,102,255,247,247,255,102,102,102,102,255,239,239,255,102,102
1010 DATA56,254,254,56,40,254,254,56
1020 DATA56,254,254,40,56,254,254,56
1030 DATA0,0,0,0,0,0,0,0
1040 DATA126,129,129,129,129,129,129,126
1050 DATA0,31,32,66,64,33,18,10,0,192,60,130,1,33,161,162
1060 DATA10,9,8,16,16,15,0,0,162,58,4,4,4,8,240,0
1070 DATA96,144,136,132,130,129,129,126,6,9,17,33,65,129,129,126,126,129,129,65,
33,17,9,6
1080 DATA126,129,129,130,132,136,144,96,126,129,129,129,129,66,36,24
1090 DATA0,16,24,28,16,16,16,56,255,255,255,255,255,255,255,255
1100 DATA121,121,121,121,121,121,121,121,80,85,73,79,108,122,254,252,76,123
1110 DATA160,160,160,122,104,49,55,104,76,160,160,160,160,160,160,78,102,102,102
,102
1120 DATA77,160,160,160,160,160,160,99,119,119,119,119,99,160,160,160
1130 DATA173,19,145,72,169,0,141,19,145,173,17,145,41,28,74,74,133,99,173,17,145
,41,32
1140 DATA10,10,5,99,133,99,104,141,19,145,162,127,142,34,145,173,32,145,162,255,
142,34
1150 DATA145,41,128,74,74,74,74,5,99,73,143,133,99,169,0,133,98,162,144
1160 DATA56,32,73,220,96
1200 DATA"
**FORMEL 1** EIN RASANTES RENNEN GEGEN DEN COMPUT
ER ! "
1210 DATA"DU STEUERST DEN WEISSEN WAGEN MIT DEN TASTEN (Z) (X) (F5) (F7).
"
1220 DATA"DAS RENNEN GEHT UEBER FUENF RUNDEN. LIEGENBLEIBENDE KONKURRENTEN UND O
ELFLECKE
1230 DATA" BEHINDERN DICH! VIEL SPASS!"
2000 FA=158:READB$,C$,D$,E$:A$=B$+C$+D$+E$:PRINT"#####"
2010 W$=A$:W$=LEFT$(W$,22):A$=RIGHT$(A$,LEN(A$)-22)+" "
2020 Z$=LEFT$(W$,1):W$=RIGHT$(W$,21)+LEFT$(A$,1)
2030 A$=RIGHT$(A$,LEN(A$)-1)+Z$
2040 PRINTCHR$(145);CHR$(145);W$:FORI=0TO99:NEXT
2050 GETI$:IFI$<>" "THEN600
2060 X=X+1:IFX>50THEN2100
2070 GOTO2020
2100 X=0:FA=FA+16:IFFA>255THENFA=158
2110 POKE36879,FA:GOTO2020
```

Teil 2:

```
2 P=7712:Q=7734:RS=36877:S1=RS-1:S2=RS-3:LS=RS+1:U1=8057:U2=7934:U3=8090:U4=7727
3 FA=30720:F=48:E=1:POKE36879,26:POKE36869,240
4 PRINT"LEERJ=OYSTICK":INPUTA$:IFA$="J"THENXS=1
5 POKE1,6:POKE2,29:PRINT"LEER":GOSUB200:POKE36869,255:POKE36879,8:POKELS,3:GOSUB3
00
6 POKES1,128:J=2
10 IFXS<>1GOTO15
11 IFUSR(0)>9THEN21
12 X=USR(0):ONX+1GOTO21,27,28,21,25,21,21,26
15 GETS$
16 IFS$=""GOTO21
17 IFS$="Z"GOTO25
18 IFS$="X"GOTO26
19 IFS$="|"GOTO27
20 IFS$="|"GOTO28
21 POKES2,Z:ONJGOTO25,26,27,28
25 POKEP,51:P=P-1:GOSUB400:POKEP,47:J=1:Z=170:GOTO30
26 POKEP,51:P=P+1:GOSUB400:POKEP,48:J=2:Z=135:GOTO30
```

programme

```
27 POKEP,51:P=P-22:GOSUB400:POKEP,49:J=3:Z=160:GOTO30
28 POKEP,51:P=P+22:GOSUB400:POKEP,50:J=4:Z=145
```

```
30 POKEP+30720,1
31 GOSUB500:GOSUB600:GOTO10
200 A$(1)="444444444444444444444444"
202 A$(2)="433433433343333333333334"
203 A$(3)="433333333333333333333334"
204 A$(4)="433:44444444444444493334"
```

```
205 A$(5)="433433433343334333333334"
206 A$(6)="433433433343333;4444444"
207 A$(7)="4333334333333333333334"
208 A$(8)="4334333333433333343334"
209 A$(9)="4333334333;44444443334"
210 FORX=1TO22:READI:PRINTA$(I);:NEXT
```

```
211 POKE7690,62:POKE7778,62:POKE7690+FA,5:POKE7778+FA,5
212 FORI=8164TO8185:POKEI,52:POKEI+FA,7:NEXT:RETURN
300 FORX=7706TO7711:POKEX,51:POKEX+1,48:POKEX+1+FA,1:GOSUB350:NEXT
302 FORX=7728TO7733:POKEX,51:POKEX+1,48:POKEX+1+FA,2:GOSUB350:NEXT
303 FORY=1TO3:FORX=150TO180STEP5:GOSUB340:NEXTX:FORX=200TO150STEP-1:GOSUB340
304 NEXTX:NEXTY
305 FORD=0TO2000:NEXTD:POKES1,220:POKELS,3:FORD=1TO300:NEXTD:RETURN
340 POKES1,X:POKES2,X:POKELS,2:FORD=0TO10:NEXTD:RETURN
350 POKES2,158:FORD=0TO200:NEXTD:RETURN
```

```
400 V=PEEK(P):IFV>52ANDV<57THEN800
402 IFV<>51THEN 900
403 Q=Q+1:IFQ/20=INT(Q/20) THENGOSUB1000:RETURN
500 C=INT(RND(1)*4+1)
501 IFC=1THENU=U1:GOTO505
502 IFC=2THENU=U2:GOTO505
```

```
503 IFC=3THENU=U3:GOTO505
504 U=U4
505 Y=INT(RND(1)*15):IFY=9THEN510
506 IFY=3ORY=2ORY=1THEN520
507 RETURN
```

```
510 POKEU,53:POKEU+1,54:POKEU+22,55:POKEU+23,56
511 POKEU+FA,6:POKEU+1+FA,6:POKEU+22+FA,6:POKEU+23+FA,6:RETURN
520 POKEU,51:POKEU+1,51:POKEU+22,51:POKEU+23,51:RETURN
600 IFQ=7820ORQ=7875ORQ=8103ORQ=8116THENE=-1:F=47:GOTO651
602 IFQ=8100ORQ=8106THENE=-22:F=49:GOTO651
603 IFQ=7726ORQ=8022THENE=1:F=48:GOTO651
604 IFQ=7743ORQ=7817ORQ=7868ORQ=8028THENE=22:F=50
651 POKEQ,51:POKEQ+E,F:POKEQ+E+FA,4:Q=Q+E
652 IFQ=7734THENW1=W1+1
653 IFW1=5THEN1200
654 IFP=7734ORP=7712ORP=7756THENW2=W2+1
```

```
655 IFW2=5THEN1210
656 IFK=1THENRETURN
657 IFW2=3THENGOSUB700
658 RETURN
```

```
700 B$="0057025822632463446346636663686388609059":K=1
703 FORX=1TO39STEP4:A=(VAL(MID$(B$,X,2))):B=(VAL(MID$(B$, (X+2), 2)))
704 POKE7907+A,B:POKE7907+A+FA,7:NEXTX
706 FORX=8130TO8137:POKEX,52:POKEX+FA,7:NEXTX:RETURN
800 POKERS,230:J=INT(RND(1)*4+1):FORX=0TO1000:NEXTX:POKES1,255:POKERS,0:GOTO21
900 POKES1,0:POKES2,0:POKERS,128:FORX=15TO0STEP-.02:POKEP,X:POKELS,X:NEXTX:POKER
S,0
901 POKE198,0:RUN
1000 HP=INT(RND(1)*436+7703):HI=INT(RND(0)*4+47):HF=INT(RND(0)*7+1)
1001 IFPEEK(HP)=51THENPOKEHP,HI:POKEHP+FA,HF
1002 RETURN
1200 POKE36869,240:PRINT"LOREN DU HAST LEIDER VER- LOREN":FORX=0TO2000:NEXT:GOTO9
00
1210 POKE36869,240:PRINT"LORENGEWONNEN!!!! ":POKES1,0:POKES2,0:GOSUB2000:WAIT198,2
:RUN
2000 READD,T,A:IFA=-1THENLS=0:RETURN
2001 POKELS,9:POKES3,T+200:POKES1,T+199:FORX=0TOD*50:NEXTX:POKES3,0:POKES1,0
2002 FORX=0TOD*15:NEXTX:GOTO2000
2003 DATA 1,3,3,3,4,8,8,8,2,5,5,5,5,6,2,2,2,9,7,7,7,7
2004 DATA1,12,1,1,1,1,1,1,1,1,1,1,2,1,12,1,1,1,1,1,1,1,1,1,1,1,1,3,1,15,1,1,15,1,1,12,1,
1,12,1
2005 DATA3,7,4,1,15,1,1,15,1,1,12,1,1,12,1,3,7,3,1,1,1,1,1,1,1,1,7,1,1,7,1
2006 DATA1,1,1,1,1,1,4,-15,0,0,0,-1
```

6502-Assembler-Kurs

Teil 9

JMP Jump to New Location

(Springe zu einer Speicherstelle)

Dieser Befehl ist vergleichbar mit dem GOTO Befehl der Programmiersprache Basic. Anstelle der Zeilennummer wird hier jedoch die Adresse der anzuspringenden Speicherstelle, angegeben. Der JMP-Befehl ist der einzige Befehl, welcher die indirekte Adressierung beherrscht.

HEX	Adressierung	Bytes	Syntax	
4C	Absolut (direkt)	3	JMP addr	7 6 5 4 3 2 1 0
6C	Indirekt	3	JMP addr	NV BDIZC

```

10 SYS 36864
20 ;
30 ; **** BEISPIEL JMP-BEFEHL ****
40 ;
50 .OPT P,00
60 ;
70 *=49152
80 ;
100 VIDEO = 1024 ; ANFANGSADRESSE DES BILDSCHIRMSPEICHERS
110 TEMP = 832 ; SPEICHERPLATZ FUER ZEICHEN
120 START LDX #0
130 STX TEMP ; ZEICHEN ABLEGEN
140 LOOP LDA TEMP ; LADE ZEICHEN
150 STA VIDEO,X ; ZEICHEN IN BILDSCHIRMSPEICHER ABLEGEN
160 INX ; ZAEHLER UM 1 ERHOEHEN
170 BNE LOOP ; NAECHSTES ZEICHEN AUSGEBEN
180 INC TEMP ; ERHOEHE ZEICHENCODE
185 JMP LOOP
190 .END
200 PRINT"WICHTIGWICHTUNG !!!!!"
210 PRINT"DAS PROGRAMM LAEUFT IN EINER"
220 PRINT"ENDLOSSCHLEIFE.AUSSCHALTEN NUR"
230 PRINT"MIT RUN-STOP/RESTORE !"
240 SYS 49152

```

Das obige Beispiel veranschaulicht sehr deutlich die Funktion und Möglichkeiten des JMP-Befehles. Wie immer verwenden wir wieder den Bildschirmspeicher zur Demonstration unseres Programmes. Nach dem Start durch RUN, wird unser Programm in den Maschinencode übersetzt und durch den Befehl SYS 49152, in der Zeile 240, aufgerufen. Das Programm besteht aus der gleichen Programmschleife und dem gleichem Algorithmus wie er schon im letzten Teil verwendet wurde, jedoch wurde diesmal ein speziel-

ler Speicherplatz für den auszugebenden Zeichencode reserviert. Zu Beginn des Programmes wird, in Zeile 130, dieser Speicherstelle der Bildschirmcode 0 zugewiesen. Danach werden 255 Zeichen mit diesem Code auf den Schirm gebracht. Im Gegensatz zum letzten Programm wird jedoch nicht zum Basic zurückgesprungen, sondern die Zeilen 180 und 185 ausgeführt. In Zeile 180 wird durch den INC Befehl der aktuelle Zeichencode um 1 erhöht. Der JMP-Befehl in Zeile 185 sorgt nun dafür, daß das Pro-

gramm in Zeile 140 weitergeführt wird. Durch diese Technik befindet sich das Programm in einer sogenannten Endlosschleife, d. h., das Programm beginnt jedesmal von vorne und bricht niemals ab. Bei vielen Programmen würde nur das Ein- und Ausschalten des Computers diesen Vorgang beenden, dadurch wäre selbstverständlich das Programm gelöscht.

Der C64 besitzt jedoch die Möglichkeit durch gleichzeitiges Betätigen der RUN/STOP- und RESTORE-Taste einen NMI-In-

terrupt auszulösen und dadurch in den Basic-Interpreter zurückzuspringen. Das nachfolgende Listing zeigt den entsprechenden Maschinencode unserer kleinen Routine.

```

; **** BEISPIEL JMP-BEFEHL ****
;
50:  C000          .OPT P1,00
;
70:  C000          *= 49152
;
100: C000          VIDEO = 1024 ; ANFANGSADRESSE DES BILDSCHIRMSPEICHERS
110: C000          TEMP  = 832  ; SPEICHERPLATZ FUER ZEICHEN
120: C000 A2 00    START  LDX  #0
130: C002 BE 40 03 STX   TEMP ; ZEICHEN ABLEGEN
140: C005 AD 40 03 LOOP   LDA  TEMP ; LADE ZEICHEN
150: C008 9D 00 04 STA   VIDEO,X ; ZEICHEN IN BILDSCHIRMSPEICHER ABLEGEN
EN
160: C00B EB          INX          ; ZAEHLER UM 1 ERHOEHEN
170: C00C D0 F7      BNE  LOOP    ; NAECHSTES ZEICHEN AUSGEBEN
180: C00E EE 40 03  INC  TEMP    ; ERHOEHE ZEICHENCODE
185: C011 4C 05 C0    JMP  LOOP

```

Der JMP-Befehl kann auch die absolute (direkte) Adressierungsart verwenden. In diesem Fall wird das zweite Byte des Befehls in das niederwertige Byte des Befehlszählers und das dritte Byte des Befehlszählers in das höherwertige Byte des Befehlszählers geladen. Die Befehlsausführung wird von dieser Adresse an fortgesetzt.

JSR Jump to New Location Saving Return Address (Springe in Unterprogramm und registriere Rücksprungadresse)

Der JSR-Befehl arbeitet ähnlich wie der JMP-Befehl, jedoch mit der Ausnahme, dass vor Ausführung dieses Befehles, die Adresse des nächstfolgenden Befehles auf den Stack abgelegt wird. Kommt der Prozessor auf einen RTS-Befehl, wird das Unterprogramm beendet und der Prozessor springt an die Adresse, welche auf dem Stack abgelegt wurde.

HEX	Adressierung	Bytes	Syntax	7 6 5 4 3 2 1 0
20	Absolut (direkt)	3	JSR addr	NV B D I Z C

Beispiel:

```

100 VIDEO = 1024 ; ANFANGSADRESSE
    DES BILDSCHIRMSPEICHERS
110 TEMP = 832
120 START LDX #0
130 STX TEMP
140 LOOP LDA TEMP
150 STA VIDEO,X
155 JSR ZEIT ; VERZOEGERUNG
160 INX
170 BNE LOOP
180 INC TEMP
185 JMP LOOP
190 ;
200 ZEIT LDY #255 ; UNTERPROGRAMM ZUR
210 ZEITA DEY ; VERZOEGERUNG
220 BNE ZEITA
230 RTS
240 ;
250 .END
300 SYS 49152

```

In diesem Beispiel wurden die Zeilen 10 bis 80 weggelassen da diese identisch mit dem obigen Beispiel-Listing sind. Auch dieses Beispiel ist in Funktion und Algorithmus ähnlich wie das oben-abgedruckte Programm. Neu hinzugekommen ist ein kleines Unterprogramm in den Zeilen 200 bis 230. Dieses Unterprogramm zählt nach jedem Aufruf von 0 bis 255 und stellt somit eine Art Verzögerungsschleife dar. In Zeile 155 rufen wir diese Verzögerungsschleife durch unseren neuen Befehl „JSR“ auf. Wenn wir dieses Programm nun starten, können wir den Ablauf, durch die etwas langsamere Abarbeitung, wesentlich besser als bisher, verfolgen. Bei der Betrachtung sollte man jedoch bedenken, daß die Verzögerung hinter jeder einzelnen Zeichenausgabe aufgerufen wird. Das bedeutet, bevor das nächste Zeichen auf den Schirm gebracht wird, wurde die Verzögerungs-Routine 255 mal aufgerufen. An diesem Beispiel wird wieder einmal die Geschwindigkeit eines Maschinenprogrammes demonstriert. Probieren Sie dieses Beispiel doch einmal in Basic zu programmieren und vergleichen Sie die unterschiedlichen Ablaufgeschwindigkeiten.

Literatur:

6502 Microcomputer-Programmierung, Peter Heuer, Hofacker Verlag
6502 Programmierung in Assembler, Lance A. Leventhal

64-Intern, Angershausen, Becker, Englisch, Gerits, Data-Becker Buch
6502/65c02, Christian Persson, Heinz Heise Verlag

CHECKSUMMER 1.0 CPC

Der Checksummer 1.0 CPC ist das Eingabehilfsprogramm für alle Programme des TRONIC-Verlages. Es wird in Zukunft in jeder Ausgabe abgedruckt, um auch neuen Lesern die Eingabe der Programme zu erleichtern.

Wie funktioniert der Checksummer?

Tippen Sie zunächst das Programm ab, wobei sie darauf achten müssen, daß alle Zwischenräume mit dem Listing übereinstimmen. Auch REM-Zeilen müssen original übernommen werden.

Sichern Sie nun das Programm auf KASSETTE oder DISKETTE bevor Sie es starten!

Nach dem Speichern wird das Programm mit „RUN 65200“ aktiviert. Es erscheint die Meldung: CHECKSUMMER AKTIV FUNKTIONSTASTE 0. Betätigen Sie nun die Taste 0, wird der Checksummer aufgerufen. Befindet sich kein zweites zu prüfendes Programm im Speicher, wird ein Selbsttest durchgeführt. ertönt am Ende dieses Selbsttestes ein Signal, ist das Programm noch nicht

fehlerfrei. Sie sollten in diesem Fall die auf dem Bildschirm ausgegebenen Checksummen mit denen in Ihrer Zeitschrift vergleichen und korrigieren. Arbeitet der Checksummer überhaupt nicht, ist der Fehler so gravierend, daß Sie das Programm neu eingeben müssen.

Ist Ihr Programm fehlerfrei, entfernen Sie bitte die Zeilen 65500 bis 65530 (Delete 65500 →) um den Selbsttest auszuschalten. Speichern Sie jetzt das Programm mit dem Befehl SAVE "CHECKV1", A auf Kasette oder Diskette. Nur so können Sie es später mit MERGE an Ihre Programme anhängen.

Wie benutzt man den Checksummer?

Wollen Sie ein Programm mit CHECK überprüfen, laden Sie zunächst das zu prüfende Programm. Dann geben Sie den Befehl MERGE "CHECKV1" ein, um das Prüfprogramm anzuhängen. Wichtig ist, daß das zu prüfende Programm keine Zeilennummer besitzt, die größer als 65199 ist. Jetzt aktivieren Sie den Checksummer mit

RUN 65200 und Sie können jederzeit die Prüfung mit der Funktionstaste 0 überprüfen. Vergleichen Sie nun die im Heft abgedruckten Prüfsummen mit denen, die Ihr Programm ausgibt.

Falls Sie einen Fehler feststellen, müssen Sie nicht das Programm neu laden, sondern können ihn normal berichtigen und dann wieder mit 0 das Check-Programm starten. Möchten Sie den Checksummer löschen, geben Sie einfach den Befehl DELETE 65200- ein.

Anmerkung zu Checksummer 1.0 CPC:

Groß-Kleinschreibung beachten!
Leerzeichen innerhalb einer Zeile müssen dem ORIGINAL-Listing entsprechen!
Leerzeichen am Ende einer Zeile werden ignoriert!
Umgewandelte TOOKEN werden automatisch in den Ursprungswert zurückgesetzt!
REM-Zeilen müssen eingegeben werden!

Wer den Checksummer nicht eingeben möchte kann diesen auch unter der folgenden Bestellnummer beziehen:

Bestellnummer:

CVS 10K Kassette
CVS 10D Diskette

10 DM
20 DM

```
65200 '*** Tronic CPC Checksummer ***
65210 '*** Autor Frank Brall 1985 ***
65220 '          VERSION 1.0
65230 KEY 128,"65530 REM TRONIC"+CHR$(13
)+ "run 65250"+CHR$(13)
65240 PRINT:PRINT"CHECKSUMMER AKTIV FUN
KTIONSTASTE 0":PRINT:END
65250 CLS:PRINT"Tronic-Verlag Checksumme
r Version 1.0"
65260 PRINT:PRINT:PRINT:PRINT"<B>ILDSCHI
RM <D>RUCKER"
65270 E$=INKEY$:E$=LOWER$(E$):IF E$="" T
HEN 65270
65280 IF E$="b" THEN a=0:GOTO 65310
65290 IF E$="d" THEN a=8:GOTO 65310
65300 GOTO 65200
65310 ADRESS=368:GCHECK=0:CLS:PRINT#A,"C
HECK V1:PRUEFSUMMEN:":PRINT#A:PRINT#A,"Z
EILENNR.: SUMMEN"
65320 LAENGE=(PEEK(ADRESS+1)*256)+PEEK(A
DRESS)
65330 ZNUM=(PEEK(ADRESS+3)*256)+PEEK(ADR
ESS+2)
65340 IF ZNUM=65200 THEN IF F=0 THEN 654
80 ELSE PRINT #A, USING "*****";ZNUM1:P
RINT#A,"-";PRINT#A,":":PRINT #A,
USING "*****";ZCHECK:F=0:GOTO 65480
65350 IF ZNUM=65500 THEN 65510
65360 ZCHECK=0:FOR I=0 TO LAENGE-6
```

```
65370 WERT=PEEK(I+ADRESS+4)
65380 IF WERT =32 THEN 65410
65390 ZCHECK=ZCHECK+WERT+(I*WERT)
65400 IF ZCHECK>65535 THEN ZCHECK=ZCHECK
-65536:GOTO 65400
65410 NEXT I
65420 IF F=0 THEN ZCHECK1=ZCHECK:ZNUM1=Z
NUM:ADRESS=ADRESS+LAENGE:F=1:GOTO 65320
65430 F=0:ZNUM2=ZNUM
65440 PRINT #A,USING "*****";ZNUM1:P
RINT#A,"-";PRINT #A,USING "*****";ZNUM2:P
RINT#A,":":PRINT #A,USING "*****";ZCHE
CK+ZCHECK1
65450 GCHECK=GCHECK+ZCHECK+ZCHECK1
65460 IF GCHECK>65535 THEN GCHECK=GCHECK
-65536:GOTO 65460
65470 ADRESS=ADRESS+LAENGE:GOTO 65320
65480 PRINT#A,"-----
-"
65490 PRINT #a,"GESAMTCHECKSUMME: "GCHEC
K
65500 GCHECK=0:A=0:PRINT#A:PRINT#A,"CHEC
KSUMMER SELBSTTEST !":GOTO 65360
65510 IF GCHECK=62078 THEN PRINT#A,"--EN
DE--":END
65520 PRINT "FEHLER IM CHECKSUMMER !":SO
UND 1,100,500
65530 REM TRONIC
```

CHECKSUMMER SELBSTTEST !

65200-65210: 73781
65220-65230: 36739
65240-65250: 11826
65260-65270: 86096

65280-65290: 53556
65300-65310: 24701
65320-65330: 43809
65340-65350: 74115

65360-65370: 89147
65380-65390: 32907
65400-65410: 49578
65420-65430: 28838

65440-65450: 18832
65460-65470: 84126
65480-65490: 74923
--ENDE--

programme

Wir schreiben das Jahr 2001 und befinden uns auf der Erde. Der Wandel der Zeit ist nicht ganz spurlos an uns vorübergegangen und wir haben Kontakt zu außerirdischen Lebewesen. Einer dieser Außerirdischen muß mit seinem Raumschiff auf einem Baugerüst in einer Stadt notlanden, da sein Energievorrat verbraucht ist.

Alien 2001 Schneider CPC 464

Alien versucht, die auf dem Baugerüst verteilten Batterien einzusammeln und zu seinem Raumschiff zu bringen. Auf die-

sem Weg muß er jedoch über Löcher und Fließbänder springen, Leitern herauf- oder hinabklettern, Rutschen und Aufzüge benutzen und dabei stets auf etwaige Hindernisse achten. Aber beachten Sie bitte, daß Alien nur über begrenzte Kraft verfügt. Springt er gegen einen Hammer oder fällt er zu tief, verliert er Kraft. Sollte ihm die gesammte Kraft ausgegangen sein, bedeutet das seinen Tod. Die Berührung einer Bombe ist das sofortige Ende. Achten Sie auch auf Wasser und verschwindende Bretter, da beides eine große Gefahr darstellt.

Ingesamt gibt es bei diesem Spiel vier verschiedene Bilder, in denen jeweils ein Bonuszähler läuft. Je schneller Sie ein Spiel beenden, um so mehr Punkte erhalten Sie. Verlassen können Sie die Bilder durch die Tür, wenn Sie zuvor die einzelnen Batterien eingesammelt haben.

Hinweise: Vor dem Abtippen des Programms sollte „AUTO“ eingegeben werden. Zum Spielen wird ein Joystick benötigt. Gestartet wird das Spiel mit „CTRL“ und „RUN“.



Teil 1:

```
10 MODE 1: BORDER 11
20 FOR i=&C000 TO &FFFF STEP 5
30 IF INT(RND*25)=3 THEN POKE i,50
40 NEXT
50 MOVE 0,0: DRAW 0,399,1: DRAW 639,399: DR
AW 639,0: DRAW 0,0
60 INK 2,3: INK 3,9
70 y=2: FOR x=4 TO 24 STEP 2
80 LOCATE x,y: PEN 2: PRINT "ALIEN " : PEN
3: PRINT "2001": y=y+2: NEXT
90 MOVE 414,0: DRAW 16,399,1: MOVE 590,0: D
RAW 193,399
100 LOCATE 2,20: PEN 3: PRINT CHR$(164); "
by RAINER KROTZ"
110 LOCATE 22,6: PEN 2: PRINT CHR$(164); "
by RAINER KROTZ"
120 LOCATE 27,24: PEN 1: PRINT "LOADING"
130 RUN "!Alien 2001"
```

Teil 2:

```
10 REM *** RAINER KROTZ ***
20 REM
30 sc=0
40 DIM a$(20)
50 MODE 2: PAPER 0: PEN 1: INK 1,24: CLS
60 SYMBOL AFTER 255
```

```
70 ENT 1,30,-20,3
80 ENT 2,200,5,6
90 ENV 1,5,2,20,5,-2,20
100 MODE 0
110 INK 0,0: INK 1,3: INK 2,24: INK 3,4: INK
4,13: INK 5,15: INK 6,18,21: INK 7,9: INK 8
,16: INK 9,12: INK 10,26,12: INK 11,1: INK 1
2,2: INK 13,26,20: INK 14,20,26
120 b=1: kra=13: e=7
130 WINDOW #1,1,20,1,20: PAPER #1,0
140 WINDOW #2,1,20,21,25: PAPER #2,3
150 CLS #1: CLS #2
160 LOCATE 1,22: PEN 0: PAPER 3: PRINT "SCD
RE " : sc
170 LOCATE 1,23: PRINT "POWER " : PEN 1: P
RINT CHR$(143); PEN 4: FOR i=1 TO 5: PRINT
CHR$(143); : NEXT
180 LOCATE 1,24: PEN 0: PRINT "ENERGY:"
190 MOVE 0,0: DRAW 0,79,2: DRAW 639,79: DRA
W 639,0: DRAW 0,0
200 GOSUB 2950
210 pa=0: ON b GOSUB 1960,2210,2460,2710

220 bon=5000
230 BORDER 11: PAPER 3: PEN 0: LOCATE 8,22:
PRINT sc: PAPER 0
240 LOCATE 1,1
250 FOR i=1 TO 20: FOR t=1 TO 20
260 z$=MID$(a$(i),t,1)
270 IF z$=CHR$(64) OR z$=CHR$(66) OR z$=
```

```

CHR$(88) THEN p=5
280 IF z$=CHR$(65) OR z$=CHR$(89) THEN p
=2
290 IF z$=CHR$(67) THEN p=5
300 IF z$=CHR$(68) THEN p=6
310 IF z$=CHR$(69) THEN p=5
320 IF z$=CHR$(70) OR z$=CHR$(71) OR z$=
CHR$(72) THEN p=1
330 IF z$=CHR$(73) THEN p=7
340 IF z$=CHR$(74) OR z$=CHR$(75) THEN p
=2
350 IF z$=CHR$(76) THEN p=1
360 IF z$=CHR$(77) THEN p=7
370 IF z$=CHR$(89) THEN p=2
380 IF z$=CHR$(96) THEN p=13
390 IF z$<>CHR$(64) AND z$<>CHR$(66) AND
z$<>CHR$(88) THEN pa=0 ELSE pa=12
400 IF z$=CHR$(85) THEN p=11
410 IF z$=CHR$(85) THEN pa=4
420 IF z$=CHR$(96) THEN pa=14
430 IF z$=CHR$(97) THEN p=2
440 PAPER pa: PEN p: PRINT z$:
450 NEXT: NEXT
460 PAPER 0: PEN 10: LOCATE ex1,ey1: PRINT
"N": LOCATE ex2,ey2: PRINT "N": LOCATE ex3,
ey3: PRINT "N"
470 PEN 7: LOCATE x,y: PRINT m$
480 e1=1: e2=1: e3=1: tu=0: weg=0
490 sf=0
500 IF x=14 AND e=19 AND y=5 AND b=4 THE
N GOTO 1760
510 IF x=15 AND e=19 AND y=5 AND b=4 THE
N GOTO 1760
520 PAPER 3
530 IF x=ex1 AND y=ey1 AND e1=1 THEN e=e
+1: zf=1: e1=0: tu=tu+1
540 IF x=ex2 AND y=ey2 AND e2=1 THEN e=e
+1: zf=1: e2=0: tu=tu+1
550 IF x=ex3 AND y=ey3 AND e3=1 THEN e=e
+1: zf=1: e3=0: tu=tu+1
560 IF zf=1 THEN sc=sc+100*b: SOUND 1,478
,20,5: SOUND 1,319,30,6: SOUND 1,213,40,7:
LOCATE 8,22: PAPER 3: PEN 0: PRINT sc: LOCAT
E e,24: PEN 1: PRINT CHR$(246): zf=0: LOCATE
x,y: IF tu=3 THEN weg=1
570 PAPER 0
580 IF b<4 AND weg=1 AND x=1 AND y=0 THE
N 1260
590 IF bon>1 THEN bon=bon-1
600 w$=MID$(a$(y),x,1)
610 IF w$=CHR$(96) THEN LOCATE x,y: PEN 1
3: PAPER 14: PRINT CHR$(96): y=y-1: PRINT CH
R$(22): LOCATE x,y: PEN 7: PRINT m$
620 w$=MID$(a$(y),x,1)
630 l$=w$
640 IF w$=CHR$(68) THEN 1320
650 IF w$=CHR$(96) THEN a=13: ab=14
660 IF w$=CHR$(65) THEN a=2: ab=0
670 IF w$=CHR$(97) THEN a=2: ab=0
680 q$=MID$(a$(y+1),x,1)
690 IF q$=CHR$(66) THEN LOCATE x,y: PRINT
" ": x=x+1: PRINT CHR$(22): LOCATE x,y: PEN
7: PRINT m$: GOTO 490
700 IF q$=CHR$(32) THEN GOSUB 1110

```

```

710 IF q$=CHR$(68) THEN GOTO 1320
720 IF q$=CHR$(88) THEN LOCATE x,y: PRINT
" ": x=x-1: PRINT CHR$(22): LOCATE x,y: PEN
7: PRINT m$: GOTO 490
730 IF q$=CHR$(89) THEN LOCATE x,y: PRINT
" ": x=x-1: y=y+1: LOCATE x,y: PEN 7: PRINT
m$: GOTO 680
740 IF q$=CHR$(85) THEN 1590
750 IF q$=CHR$(69) THEN MID$(a$(y+1),x,1
)=CHR$(32): LOCATE x,y+1: PRINT " "
760 PRINT CHR$(22)CHR$(0): PAPER 0
770 IF JOY(0)=1 AND w$=CHR$(65) THEN LOC
ATE x,y: PEN 2: PRINT w$: y=y-1: sf=1
780 IF JOY(0)=2 AND q$=CHR$(65) THEN LOC
ATE x,y: PEN 2: PRINT w$: y=y+1: sf=1
790 IF JOY(0)=4 THEN LOCATE x,y: PEN a: PA
PER ab: PRINT l$: x=x-1: m$=CHR$(84): sf=1
800 IF JOY(0)=8 THEN LOCATE x,y: PEN a: PA
PER ab: PRINT l$: x=x+1: m$=CHR$(83): sf=1
810 ab=0
820 IF JOY(0)=16 THEN GOSUB 920
830 IF x<1 THEN x=1
840 IF x>20 THEN x=20
850 IF x<1 THEN x=1
860 IF x>20 THEN x=20
870 PRINT CHR$(22)
880 PEN 7: LOCATE x,y: PRINT m$
890 PRINT CHR$(22)CHR$(0)
900 IF sf=1 THEN SOUND 1,0,2,4,0,0,10
910 GOTO 490
920 IF x<3 AND m$=CHR$(84) THEN RETURN
930 IF x>18 AND m$=CHR$(83) THEN RETURN

940 SOUND 1,402,22,5,0,1
950 IF m$=CHR$(83) THEN s=1 ELSE s=-1
960 LOCATE x,y: PEN a: PRINT l$: x=x+s: y=y-
1
970 t$=MID$(a$(y),x,1)
980 IF t$=CHR$(65) THEN PRINT CHR$(22)
990 LOCATE x,y: PEN 7: PRINT m$
1000 PRINT CHR$(22)CHR$(0)
1010 FOR k=1 TO 20: NEXT
1020 IF t$=CHR$(67) THEN GOSUB 1050: GOTO
1040
1030 LOCATE x,y: PEN a: PRINT t$: x=x+s: y=y
+1
1040 RETURN
1050 SOUND 2,536,200,4,1
1060 LOCATE x,y: PEN 5: PRINT CHR$(67): y=y
+1
1070 PAPER 3: LOCATE kra,23: PRINT " ": PAP
ER 0: kra=kra-1
1080 IF kra=7 THEN GOTO 1690
1090 l=1-1
1100 RETURN
1110 fa=0
1120 w$=MID$(a$(y),x,1)
1130 IF w$=CHR$(64) THEN p=5
1140 IF w$=CHR$(73) THEN p=7
1150 IF w$=CHR$(70) OR w$=CHR$(71) OR w$
=CHR$(72) THEN p=1
1160 IF w$<>CHR$(64) THEN pa=0 ELSE pa=1
2
1170 LOCATE x,y: PEN p: PAPER pa: PRINT w$

```

programme

```
1180 y=y+1:fa=fa+1
1190 q$=MID$(a$(y+1),x,1)
1200 PEN 7:LOCATE x,y:PRINT m$
1210 IF q$=CHR$(32) THEN GOTO 1120
1220 IF fa>1 THEN LOCATE kra,23:PAPER 3:
PRINT " ":kra=kra-1:FOR h=10 TO 20:SOUND
  1,1136,3,5,0,0,h:NEXT
1230 IF kra=7 THEN 1490
1240 PAPER 0
1250 RETURN
1260 FOR i=399 TO 80 STEP -1:MOVE 0,i:DR
AW 639,i,0:NEXT
1270 SYMBOL AFTER 255
1280 LOCATE 6,10:PEN 10:PRINT "BONUS:";b
on
1290 b=b+1
1300 sc=sc+bon
1310 GOTO 200
1320 FOR t=7 TO 1 STEP -0.3
1330 SOUND 1,0,15,t,0,0,15
1340 PEN 10:LOCATE x,y:PRINT CHR$(87)
1350 NEXT
1360 LOCATE x,y:PRINT CHR$(86):FOR w=1 T
O 1000:NEXT:LOCATE x,y:PRINT " "
1370 BORDER 3:PAPER 0:FOR f=y TO 1 STEP
-1
1380 q$=MID$(a$(f),x,1)
1390 IF q$<>CHR$(32) THEN f=f-1:IF f=0 T
HEN GOTO 1430 ELSE GOTO 1380
1400 LOCATE x,f:PEN 1:PRINT CHR$(228):FO
R wt=1 TO 200:NEXT
1410 LOCATE x,f:PRINT " "
1420 NEXT
1430 LOCATE x,f+1:PRINT " "
1440 FOR a=0 TO 320
1450 MOVE a,0:DRAW a,399,1:MOVE 639-a,0:
DRAW 639-a,399:NEXT
1460 SYMBOL AFTER 255
1470 IF sc>hsc THEN hsc=sc
1480 FOR s=5 TO 16:PEN 2:PAPER 1
1490 READ z$
1500 LOCATE s,10:PRINT z$
1510 SOUND 1,0,2,7,0,0,5
1520 FOR wt=1 TO 250:NEXT
1530 NEXT
1540 DATA *,G,A,M,E,-,-,0,V,E,R,*
1550 PEN 10:LOCATE 5,15:PRINT "SCORE:";s
c
1560 PEN 7:LOCATE 5,17:PRINT "HISCORE:";
hsc
1570 PEN 11:LOCATE 1,24:PRINT "Press <FI
RE> to play"
1580 IF JOY(0)=16 THEN RUN ELSE GOTO 158
0
1590 PEN 7:LOCATE x,y:PRINT CHR$(84)
1600 s=478
1610 FOR i=90 TO 95
1620 LOCATE x,y:PRINT CHR$(i)
1630 FOR r=1 TO 5
1640 s=s+10
1650 SOUND 1,s,20,5
1660 NEXT:NEXT
1670 LOCATE x,y:PRINT " "
1680 GOTO 1370
1690 SOUND 4,60,200,7,0,2
```

```
1700 FOR d=1 TO 3
1710 t=97+d:PAPER 0
1720 LOCATE x,y:PEN 7:PRINT CHR$(t)
1730 FOR w=1 TO 500:NEXT
1740 NEXT
1750 y=y-1:GOTO 1370
1760 PEN 2:PAPER 0
1770 LOCATE 14,5:PRINT " "
1780 x=14:y=5
1790 y=y-1:IF y=0 THEN GOTO 1840
1800 LOCATE x,y:PRINT "JK":FOR wt=1 TO 3
OO:NEXT
1810 LOCATE x,y:PRINT " "
1820 x=x-1
1830 GOTO 1790
1840 LOCATE x,1:PRINT " "
1850 FOR v=399 TO 0 STEP -1:MOVE 0,v:DR
AW 639,v,0:NEXT:MODE 2:PAPER 0:PEN 1
1860 FOR s=49152 TO 65535 STEP 10
1870 IF INT(RND*20)=3 THEN POKE s,7
1880 NEXT
1890 SYMBOL AFTER 255
1900 MOVE 300,0:DRAW 120,399,1:MOVE 639,
120:DRAW 120,399
1910 LOCATE 26,10:PRINT "*CONGRATULATION
S*"
1920 LOCATE 30,12:PRINT "You have saved
the ALIEN"
1930 LOCATE 36,16:PRINT "Your Score is "
;sc
1940 LOCATE 44,22:PRINT "Prepare to cont
inue !"
1950 FOR wt=1 TO 6000:NEXT:GOTO 50
1960 a$(1)=" "
1970 a$(2)=" "
1980 a$(3)=" Aedaaa "
1990 a$(4)=" A A @aaaa "
2000 a$(5)="a A A A "
2010 a$(6)="aaa A A "
2020 a$(7)=" eeee eeeeeea "
2030 a$(8)=" A "
2040 a$(9)=" A "
2050 a$(10)=" Aee @ @ eeeAeeeeeee"
2060 a$(11)=" A A "
2070 a$(12)=" A A "
2080 a$(13)=" @aaa A "
2090 a$(14)=" A eeeeeeeeeee "
2100 a$(15)=" A C C "
2110 a$(16)=" A "
2120 a$(17)=" @eeBAeeeeeeeeea "
2130 a$(18)=" A A "
2140 a$(19)=" A A "
2150 a$(20)="eeeeeeeeeeUUUUeeeeee"
2160 ex1=13:ey1=3:ex2=18:ey2=13:ex3=16:e
y3=19
2170 x=1:y=19
2180 m$=CHR$(83)
2190 a=5
2200 RETURN
2210 a$(1)=" "
2220 a$(2)=" "
2230 a$(3)="AIDIIDIA "
2240 a$(4)="A A "
2250 a$(5)="A A "
```



```

2260 a$(6)="A   IIII           VII"
2270 a$(7)="A                   Y  "
2280 a$(8)="A                   Y  "
2290 a$(9)="A                   D  "
2300 a$(10)="AIII III III III III"
2310 a$(11)="A                   C  "
2320 a$(12)="A                   "
2330 a$(13)="IIIXIXIIXIXIXIXIIIA"
2340 a$(14)="                   CCC  A"
2350 a$(15)="                   A"
2360 a$(16)="                   YIDIDIA VII"
2370 a$(17)="                   Y  A  Y  "
2380 a$(18)="                   Y  A  Y  "
2390 a$(19)="a                   A  "
2400 a$(20)="IIIIIIIIUUUUUIIIIIIII"
2410 ex1=5:ey1=5:ex2=20:ey2=9:ex3=7:ey3=
19
2420 x=20:y=5
2430 m$=CHR$(84)
2440 o=19
2450 RETURN
2460 a$(1)="                   "
2470 a$(2)="a                   "
2480 a$(3)="G`H           YGH FH FGG`"
2490 a$(4)="`           Y           C`"
2500 a$(5)="`           Y           "
2510 a$(6)="`           Y           YGGG`"
2520 a$(7)="`           Y           "
2530 a$(8)="FGH FGGGGGH           "
2540 a$(9)="                   C           FGGGGH`"
2550 a$(10)="                   D           M`"
2560 a$(11)="FGGGAGGGGAH           M`"
2570 a$(12)="                   C A           A           C`"
2580 a$(13)="                   A           A           "
2590 a$(14)="`GGGGH FGGGGDGGGGGG`"
2600 a$(15)="`           "
2610 a$(16)="`           "
2620 a$(17)="`           FAGDGAH           YGGAH`"
2630 a$(18)="`           A           A           Y           A`"
2640 a$(19)="`           A           A           A`"
2650 a$(20)="`IIXBIIUUUIIIIIUUUII`"
2660 ex1=10:ey1=7:ex2=16:ey2=8:ex3=2:ey3=
10
2670 x=20:y=19
2680 m$=CHR$(84)
2690 o=2
2700 RETURN
2710 a$(1)="                   "
2720 a$(2)="                   "
2730 a$(3)=" A@A                   A@"
2740 a$(4)=" A A                   A`"
2750 a$(5)=" A A                   JK A`"
2760 a$(6)="` IIIIEEIEIEEIII A`"
2770 a$(7)="`                   A`"
2780 a$(8)="`                   D A`"
2790 a$(9)="`           FGAGGGGH           A`"
2800 a$(10)="`           A           YGA AI`"
2810 a$(11)="`           A           Y A A`"
2820 a$(12)="`           YGAGH           Y A A`"
2830 a$(13)="`           YC A           Y A A`"
2840 a$(14)="`           YM A           A A`"
2850 a$(15)="`           M FAGGGBGH           A AE`"
2860 a$(16)="`           M A           A A`"
2870 a$(17)="`           M A D           A A`"

```

```

2880 a$(18)="` C YEEEE A A A`"
2890 a$(19)="` D@@EDA A A`"
2900 a$(20)="`@@@X@@@@@@@@@UU@@@@@`"
2910 ex1=3:ey1=2:ex2=10:ey2=8:ex3=12:ey3=
=19
2920 x=20:y=2
2930 m$=CHR$(84)
2940 RETURN
2950 SYMBOL AFTER 63
2960 SYMBOL 64,255,0,0,0,0,0,255
2970 SYMBOL 65,255,129,255,129,255,129,2
55,129
2980 SYMBOL 66,255,8,4,126,4,8,0,255
2990 SYMBOL 67,24,24,24,24,62,60
3000 SYMBOL 68,4,8,24,44,94,126,60,24
3010 SYMBOL 69,255,255
3020 SYMBOL 70,7,9,16,38,38,16,9,7
3030 SYMBOL 71,255,153,0,102,102,0,153,2
55
3040 SYMBOL 72,224,144,8,100,100,8,144,2
24
3050 SYMBOL 73,255,24,36,36,66,66,129,25
5
3060 SYMBOL 74,1,3,7,15,255,63,11,16
3070 SYMBOL 75,128,192,224,240,255,252,2
08,8
3080 SYMBOL 77,16,16,16,16,16,16,16,16
3090 SYMBOL 78,24,24,36,44,36,44,36,60
3100 SYMBOL 83,0,34,20,12,26,127,255,173
3110 SYMBOL 84,0,68,40,48,88,254,255,181
3120 SYMBOL 85,0,255,255,255,255,255,255
,255
3130 SYMBOL 86,8,64,17,0,40,4,65,20
3140 SYMBOL 87,8,72,73,42,42,148,6,38
3150 SYMBOL 88,255,16,32,126,32,16,0,255
3160 SYMBOL 89,1,3,7,13,25,49,97,255
3170 SYMBOL 90,0,0,68,40,48,88,254,255
3180 SYMBOL 91,0,0,68,40,48,88,254
3190 SYMBOL 92,0,0,0,0,68,40,48,88
3200 SYMBOL 93,0,0,0,0,0,68,40,48
3210 SYMBOL 94,0,0,0,0,0,0,68,40
3220 SYMBOL 95,0,0,0,0,0,0,0,68
3230 SYMBOL 96,255,0,255,0,255,0,255,0
3240 SYMBOL 97,60,66,129,129,177,161,129
,129
3250 SYMBOL 98,0,0,0,0,24,60,126,255
3260 SYMBOL 99,0,0,0,0,0,24,60,126
3270 SYMBOL 100,0,0,0,0,0,0,24,60
3280 RETURN

```

Teil 1:

CHECK V1:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN
10-	20: 22196
30-	40: 24336
50-	60: 36482
70-	80: 58147
90-	100: 81173
110-	120: 64794
130-	: 6888

GESAMTCHECKSUMME: 24984

Teil 2:

CHECK V1:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN
10-	20: 14410
30-	40: 5716
50-	60: 12038
70-	80: 6818
90-	100: 8281
110-	120: 66123
130-	140: 17305
150-	160: 33101
170-	180: 20862

werkstatt

190- 200: 30826	810- 820: 15073	1430- 1440: 18714	2050- 2060: 34981	2670- 2680: 14878
210- 220: 39126	830- 840: 29149	1450- 1460: 24327	2070- 2080: 17668	2690- 2700: 2461
230- 240: 27670	850- 860: 29149	1470- 1480: 53520	2090- 2100: 31947	2710- 2720: 9442
250- 260: 45883	870- 880: 18379	1490- 1500: 10118	2110- 2120: 33556	2730- 2740: 18850
270- 280: 98163	890- 900: 23366	1510- 1520: 17535	2130- 2140: 16751	2750- 2760: 38446
290- 300: 42844	910- 920: 33798	1530- 1540: 18102	2150- 2160: 93472	2770- 2780: 16230
310- 320: 62702	930- 940: 39018	1550- 1560: 56281	2170- 2180: 14122	2790- 2800: 33803
330- 340: 77517	950- 960: 96908	1570- 1580: 64533	2190- 2200: 2479	2810- 2820: 30574
350- 360: 42994	970- 980: 42126	1590- 1600: 24417	2210- 2220: 9442	2830- 2840: 27816
370- 380: 43787	990- 1000: 21405	1610- 1620: 26453	2230- 2240: 18754	2850- 2860: 35162
390- 400: 49294	1010- 1020: 28738	1630- 1640: 13310	2250- 2260: 22985	2870- 2880: 31010
410- 420: 48671	1030- 1040: 48583	1650- 1660: 4781	2270- 2280: 15353	2890- 2900: 49707
430- 440: 39038	1050- 1060: 57136	1670- 1680: 9065	2290- 2300: 35682	2910- 2920: 62984
450- 460: 45597	1070- 1080: 15403	1690- 1700: 11386	2310- 2320: 13250	2930- 2940: 5734
470- 480: 18172	1090- 1100: 7170	1710- 1720: 34724	2330- 2340: 47916	2950- 2960: 15410
490- 500: 24008	1110- 1120: 23297	1730- 1740: 12337	2350- 2360: 29326	2970- 2980: 52475
510- 520: 21099	1130- 1140: 42854	1750- 1760: 12472	2370- 2380: 20963	2990- 3000: 24826
530- 540: 37336	1150- 1160: 88917	1770- 1780: 11885	2390- 2400: 44217	3010- 3020: 15686
550- 560: 64129	1170- 1180: 69061	1790- 1800: 76372	2410- 2420: 61969	3030- 3040: 49508
570- 580: 28790	1190- 1200: 42084	1810- 1820: 15623	2430- 2440: 7980	3050- 3060: 39512
590- 600: 60571	1210- 1220: 53510	1830- 1840: 6257	2450- 2460: 4919	3070- 3080: 46000
610- 620: 28497	1230- 1240: 11675	1850- 1860: 50992	2470- 2480: 29524	3090- 3100: 38635
630- 640: 16649	1250- 1260: 62029	1870- 1880: 23160	2490- 2500: 22015	3110- 3120: 69521
650- 660: 77508	1270- 1280: 31073	1890- 1900: 38553	2510- 2520: 28718	3130- 3140: 27277
670- 680: 63519	1290- 1300: 24445	1910- 1920: 77392	2530- 2540: 37475	3150- 3160: 38252
690- 700: 50370	1310- 1320: 26817	1930- 1940: 81141	2550- 2560: 33547	3170- 3180: 43321
710- 720: 50612	1330- 1340: 30107	1950- 1960: 22027	2570- 2580: 23350	3190- 3200: 22547
730- 740: 73915	1350- 1360: 36075	1970- 1980: 15107	2590- 2600: 38336	3210- 3220: 16987
750- 760: 21506	1370- 1380: 52507	1990- 2000: 23228	2610- 2620: 32177	3230- 3240: 50647
770- 780: 95120	1390- 1400: 40701	2010- 2020: 30308	2630- 2640: 22949	3250- 3260: 29387
790- 800: 63256	1410- 1420: 8397	2030- 2040: 13166	2650- 2660: 92373	3270- 3280: 8826

GESAMTCHECKSUMME: 53609

Durch „Directory“ wird das Handbuchblättern überflüssig!!

Das Programm „Directory“, ist wie der Name schon sagt, zur Bearbeitung des Directory's gedacht. Hierdurch kann man einzelne Programme in „Files“ umbenennen,

„RENAMEN“ oder ganz und gar aus dem Directory löschen. Außerdem besteht nach erfolgter Eingabe die Möglichkeit, das Directory zu überprüfen oder wieder

zu bearbeiten.

Welche Vorgehensweise in den einzelnen Programmteilen nötig ist, wird im Programm selbst genau erläutert.

```

10 *****
20 **          DIRECTORY          **
30 ** by OLIVER IHNS in 1985 **
40 ** for   C P C 4 6 4   **
50 ** COPYRIGHT '85 BY O. IHNS **
60 *****
70
80 ++++++++ Titelbild ++++++++
90
100 SYMBOL AFTER 250
110 :SYMBOL 251,129,66,36,126,90,126,90,
102
120 SYMBOL 252,24,24,255,255,231,126,60,
36
130 a$=CHR$(251):b$=CHR$(252)
140 MODE 0:INK 0,0:INK 1,24:BORDER 0:PEN
1:PAPER 0:INK 2,20,0
150 PEN 6:x=1:y=1:FOR y=1 TO 21 STEP 1:L
OCATE x,y+1:PRINT"COPYRIGHT (c) '85 by":S
OUND 2,y*50,7:LOCATE x,y:PRINT"
":NEXT
160 x=0:y=23:FOR x=1 TO 4:LOCATE x,y:PRI
NT" ":LOCATE x+1,y:PEN 1:PRINT"Oliver Ih
ns":SOUND 1,x*478,5:NEXT
170 x=1:y=13:FOR x=1 TO 2 :LOCATE x+1,y:
PEN 6:PRINT a$:PEN 12:PRINT"  DIRECTORY
":LOCATE x+1,y+1:PEN 13:PRINT b$:LOCATE

```

```

x,y:PRINT " ":LOCATE x,y+1:PRINT " ":NEX
T
180 PEN 0:PAPER 0:
190 PEN 7: LOCATE 1,25:PRINT"CPC 464 CP
C 464 C"
200 FOR x%=0 TO 250 STEP 3
210 FOR y%=0 TO 14 STEP 1
220 IF TEST (x%,y%) THEN PLOT x%+165,y%*
4+336,3:IF TEST (X%,Y%) THEN PLOT X%+169
,Y%*4+340,4
230 NEXT
240 NEXT
250 LOCATE 1,8:PEN 2:PRINT"> BITTE EINE
TASTE <":CALL &BB18
260 CLS:MODE 1:INK 0,1:INK 1,24:INK 2,18
:INK 3,6:PAPER 0
270 LOCATE 1,1:PEN 3:PRINT STRING$(40,21
6)
280 LOCATE 12,2:PEN 1:PRINT"D I R E C T
O R Y"
290 LOCATE 1,4:PEN 2:PRINT STRING$(40,CH
R$(251)):LOCATE 1,5:PEN 3:PRINT STRING$(
40,208)
300 PEN 2:LOCATE 1,7:PRINT CHR$(24):" Di
eses Programm hilf bei der Bearbei-
310 LOCATE 1,8:PRINT" tung von Disketten
"

```

```

320 LOCATE 1,10:PRINT" Man kann mit dies
em Programm das      "
330 LOCATE 1,11:PRINT" Directory bearbei
ten. Man kann Pro-   "
340 LOCATE 1,12:PRINT" gramme aus dem Di
rectory loeschen oder "
350 LOCATE 1,13:PRINT" einzelne Files um
bennen (RENAMEN)    "
360 LOCATE 1,14:PRINT"          VIE
L SPASS              ";CHR$(24)
370 LOCATE 1,16:PEN 3:PRINT CHR$(150);ST
RING$(38,154);CHR$(156)
380 FOR i=17 TO 22:LOCATE 1,i:PRINT CHR$
(149):LOCATE 1,23:PRINT CHR$(147):LOCATE
2,23:PRINT STRING$(38,154):NEXT:FOR i=1
7 TO 22:LOCATE 40,i:PRINT CHR$(149):LOCA
TE 40,23:PRINT CHR$(153):NEXT:

390 LOCATE 2,19:PEN 3:PRINT STRING$(38,1
54):LOCATE 2,21:PEN 3:PRINT STRING$(38,1
54)
400 LOCATE 5,15:PEN 1:PRINT CHR$(24);"Co
pyright (c)'85 by Oliver Ihns";CHR$(24)
410 LOCATE 2,18:PEN 2:PRINT"1:Loeschen
      : 2: Renamen"
420 LOCATE 2,20:PEN 2:PRINT"3:Directory
anz.      : 4: End"
430 LOCATE 2,25:PEN 1:INPUT"BITTE WAEHLE
N SIE ![1-4] ",wahl
440 '+++++++ Abfrage ++++++
450 ON wahl GOTO 460,610,860,960
460 '+++++++ LOESCHEN +++++
470 CLS:LOCATE 16,2:PEN 3:PRINT"LOESCHEN
":GOTO 580
480 LOCATE 1,3:PEN 2:PRINT STRING$(40,20
8)
490 LOCATE 1,5:PEN 1:INPUT"Welches Progr
amm soll geloescht werden ---> ";na$
500 LOCATE 1,8:PEN 2:PRINT"Soll das Prog
ramm ---> ";na$;"wirklich geloescht werd
en "":LOCATE 27,9:INPUT"";j$
510 IF antwort$="N" OR antwort$="n" THEN
GOTO 260 ELSE IF antwort$="J" OR antwort
$="j" THEN GOTO 520
520 LOCATE 1,5:PRINT STRING$(40," "):
LOCATE 1,6:PRINT STRING$(40," "):
LOCATE 1,7:PRINT STRING$(40," "):
LOCATE 1,8:PRINT STRING$(40," "):
LOCATE 1,9:PRINT STRING$(40," ")
530 LOCATE 1,5:PEN 3:PRINT"Bitte legen S
ie die Diskette ein und druecken Sie
dann eine Taste:_":CALL &BB18
540 !ERA,@na$
550 LOCATE 1,8:PEN 2:PRINT" F E R T I G "
560 LOCATE 1,9:PEN 1:INPUT"Soll noch ein
Programm geloescht werden ---> [j/n]";w
$
570 IF w$="J" OR w$="j" THEN 470 ELSE IF
w$="N" OR w$="n" THEN 260
580 LOCATE 1,18:PEN 2:PRINT"WICHTIG:--->
Es muss zuerst der Pro- gramname, d
ann ein Punkt und dann Art des Prog
rammes eingegeben werden."
590 LOCATE 1,24:PEN 3:PRINT"Beispiel ---

```

```

> Alien.bas"
600 GOTO 480
610 '+++++++ Renamen ++++++
620 CLS
630 LOCATE 1,1:PEN 1:PRINT"RENAMEN"
640 LOCATE 1,3:PEN 1:PRINT STRING$(40,15
4)
650 LOCATE 1,4:PEN 1:PRINT"So muss einge
geben werden ---> flug2.bas"
660 LOCATE 1,5:PEN 2:PRINT"Es muss zurer
st der Filename eingegeben werden, dann
ein Punkt und Art des Files"
670 LOCATE 1,8:PEN 3:INPUT"Alter Filenam
e:";a$
680 LOCATE 1,10:PEN 2:INPUT"Neuer Filena
me:";b$
690 LOCATE 1,8:PRINT STRING$(40," ")
700 LOCATE 1,10:PRINT STRING$(40," ")
710 LOCATE 1,8:PEN 2:PRINT"Please insert
disc to be renamed in drive A then
press any key:-":CALL &BB18
720 LOCATE 13,10:PEN 1:PRINT CHR$(150);S
TRING$(13,154);CHR$(156)
730 FOR i=11 TO 22:LOCATE 13,i:PRINT CHR
$(149):LOCATE 27,i:PRINT CHR$(149):NEXT
740 LOCATE 13,23:PRINT CHR$(147);STRING$
(13,154);CHR$(153)
750 FOR a=1 TO 360
760 PLOT 310+20*COS(a), 150+20*SIN(a),3
770 NEXT a
780 FOR a=1 TO 360
790 PLOT 310+8*COS(a), 220+20*SIN(a),3
800 NEXT
810 LOCATE 6,24:PEN 1:PRINT"Moment bitte
, es wird renamed !"
820 !REN,@b$,@a$
830 LOCATE 1,24:PRINT STRING$(40," ")
840 LOCATE 1,24:PEN 3:PRINT"FERTIG !!":F
OR x=1 TO 2000:NEXT:LOCATE 1,24:PEN 2:IN
PUT"Soll noch etwas renamed werden ";d$
850 IF d$="N" OR d$="n" THEN 260 ELSE IF
d$="J" OR d$="j" THEN 610
860 '+++++++ Directory ++++++
870 CLS
880 LOCATE 16,1:PEN 3:PRINT"Directory"
890 LOCATE 1,3:PEN 1:PRINT STRING$(40,15
4)
900 LOCATE 1,5:PEN 2: !DIR
910 LOCATE 1,20:PEN 1:PRINT"Bitte eine T
aste zum weitermachen":CALL &BB18
920 FOR i=4 TO 21:LOCATE 1,i:PRINT STRIN
G$(40," "):NEXT
930 LOCATE 1,5:PEN 3:INPUT"Wieder zum Me
nue zurueck ";w$
940 IF w$="N" OR w$="n" THEN LOCATE 1,5:
PRINT STRING$(40," "):LOCATE 1,5:PEN 2:
!DIR:GOTO 910
950 IF w$="J" OR w$="j" THEN 260
960 '+++++++ End ++++++
970 CLS
980 LOCATE 19,1:PEN 1:PRINT"End"
990 LOCATE 1,3:PEN 2:PRINT STRING$(40,"
")
1000 LOCATE 1,6:PEN 3:INPUT"Soll das Pro

```

tips & tricks

```
gramm wirklich beendet
1010 IF f$="J" OR f$="j" THEN 1030
```

```
1020 IF f$="N" OR f$="n" THEN 260
1030 CALL 0 ' Programm-Ende
```

CHECK V1:PRUEFSUMMEN:	190- 200: 38565	430- 440: 29065	670- 680: 84800	890- 900: 24335
ZEILENR.: SUMMEN	210- 220: 14301	450- 460: 41102	690- 700: 19043	910- 920: 98371
10- 20: 34847	230- 240: 352	470- 480: 43049	710- 720: 75541	930- 940: 72277
30- 40: 37201	250- 260: 71544	490- 500: 37585	730- 740: 77395	950- 960: 48575
50- 60: 45962	270- 280: 38755	510- 520: 97704	750- 760: 55782	970- 980: 9960
70- 80: 27916	290- 300: 23751	530- 540: 55659	770- 780: 8943	990- 1000: 37735
90- 100: 2601	310- 320: 51822	550- 560: 64071	790- 800: 47564	1010- 1020: 51339
110- 120: 52996	330- 340: 64910	570- 580: 93211	810- 820: 25399	1030- : 18822
130- 140: 55201	350- 360: 59118	590- 600: 53849	830- 840: 60594	
150- 160: 42694	370- 380: 82540	610- 620: 23103	850- 860: 82487	
170- 180: 54883	390- 400: 4286	630- 640: 29350	870- 880: 23286	
	410- 420: 118728	650- 660: 89211		

GESAMTCHECKSUMME: 58526

Tips & Tricks für den Schneider

Der Joystick am CPC

Bei vielen Lesern scheint die Abfrage des Joysticks immer noch ein Problem darzustellen. Daher wollen wir an dieser Stelle einmal kurz beschreiben, wie eine Abfrage richtig funktioniert.

Meistens erfolgt die Abfrage mittels der Befehlsfolge:

```
IF JOY(0)=2 THEN.....
```

```
IF JOY(0)=1 THEN.....
```

usw.

Das ist zwar im Prinzip richtig, erlaubt aber nur die Bewegung des Joysticks in 4 Richtungen. Zudem muß der Joystick exakt in die jeweilige Richtung bewegt werden. Gleichzeitiges Bewegen und Schieben ist nicht möglich.

Wird aber bei der Abfrage statt des Gleichheitszeichens die AND-Funktion angewandt, kann mit 5 Befehlen die Steuerung in 8 Richtungen erfolgen, wobei gleichzeitige Betätigung des Feuerknopfes möglich ist.

Ein Beispiel:

A ist die Variable für die Waagerechte.

B für die Senkrechte

Die Befehlsfolge lautet:

```
10 IF JOY(0) AND 1 THEN B=B-1
```

```
20 IF JOY(0) AND 2 THEN B=B+1
```

```
30 IF JOY(0) AND 4 THEN A=A-1
```

```
40 IF JOY(0) AND 8 THEN A=A+1
```

```
50 IF JOY(0) AND 16 THEN GOTO.....:REM
```

FEUERKNOPF

```
760 LOCATE 35,25:PRINT grad;"←"geschw:LOCATE 37,25:PRINT". "
```

Verbesserung zum Programm „Superfile“ aus Heft 7/85

Einer unserer kritischsten Programm-Anwender – Herr Hummel aus Winterbach – hat nach regem Briefwechsel mit unserer Programmierabteilung einen Verbesserungsvorschlag zum Programm „Superfile“ aus Heft 7/85. Das bisherige Programm war zwar voll funktionsfähig, bei Falscheingaben konnten jedoch Fehler auftreten. Diese Änderung schließt nun solche Fehler aus. D. h. im Klartext, daß

bei falschen Eingaben oder Bedienungsfehler keine Datenverluste oder Programmabbrüche auftreten können. Die Redaktion bedankt sich bei Herrn Hummel für die Geduld und Hartnäckigkeit bei der Durchsetzung dieses Verbesserungsvorschlages.

Wir hoffen, daß alle Anwender nun eine optimale Datenverwaltung verwirklichen können.

```
390 IF UPPER$(wahl$)="J" THEN 400 ELSE 1
50
900 hdat=19000\laenge:DIM inhalt$(hdat)
1240 IF a$=CHR$(127) THEN FOR felder=1 T
O hfelder:inhalt$(datensaezte)="":NEXT:d
atensaezte=datensaezte-2:CLS:GOTO 1090
1970 LOCATE 1,10:INPUT"Nummer des Feldes
, nachdem sortiert werden soll";sfelder
1980 IF sfelder<1 THEN sfelder=1 ELSE IF
sfelder>hfelder THEN sfelder=hfelder
3500 inhalt$(datensaezte)=LEFT$(inhalt$(
datensaezte),anf(felder))+LEFT$(inhalt$,
laenge(felder))+MID$(inhalt$(datensaezte
),(anf(felder)+laenge(felder)+1),ABS(LEN
(inhalt$(datensaezte)-(anf(felder)+laen
ge(felder))))
```

Korrektur zu „Two Cherry Chaps“

In unserer Ausgabe Heft 12/85 ist in dem Programm „Two Cherry Chaps“ ein Symbol nicht ausgedruckt worden. Bitte fügen

Sie die Änderung in Zeile 760 nachträglich ein und Ihr Programm funktioniert.

Suche Hardware

◆◆◆◆◆
Wer verkauft Commodore Floppy-Laufwerk 1541 mit C-16 Anschluß? Zuschriften an: Holger Schabio, Fritz-Goßler-Str. 2, 8671 Leupoldsdgrün.
 ◆◆◆◆◆

◆⇒⇒ **SUCHE BILLIG!!!** ⇐⇐⇐
 Floppy-1541 oder 1570!!!
 Biete 350 DM oder 280 Fr.!!!
 Suche laufend die neuesten Prgr. auf Disk!!! Angebote an: Desarzens Renaud, Ringstraße 56, CH-3327 Lyssach, ☎ 034/4537 08, (Vorwahl für Schweiz: 0041.

Wer hat **billig** 16K-Speichererweiterung für VC-20 anzubieten? Bitte melden bei D. Lienau, Falkenhagener Str. 8, 1000 Berlin 20. Suche auch Kontakte zu anderen VC-20 und C-64 Besitzern, die mir beibringen können Programme zu schreiben. Ich bin 21 Jahre.

Suche Sprachmodul + Software auf Kassette + preiswerte Floppy für C-64. Tel. 05 11/75 0128.

⇒ **Suche Drucker für CPC 464!**
 ☎ 023 73/7 10 75.

Suche Erweiterungsmodul für VC-20. Angebote an R. Weigelt, 2909 Boesel, Lessingstraße 6.

Welcher Drucker kann **POST-KARTEN bedrucken?**
 Suche Dt. und engl. Adventures für CPC-464! Bitte melden. Albert Bähr, Germanenstr. 6, 7470 Albstadt 15, Tel.: 074 31/7 45 61.

Suche für VC-20 32K Erweiterung schaltbar evtl. 64 K.
 Angebote an D. Keiterer, Gempenweg 3, CH-4707 Ettingen, Tel. 0 61/73 76 97.

Suche für CPC 464 DRUCKER.
 ⇒ EILT ⇒ EILT ⇒ EILT ⇒ EILT ⇒ EILT
 Frank Sigel, In den Stuben 7, 7312 Kirchheim-Teck, Tel. 070 21/67 24.

Durch unseren Kleinanzeigenmarkt erreichen Sie eine Vielzahl von Lesern! Nutzen Sie Ihre Chance!

Suche Software

Achtung ☆ Achtung ☆ Achtung
 ★ Suche für C-16 und VZ- ★
 ☆ 200 Software. Angebote ☆
 ★ bitte an Axel Waszynski, ★
 ☆ Erfurter Str. 33 a, 2351 ☆
 ★★ Trappenkamp. ★★

★★ **Commodore 16/116** ★★
 Suche Programme aller Art!!!
 Spiele (a. Compute mit), Adventure, Tips u. Tricks (auch Listings) usw. Angebote an:
 Andreas Schreier, Zur Tauberquelle 8, 8801 Wetztrungen, Tel. 098 69/277.

⇒⇒ **SCHNEIDER CPC 464** ⇐⇐
 Suche Programme aller Art
 Zinsb., Börsen- und Wertpapierprogramme, Spiele, Adventure etc.) Bin auch am Tausch von Programmen interessiert.
 Wer verkauft günstig Zubehör, Drucker für CPC 464. Angebote u. Anfragen: U. Hollenberg, von-Witzleben-Straße 13, 4400 Münster.

Suche Input 64 Cassetten Nr.: 1, 2, 3, 4 ---Zahle 7 DM (nur Originalprg.) Suche Floppy bis 25 DM (1541).
 Suche Original Ghostbusters u. A View to a kill.

Suche Spiele für CPC 464 u C-64 zum Tauschen. Liste an: Andreas Meyer (A. M. SOFT) Dennerstr. 100, 7000 Stuttgart 50.

Suche Sprachsynthesizer-Programm auf Kassette für C-16/C-116.
 Tausche gegen 5 Spielprogramme. Schreibt an: Norbert Schmitz, Spanische Schanzen 10 b, 4134 Rheinberg 4. Außerdem verkaufe ich Computer ORIC 1, 16 K mit Spiele und Kassettenrecorder für 100 DM oder ruft an. Tel.: 028 43/29 69.

Suche jede Art von Spielsoftware für den Schneider CPC 664. Biete gute Preise!!!
 Jörn B. Gruhlke, Postfach 1208, 2270 Wyk auf Föhr, Tel.-Nr. 04681/34 44, ab 18.00 Uhr.

Biete Software

★★★ **C 64 Software** ★★★
 ★ Arcade Hall of Fame (D) 49,- ★
 ★ Little Comp. People (D) 52,90 ★
 ★ Kennedy Approach (D) 55,- ★
 ★ Fourth Protocol 32,90/49,- ★
 ★ Winter Games 33,90/39,- ★
 ★ The Goonies (K) 34,90 ★
 ★ Software schon ab 7,95/11,90 ★
 ★ Hotline: 073 07/62 30 18-20 Uhr ★
 ★ CS Postfach 1304 7913 Senden ★

★★ **SCHNEIDER Software** ★★
 ★ Everyone's a Wally K 32,- ★
 ★ Frank Bruno's Boxing K 29,90 ★
 ★ Hacker K 38,- ★
 ★ Gremlins K 35,- ★
 ★ Jewels of Babylon K 19,90 ★
 ★ Red Moon K 22,90 ★
 ★ Supersoftware schon ab 9,90 ★
 ★ Hotline: 073 07/62 30 18-20 Uhr ★
 ★ CS Postfach 1304 7913 Senden ★

SONDERANGEBOT!!! 60 gr.-Papier ws Rand perf. 2000 Bl. nur DM 39,-. Weitere Angebote gegen DM 1,- in Briefm. Versand nur gegen Vorkasse. V-Kosten DM 4,-, ab DM 100,- frei.
 Fa. Leichtlein, Holzhofallee 32, 6100 Darmstadt.

CPC-464. Tolle Textverarbeitung plus Adressverwaltung mit deutscher Tastatur für nur 30,- DM. Cassetten einlegen und Drucker NLQ 401 einschalten, schon wird der ges. Inhalt ausgedruckt mit Cassettdruckprogramm für nur 10,- DM. Lieferung gegen Vorkasse plus 2,- DM Porto.
 Hans Thielen, Am Hoverkamp 48, 4044 Kaarst 1, Tel.: 02101-60 42 78.

Schneider CPC · Schneider CPC Spiel- und Anwenderprogramme aus Eigenentwicklung für Ihren CPC zu zivilen Preisen erhalten Sie bei Friedrich Neuper, Postfach 72, 8473 Pfreimd. Fordern Sie einfach das kostenlose Info an.

Commodore 16 ★ Commodore Verkaufte Spiele, Tips und Tricks, Anwenderprogramme und ganz tolle Supergrafiken. Infos kostenlos bei Hatteler, Werner, 4981 Reichersber/Österreich.

Supertape für C-16/116/Plus 4. Cassetten Save- und Laderoutine. mit 3600 oder 7200 Baud. Cassette mit Bed. 15 DM. Tel.: 02162/5 43 35.

C-16/C-116/Plus 4 Superspiele! Info 1 und 2 gegen Rückporto bei Hannes Kaltenbach, Prielmayerstr. 16, 7990 Friedrichshafen 1.

VC-20 + Datas. + 20 Spielen + Tasche + Staubsch. + Listings.
C-64 (fast neu) + Datas. + 25-30 Sp. + Joy. + Heft mit Listings.
C-16 (fast neu) + Datas. + 10-20 Sp. + Joy. + Listings.
ATARI 2600 + Joy. + 2 Spielen.
 Verkäufe an Meistbietenden. An D. Lienau, Falkenhagener Str. 8, 1000 Berlin 20. 100% Antwort für Jeden.

SCHNEIDER-CPC ☆ Diverse Programme in Originalverp. mit Handbuch auf Disketten 3" bzw. Cassetten wegen Systemwechsel günstig abzugeben, Liste anfordern bei W. Kiefer, PF. 44, 7801 Umkirch.

Diamond-Hunter

für Schneider CPC 464

Diamantensuche in den Höhlen der Anden

Steinschlag-Harry ist bei seiner Arbeit in einer Diamantenmine in einen verborgenen Schacht gefallen und entdeckt ein riesiges, unberührtes Diamantenvorkommen.

Geblendet von soviel Schönheit und in der Hoffnung der reichste Mann der Welt werden zu können, beginnt er ohne jegliche Vorsichtsmaßnahme zu berücksichtigen, mit der Bergung der Diamanten.

Doch schon nach kurzer Zeit muß er feststellen, daß das gar nicht so einfach ist. Das Berühren der umherliegenden radioaktiven Steine führt sofort zum Verlust eines seiner drei Leben. Auch herabfallende

Steine oder Diamanten verletzen Harry tödlich und darüberhinaus werden die Diamanten noch von einem Höhlenmonster bewacht.

Diese Boulder-Dash-Variante hat einen eigenen Editor mit dem man Spielfelder verändern oder neu erstellen kann. Diese Spielfelder können natürlich auch abgespeichert und eingeladen werden, wobei die aktuelle HI-Score-Liste mit abgespeichert wird.

Die Steuerung der Spielfigur und des Cursors beim Editor erfolgt über Joystick oder über die Cursortasten.

Hinweise zum Abtippen:

Teil 1:

Sie können die Zeilen 780-1310 weglassen. Sie müssen sich dann jedoch ein eigenes Spielfeld aufbauen und einladen. Bitte beachten Sie, daß die erste Zeile die Zeilennummer 10 und die letzte Zeile die Nummer 1560 sein muß.

Teil 2:

Die erste Zeilennummer muß 5 lauten. Das Programm müssen Sie unter dem Filenamen „HUNTER2“ abspeichern. Bei einem Neustart muß „GOTO 5“ verwandt werden, da sonst die Variablen gelöscht werden.

```

10 GOTO 30
11 :
12 "*****
13 "
14 "   D I A M O N D   H U N T E R   *
15 "
16 "   (C) 1985 by Andromeda-Soft   *
17 "
18 "   written by Oliver Heggelbacher *
19 "
20 "           f o r   T R O N I C   *
21 "
22 "*****
23 :
24 :           T E I L   1
25 :
30 PRINT:PRINT "Please wait!"
40 DEFINT a-z
50 '
60 '   Hi-Score Tabelle
70 '
80 DIM hi(10),hi$(10)
90 RESTORE 100:FOR t=1 TO 10:hi(t)=1100-
t*100:READ f:READ n$:hi$(t)=CHR$(15)+CHR
$(f)+n$+CHR$(15)+CHR$(2):NEXT t
100 DATA 3,"Vorsicht Steinschlag"
110 DATA 2,"Es wirken mit:"
120 DATA 1,"Steinschlag Harry"
130 DATA 1,"(mit 2 Double)"
140 DATA 1,"Diamanten Monster"
150 DATA 1,"radioaktive Steine"
160 DATA 1,"154 Diamanten"
170 DATA 2,"-----"

```

```

180 DATA 1,"   Im Verleih der"
190 DATA 3,"AS-Software Company"
200 :
210 :
220 EVERY 80 GOSUB 230:GOTO 270
230 BORDER INT(RND*18)+6:IF INKEY$<>" "
THEN RETURN ELSE IF zp=0 THEN RETURN ELS
E 1530
240 '
250 '   Melodie einlesen
260 '
270 DIM so(144,1)
280 FOR t = 1 TO 131:READ so(t,0):NEXT t
290 DATA 179,0,142,0,179,0,119,0,179,0
300 DATA 89,0,95,106,119,106,119,134,142
,134,142,159
310 DATA 179,0,142,0,119,0,142,0,89,0,11
9,0
320 DATA 71,60,67,60,71,60,67,60,71,60,6
7,60
330 DATA 89,71,80,71,89,71,80,71,89,71,8
0,71
340 DATA 106,89,95,89,106,89,95,89,106,8
9,95,89
350 DATA 127,0,159,0,106,0,127,0,89,0,10
6,0
360 DATA 80,71,80,89,95,89,95,106,119,10
6,119,134
370 DATA 142,0,106,119,127,119,127,142,1
59,142,159,179
380 DATA 190,179,190,213,239,0,119,127,1
19,0,190,0
390 DATA 179,0,119,0,190,0,119,0,213,0,1

```

```

27,0
400 DATA 119
410 FOR t = 1 TO 131:READ so(t,1):NEXT t
420 DATA 0,0,0,0,0,0,0,0,0,0
430 DATA 0,0,716,0,568,0,716,0,478,0,716
,0
440 DATA 358,0,379,426,478,426,478,536,5
68,478,568,638
450 DATA 358,0,284,0,239,0,284,0,179,0,2
39,0
460 DATA 142,119,134,119,142,119,134,119
,142,119,134,119
470 DATA 179,142,159,142,179,142,159,142
,179,142,159,142
480 DATA 213,179,190,179,213,179,190,179
,213,179,190,179
490 DATA 253,0,319,0,239,0,319,0,190,0,2
39,0
500 DATA 179,159,179,190,213,190,213,239
,253,239,253,284
510 DATA 319,0,239,253,284,253,284,319,3
58,319,358,379
520 DATA 426,379,426,478,319,358,379,358
,319,0,648,0
530 DATA 478
540
550 Zeichen definieren
560
570 SYMBOL AFTER 240
580 SYMBOL 240,16,56,80,56,20,16,40,104
590 SYMBOL 241,16,56,16,124,16,16,40,40
600 SYMBOL 242,16,56,20,56,80,16,40,44
610 SYMBOL 243,24,102,202,145,198,170,10
2,24
620 SYMBOL 244,0,16,56,108,56,16,0,0
630 SYMBOL 245,85,170,85,170,85,170,85,1
70
640 SYMBOL 246,32,32,255,4,4,4,255,32
650 SYMBOL 247,0,0,0,0,0,0,0,126
660 SYMBOL 248,4,10,17,34,100,120,112
670 SYMBOL 249,0,0,30,42,22,28,0,0
680 SYMBOL 250,0,28,34,78,86,38,24,0
690 SYMBOL 251,60,82,165,211,219,111,61
700 SYMBOL 252,62,87,195,131,163,154,94,
32
710 SYMBOL 253,126,130,129,129,129,129,6
6,60
720 SYMBOL 254,126,203,16,32,16,8,211,12
6
730 SYMBOL 255,60,126,90,126,102,90,126,
219
740
750 Bilder einlesen
760
770 DIM bild (32,23,6,1),ab(6),x(20),y(2
0),x2(20),y2(20),abf(20)
780 FOR t = 0 TO 1
790 FOR i=1 TO 23:READ a$
800 ps=0:FOR k=1 TO 32:a=VAL(MID$(a$,k,1
)):bild(k,i,t,0)=a:bild(k,i,t,1)=a:ps=ps
+a:NEXT k:READ ps2:IF ps<>ps2 THEN PRINT
:PRINT"Fehler in Zeile ";840+t*240+i*10:
END
810 NEXT i,t
820 ab(0)=i:ab(1)=1

```

```

830
840 Bild 1
850 DATA 01111112111111111111111112211111
1,34
860 DATA 11111113111111111111111122211111
1,37
870 DATA 1112211111222221111211113311111
1,44
880 DATA 1111221111113311111111112210221
1,41
890 DATA 1111311121111111111111111111131
1,37
900 DATA 11111111111000000011112111111
1,26
910 DATA 444444444444444444444444112211111
1,94
920 DATA 1111111111111111111111113311211
1,37
930 DATA 1112211111112321111111111111311
1,40
940 DATA 1112311121122122111111111111111
1,40
950 DATA 1112111211113113111111111111111
1,38
960 DATA 1113111111111111111111111111111
1,34
970 DATA 1221112111111111111111111111111
1,35
980 DATA 11311111111444444444444444444444
4,97
990 DATA 111111211100000003330000000000
0,21
1000 DATA 12211111111221111113112211111
11,40
1010 DATA 113111111111311111111111311211
11,39
1020 DATA 111221121111112211112111111311
11,40
1030 DATA 111131131111111311113112211111
11,42
1040 DATA 1111111111111111111111111311121
11,35
1050 DATA 111111111221111121111111111131
11,37
1060 DATA 11111111113112111111122311111
11,39
1070 DATA 1111111111111111111111111111111
11,32
1080 Bild 2
1090 DATA 00000221111111211122212222221
11,38
1100 DATA 11222131111222231111112311311
11,48
1110 DATA 113112111111311111111111111122
21,40
1120 DATA 1111111111111111111111111111113
11,34
1130 DATA 4444444444444444444444444444111
11,113
1140 DATA 11111111111111211111111214121
11,38
1150 DATA 1111222111111131112111314131
11,45
1160 DATA 1111311222111111111311114112

```

programme

```
11,43
1170 DATA 122111111311111111112111214113
11,43
1180 DATA 113111111111111111113111314111
22,43
1190 DATA 11211111111111111111111114121
13,39
1200 DATA 111110000000000000000000004131
11,16
1210 DATA 122144444444444444444444444121
00,102
1220 DATA 11311111111111111111211114122
21,41
1230 DATA 111111112111111111113112114131
31,43
1240 DATA 1111111311122211111113114111
11,42
1250 DATA 11111112111132111112122114122
11,44
1260 DATA 112111113111111111113113214123
11,46
1270 DATA 11312111122211111112111324122
11,47
1280 DATA 11113111113111111111311121123
11,42
1290 DATA 1111111111111111111111131212
11,36
1300 DATA 1111111111111111111111111313
11,36
1310 DATA 00000000000000000000000011111111
11,10
1320 '
1330 ' Schriftfarben und
1340 '
1350 ' Huellkurven definieren
1360 '
1370 INK 0,0:INK 1,0:INK 2,0:INK 3,11
1380 ENV 1,2,7,1,1,-1,1,12,-1,10
1390 ENV 2,1,15,1,15,-1,10
1400 ENV 3,11,1,1,11,-1,6
1410 ENT 1,10,5,1,10,-5,1
1420 '
1430 ' Einladebild
1440 '
1450 PAPER 0:MODE 1:FOR t=1 TO 20:a$=INK
EY$:NEXT t:zp=1
1460 PEN 3:LOCATE 1,25:PRINT "Diamond Hu
nter":LOCATE 27,25:PRINT "Andromeda-Soft
":LOCATE 11,1:PRINT "Press Space to abo
rt"
1470 LOCATE 15,23:PRINT "Please wait!"
1480 FOR t=0 TO 223 STEP 2:FOR i=0 TO 15
STEP 2:a=TEST(t+416,i):IF a=3 THEN PLOT
74+t*2+i,320+i*2,1:DRAWR 4,0:DRAWR 0,2,
2:DRAWR 2,0:PLOT 76+t*2+i,322+i*2,1:DRAW
R 4,0:DRAWR 0,2,2:DRAWR 2,0
1490 NEXT i,t
1500 LOCATE 13,8:PEN 2:PRINT "P R E S E
N T S:"
1510 FOR t=0 TO 223 STEP 2:FOR i=0 TO 15
STEP 2:a=TEST(t,i):IF a=3 THEN FOR k=0
TO 7 STEP 2:w=i*4+k:PLOT 26+t*2.5+w/2,15
0+w,1-(k MOD 4=0):DRAWR 4,0:NEXT k
1520 NEXT i,t
1530 LOCATE 11,1:PRINT SPACE$(20):LOCATE
```

```
1,25:PRINT SPACE$(40):,INK 3,11,17:LOCA
TE 14,23:PEN 3:PRINT "STILL LOADING"
1540 r=REMAIN(0):BORDER 17:LOCATE 1,1
1550 INK 1,11:INK 2,24:PEN 2
1560 CHAIN MERGE "!hunter 2",5,DELETE 10
-1560
```

Teil 2:

```
5 GOTO 2770
6 "
7 " Diamond Hunter -- Teil 2
8 "
9 :
10 '
20 ' S P I E L
30 '
40 LOCATE 3,24:INPUT "Mit welchem Bild w
ollen sie beginnen (0-6) ",a
$:IF (VAL(a$)<1 AND a$<>"0") OR VAL(a$)>
6 THEN 40
50 b=VAL(a$)-1
60 IF ab(b+1)=0 THEN 40
70 '
80 ' Bildaufbau
90 '
100 CLS:LOCATE 1,1:PRINT CHR$(150);STRIN
G$(32,154);CHR$(158);STRING$(5,154);CHR$(
156);
110 FOR t=2 TO 24:PRINT CHR$(149):LOCATE
34,t:PRINT CHR$(149);STRING$(5,32);CHR$(
149);:NEXT t
120 PRINT CHR$(147);STRING$(32,154);CHR$(
155);STRING$(5,154);CHR$(153);
130 WINDOW#1,2,33,2,24:WINDOW#2,35,39,2,
24
140 PAPER#1,0:CLS#1:PAPER#2,1:CLS#2:PEN#
2,0
150 sc=0:lives=3
160 x=1:y=1:x2=1:y2=1:au=1
170 SPEED INK 25,25
180 b=b+1
190 IF b>6 THEN b=0
200 IF ab(b)=0 THEN 180
210 WINDOW SWAP 0,1
220 FOR t=0 TO 20:abf(t)=0:NEXT t
230 LOCATE 1,1:v=0:dm=0:dm2=0
240 FOR t=1 TO 23:FOR i=1 TO 32
250 a=bild(i,t,b,0):IF a=0 THEN PRINT "
":GOTO 300
260 IF a=1 THEN PEN 1:PRINT CHR$(245);:G
OTO 300
270 IF a=2 THEN PEN 1:PRINT CHR$(243);:G
OSUB 310:GOTO 300
280 IF a=3 THEN PEN 3:PRINT CHR$(244);:d
m=dm+1:GOSUB 310:GOTO 300
290 PEN 2:PRINT CHR$(246);
300 NEXT i,t:GOTO 330
310 IF t=23 THEN RETURN ELSE IF bild(i,t
+1,b,0)<>0 THEN RETURN ELSE abf(v)=a:x(v
)=i:x2(v)=i:y(v)=t+1:y2(v)=t:v=v+1:IF v>
20 THEN v=0
320 RETURN
330 zeit=150:zu=0:ende=0:mx=32:my=23:mx2
=mx:my2=my:mon=1
```

```

340 za=0
350 LOCATE#2,1,3:PRINT#2,"Score":LOCATE#
2,1,8:PRINT#2,"Time":LOCATE#2,1,13:PRINT
#2,"Lives"
360 LOCATE#2,6-lives,15:PRINT#2,STRING$(
lives,241)
370 FOR t=0 TO 15 STEP 2:FOR i=0 TO 15 S
TEP 2:PLOT i+16,t+368,-(TEST(i+624-lives
*16,t+144)=0)*2:PLOT i+624-lives*16,t+14
4,1:NEXT i:SOUND 1,200-t*8,1,7:NEXT t
380 PEN 2:LOCATE 1,1:PRINT CHR$(22)+CHR$(
1):LOCATE 32,23:PRINT CHR$(255);:LOCATE
1,1:PRINT CHR$(22)+CHR$(0)
390 EVERY 50 GOSUB 950
400 EVERY 13,1 GOSUB 630
410 '
420 ' Herabfallen der Steine und
430 '
440 '           Diamanten
450 '
460 FOR zv=0 TO 1 STEP 0:za=za+1:IF za>2
0 THEN za=0
470 IF ende=1 THEN ende=0:r=REMAIN(0)+RE
MAIN(1):WINDOW SWAP 0,1:GOSUB 1450:GOSUB
1350:GOSUB 1300:GOTO 170
480 IF ak=1 THEN r=REMAIN(1)+REMAIN(0):G
OTO 1090
490 IF abf(za)=0 THEN e=za ELSE zv=1
500 NEXT zv
510 gx=x(za):gy=y(za):gx2=x2(za):gy2=y2(
za)
520 DI:LOCATE gx2,gy2:PRINT " ";:LOCATE
gx,gy:PEN 1-(abf(za)=3)*2:PRINT CHR$(241
+abf(za));
530 bild(gx2,gy2,b,0)=0:bild(gx,gy,b,0)=
abf(za):EI
540 DI:IF gx=x AND gy=y THEN r=REMAIN(0)
+REMAIN(1):EI:GOTO 1090 ELSE EI
550 IF mon=1 THEN IF gx=mx AND gy=my THE
N GOSUB 1260
560 a=bild(gx,gy-2,b,0):IF a=2 OR a=3 TH
EN DI:IF abf(a)=0 THEN abf(a)=a:x(a)=gx:
x2(a)=gx:y(a)=gy2:y2(a)=gy2-1:EI ELSE GO
SUB 1040:GOTO 560
570 IF gy=23 THEN S90
580 a=bild(gx,gy+1,b,0):IF a=0 OR(x=gx A
ND y-1=gy)THEN y(za)=gy+1:y2(za)=gy:GOTO
460
590 abf(za)=0:e=za:SOUND 132,0,0,0,2,0,3
1:GOTO 460
600 '
610 '           bewegen der Spielfigur
620 '
630 IF x=mx THEN IF y=my THEN IF mon=1 T
HEN ak=1:r=REMAIN(1):RETURN
640 IF(INKEY(0)>-1 OR INKEY(72)>-1)AND y
>1 THEN y=y-1:au=1:GOTO 690
650 IF (INKEY(1)>-1 OR INKEY(75)>-1) AND
x<32 THEN x=x+1:au=2:GOTO 690
660 IF (INKEY(2)>-1 OR INKEY(73)>-1) AND
y<23 THEN y=y+1:au=1:GOTO 690
670 IF (INKEY(8)>-1 OR INKEY(74)>-1) AND
x>1 THEN x=x-1:au=0:GOTO 690
680 IF x=gx AND y=gy THEN 710

```

```

690 IF bild(x,y,b,0)=3 THEN sc=sc+5:dm=d
m-1:dm2=dm2+1:SOUND 130,284,0,0,3:bild(x
,y,b,0)=0:GOSUB 870
710 IF x=x2 AND y=y2 THEN ve=ve+1:IF ve>
5 THEN ve=0:GOTO 760 ELSE RETURN
730 IF bild(x,y,b,0)=4 THEN x=x2:y=y2:RE
TURN
740 ve=0
750 SOUND 1,3000,2,6
760 c=bild(x2,y2-1,b,0):IF c=2 OR c=3 TH
EN IF y=y2+1 THEN FOR t=0 TO 20:IF abf(t
)=c AND x(t)=x AND y(t)=y2 THEN 765 ELSE
NEXT t
761 IF c=2 OR c=3 THEN IF abf(e)=0 THEN
abf(e)=c:x(e)=x2:x2(e)=x2:y(e)=y2:y2(e)=
y2-1 ELSE GOSUB 1040:GOTO 761
762 IF c=2 OR c=3 THEN IF za<21 THEN IF
abf(za)=0 THEN za=e
765 IF bild(x,y,b,0)=2 THEN ak=1:r=REMAI
N(1):RETURN
770 IF x=1 THEN IF y=1 AND zu=1 THEN LOC
ATE 1,1:PRINT CHR$(22)+CHR$(1)+CHR$(241)
+CHR$(22)+CHR$(0):ende=1
780 IF bild(x2,y2,b,0)=0 OR bild(x2,y2,b
,0)=1 THEN LOCATE x2,y2:PRINT " ";:bild(
x2,y2,b,0)=0
790 x2=x1:y2=y
800 IF ende=1 THEN 830
810 LOCATE x,y:PEN 2:PRINT CHR$(240+au);
820 IF zu=0 THEN IF x>1 OR y>1 THEN zu=1
:LOCATE 1,1:PEN 3:PRINT CHR$(254)
830 RETURN
840 '
850 '           Diamant in Matrix enthalten ?
860 '
870 FOR t=0 TO 20:IF x2(t)=x THEN IF y2(
t)=y THEN IF abf(t)=3 THEN abf(t)=0
880 NEXT t:RETURN
890 RETURN
900 '
910 '           Zeit und Punkte anzeigen
920 '
930 '           und Monster bewegen
940 '
950 zeit=zeit-1:IF zeit=0 THEN ak=1:r=RE
MAIN(0)+REMAIN(1):RETURN
960 SPEED INK zeit/10+10,zeit/10+10
970 LOCATE#2,1,5:PRINT#2,USING"#####";sc
:LOCATE#2,3,10:PRINT#2,USING"###";zeit
980 IF mon=0 THEN RETURN
990 mx=mx+(x<mx)-(x>mx):my=my+(y<my)-(y>
my):DI:LOCATE mx2,my2:a=bild(mx2,my2,b,0
):IF a=1 THEN PEN 1:PRINT CHR$(245);ELSE
IF a=2 THEN PEN 1:PRINT CHR$(243);ELSE
IF a=3 THEN PEN 3:PRINT CHR$(244);ELSE I
F a=4 THEN PEN 2:PRINT CHR$(246);ELSE PR
INT" ";
1000 SOUND 130,319,20,5,0,1:LOCATE 1,1:P
RINT CHR$(22)+CHR$(1):LOCATE mx,my:PEN 2
:PRINT CHR$(255);:LOCATE 1,1:PRINT CHR$(
22)+CHR$(0):EI:mx2=mx:my2=my:RETURN
1010 '
1020 '           Freie Matrix suchen
1030 '
1040 FOR t=0 TO 20:IF abf(t)=0 THEN e=t:

```


programme

```
RETURN
1050 NEXT t:abf(za)=0:e=za:RETURN
1060 '
1070 ' Spielertod
1080 '
1090 SOUND 1,0,0,0,2,0,10
1100 LOCATE x2,y2:PRINT " ";
1110 PEN 2:FOR t=1 TO 5:LOCATE x,y:PRINT
CHR$(t+248);:FOR i=1 TO 500:NEXT i:NEXT
t
1120 v=xi:w=y:GOSUB 1130:v=mx:w=my:GOSUB
1130:GOTO 1190
1130 LOCATE v,w:PRINT " ";:LOCATE v,w
1140 a=bild(v,w,b,0):IF a=1 THEN PEN 1:P
RINT CHR$(245);
1150 IF a=2 THEN PEN 1:PRINT CHR$(243);
1160 IF a=3 THEN PEN 3:PRINT CHR$(244);
1170 IF a=4 THEN PEN 2:PRINT CHR$(246);
1180 RETURN
1185 FOR t=0 TO 20:IF abf(t)=0 AND bild(
x(t),y(t)+1,b,0)=0 THEN y(t)=y(t)+1:y2(t
)=y
1187 NEXT t
1190 zeit=150:mon=1:x=1:x2=1:y=1:y2=1:zu
=0:ak=0:mx=32:my=23:mx2=mx:my2=my:lives=
lives-1:IF lives<1 THEN 2370
1200 GOTO 370
1210 '
1220 ' Stein oder Diamant ist
1230 '
1240 ' auf Monster gefallen
1250 '
1260 SOUND 130,0,0,0,2,0,8:DI:v=mx:w=my:
PEN 2:FOR t=1 TO 5:LOCATE v,w:PRINT CHR$(
t+248);:FOR i=1 TO 100:NEXT i:NEXT t:GO
SUB 1130:mon=0:EI:sc=sc+30:RETURN
1270 '
1280 ' Spielfeld erneuern
1290 '
1300 FOR i=1 TO 23:FOR t=1 TO 32:bild(t,
i,b,0)=bild(t,i,b,1):NEXT t,i
1310 RETURN
1320 '
1330 ' Alle Diamanten eingesammelt
1340 '
1350 IF dm<>0 THEN RETURN
1360 PEN 2:LOCATE 3,10:PRINT "Alle Diama
nten eingesammelt!"
1370 LOCATE 11,15:PRINT "Bonus : ";dm2*5
1380 sc=sc+dm2*5:LOCATE #2,1,5:PRINT #2,
USING "#####";sc
1390 RETURN
1400 '
1410 ' Melodie fuer beendete Runde
1420 '
1430 ' oder HI-Score
1440 '
1450 FOR t=0 TO 12:IF so(t,0)<>0 THEN SO
UND 129,so(t,0),0,0,1
1460 FOR i=1 TO 220:NEXT i:NEXT t:RETURN
1470 '
1480 ' Spielfelder erstellen
1490 '
1500 CLS:WINDOW#1,1,32,1,23:PAPER#1,0:CL
S#1
```

```
1510 LOCATE 1,23
1520 WINDOW#2,1,40,24,25:PAPER#2,1:PEN#2
,0:CLS#2
1530 WINDOW #3,33,40,1,23:PAPER #3,1:PEN
#3,2:CLS #3
1540 WINDOW SWAP 0,1
1550 INPUT #2,"Welches Bild wollen sie a
endern (0-6) ";bn
1560 IF bn<0 OR bn>6 THEN 1550
1570 CLS#2:LOCATE 1,1:SPEED INK 25,25
1580 IF ab(bn)=0 THEN c1=1:GOTO 1660
1590 FOR t=1 TO 23:FOR i=1 TO 32:z=bild(
i,t,bn,0)
1600 IF z=0 THEN PRINT " ";:GOTO 1650
1610 IF z=1 THEN PEN 1:PRINT CHR$(245);:
GOTO 1650
1620 IF z=2 THEN PEN i:PRINT CHR$(243);:
GOTO 1650
1630 IF z=3 THEN PEN 3:PRINT CHR$(244);:
GOTO 1650
1640 PEN 2:PRINT CHR$(246);:GOTO 1650

1650 NEXT i,t
1660 WINDOW SWAP 0,3:LOCATE 1,3:PRINT "1
":PRINT "beenden ":PRINT "2:":PRINT "CL
S"
1670 PRINT:PRINT"3: ";CHR$(245):PRINT"4:
";CHR$(243):PRINT"5: ";PEN 3:PRINT CHR
$(244)
1680 PEN 2:PRINT "6: ";CHR$(246)
1690 PRINT:PRINT"7:":PRINT"loeschen"
1700 WINDOW SWAP 0,3
1710 RESTORE 1720:FOR t = 0 TO 4:READ si
gn(t),col(t):NEXT t
1720 DATA 32,1,245,1,243,1,244,3,246,2
1730 x=2:y=1:x2=2:y2=1
1740 a$=INKEY$: IF a$<CHR$(49) OR a$>CH
R$(55) THEN 1860
1750 IF a$="1" THEN 1970
1760 IF a$="2" THEN 1960
1770 IF a$="3" THEN z=1:c1=0
1780 IF a$="4" THEN z=2:c1=0
1790 IF a$="5" THEN z=3:c1=0
1800 IF a$="6" THEN z=4:c1=0
1810 IF a$="7" THEN z=0
1820 bild(x,y,bn,0)=z:bild(x,y,bn,1)=z:L
OCATE x,y:PEN col(z):PRINT CHR$(sign(z))
;:SOUND 1,119,2,5
1830 x2=x:y2=y
1840 x=x+1:IF x>32 THEN x=1:y=y+1:IF y>2
3 THEN y=23:x=32
1850 FOR t=1 TO 20:NEXT t:GOTO 1740
1860 IF (INKEY(75)>-1 OR INKEY(1)>-1)THEN
IF x<32 THEN x=x+1:GOTO 1910 ELSE x=1-(
y=23)*31:y=y+1+(y=23):GOTO 1910
1870 IF (INKEY(73)>-1 OR INKEY(2)>-1)AND
y<23 THEN y=y+1:GOTO 1910
1880 IF (INKEY(74)>-1 OR INKEY(8)>-1)THEN
IF x>1 THEN x=x-1:GOTO 1910 ELSE x=32:y
=y-1-(y=1):GOTO 1910
1890 IF (INKEY(72)>-1 OR INKEY(0)>-1)AND
y>1 THEN y=y-1:GOTO 1910
1900 GOTO 1930
1910 IF x=1 AND y=1 THEN x=x2:y=y2
1920 a=bild(x2,y2,bn,0):LOCATE x2,y2:PEN
```

```

col(a):PRINT CHR$(sign(a));
1930 LOCATE 1,1:PRINT CHR$(22)+CHR$(1):L
OCATE x,y:PEN 2:PRINT CHR$(24B);:LOCATE
1,1:PRINT CHR$(22)+CHR$(0)
1940 x2=x:y2=y
1950 FOR t=1 TO 100:NEXT t:GOTO 1740
1960 FOR t=1 TO 23:FOR i=1 TO 32:bild(i,
t,bn,0)=0:bild(i,t,bn,1)=0:NEXT i,t:CLS:
cl=1:FOR t=1 TO 20:a$=INKEY$:NEXT t:GOTO
1730
1970 SOUND 2,2B4,0,0,3
1980 ab(bn)=1-cl:GOTO 2770
1990 '
2000 ' Bilder einladen
2010 '
2020 CLS:PEN 2:INPUT"Filename ";n$:IF LE
N(n$)>13 THEN n$=LEFT$(n$,13)
2030 PRINT:PRINT "Druecken sie die PLAY-
Taste"
2040 CLOSEIN:IF n$<>" " THEN OPENIN"!"+n
$+"-dh" ELSE OPENIN "!"
2050 PRINT:PRINT "File ";CHR$(24);n$;CHR
$(24);" wird ";"geladen"
2060 INPUT #9,bn:ab(bn)=1
2070 FOR i=1 TO 23:FOR t=1 TO 32 STEP 4
2080 INPUT #9,w:wi=INT(w/125):r1=w*MO
D 125:w2=INT(r1/25):r2=r1 MOD 25:w3=INT(
r2/5):w4=r2 MOD 5
2090 bild(t,i,bn,0)=wi:bild(t,i,bn,1)=w1
2100 bild(t+1,i,bn,0)=w2:bild(t+1,i,bn,1
)=w2
2110 bild(t+2,i,bn,0)=w3:bild(t+2,i,bn,1
)=w3
2120 bild(t+3,i,bn,0)=w4:bild(t+3,i,bn,1
)=w4
2130 NEXT t,i
2140 FOR t=1 TO 10:INPUT#9,hi(t),hi$(t):
NEXT t
2150 CLOSEIN:GOTO 2770
2160 '
2170 ' Bilder abspeichern
2180 '
2190 PEN 2:CLS:INPUT"Welches Bild soll a
bgespeichert werden (0-6) ";bn
2200 IF ab(bn)=0 THEN PRINT:PRINT "Bild
ist leer!!!":FOR t=1 TO 3000:NEXT:GOTO 2
190
2210 IF bn<0 OR bn>6 THEN 2190
2220 PRINT:INPUT"Filename ";n$:IF LEN(n$
)>13 THEN n$=LEFT$(n$,13)
2230 PRINT:PRINT"Druecken sie REC & PLAY
, danach SPACE"
2240 WHILE INKEY(47)=-1:WEND
2250 SPEED WRITE 0
2260 OPENOUT"!"+n$+"-dh"
2270 PRINT:PRINT"File ";CHR$(24);n$;CHR$
(24);" wird ";"abgespeichert"
2280 PRINT #9,bn
2290 FOR i=1 TO 23:FOR t=1 TO 32 STEP 4
2300 a=bild(t,i,bn,0)*125+bild(t+1,i,bn,
0)*25+bild(t+2,i,bn,0)*5+bild(t+3,i,bn,0
)
2310 PRINT#9,a:NEXT t,i
2320 FOR t=1 TO 10:PRINT#9,hi(t):WRITE#9
,hi$(t):NEXT t

```

```

2330 CLOSEOUT:GOTO 2770
2340 '
2350 ' Spielende
2360 '
2370 WINDOW SWAP 0,1
2380 FOR t=1 TO 23:FOR i=1 TO 32:bild(i,
t,b,0)=bild(i,t,b,1):NEXT i,t
2390 ORIGIN 320,0:FOR i=0 TO 2 STEP 2:FO
R t=0 TO 320 STEP 4:MOVE t+i,0:DRAW t+i,
400,0:MOVE -t-i,0:DRAW -t-i,400:NEXT t,i
2400 SPEED INK 25,25:LOCATE 12,2:PEN 3:P
RINT"S P I E L E N D E"
2410 PEN 2:LOCATE 10,5:PRINT"Ihre Punktz
ahl:";USING"#####";sc
2420 FOR t=1 TO 20:a$=INKEY$:NEXT t
2430 IF sc<hi(10) THEN 2570
2440 LOCATE 6,9:PRINT"Gratulation! Ihr E
rgebnis ist":LOCATE 11,11:PRINT"unter de
n besten 10!"
2450 GOSUB 1450
2460 LOCATE 10,14:PEN 3:PRINT"Ihren Name
n bitte..."
2470 PEN 2:LOCATE 10,16:PRINT STRING$(20
,247)
2480 nam$="" : x=0
2490 a$=INKEY$:IF a$=""OR a$=CHR$(34)THE
N 2490
2500 IF a$=CHR$(127)THEN IF x>0 THEN LOC
ATE 9+x,16:PRINT CHR$(247):x=x-1:nam$=LE
FT$(nam$,x):GOTO 2490 ELSE GOTO 2490
2510 IF a$=CHR$(13) THEN 2540
2520 IF x<20 THEN x=x+1:LOCATE 9+x,16:PR
INT a$:nam$=nam$+a$
2530 GOTO 2490
2540 IF sc>hi(10) AND sc<=hi(9) THEN hi(
10)=sc:hi$(10)=nam$
2550 FOR t=9 TO 1 STEP -1
2560 IF sc>hi(t) THEN hi(t+1)=hi(t):hi$(
t+1)=hi$(t):hi(t)=sc:hi$(t)=nam$ : NEXT
t
2570 LOCATE 14,25:PEN 3:PRINT"SPACE drue
cken"
2580 WHILE INKEY(47)=-1:WEND
2590 GOTO 2770
2600 '
2610 ' HI-Score Tabelle zeigen
2620 '
2630 CLS:PEN 3:LOCATE 7,1:PRINT"Diamond
Hunter --- TOP TEN"
2640 PEN 2
2650 LOCATE 7,2:PRINT STRING$(28,"-")
2660 FOR t=1 TO 10
2670 LOCATE 5,2+t*2:PRINT USING"##";t:P
RINT". ";hi$(t)
2680 LOCATE 30,2+t*2:PRINT": ";USING"####
#"hi(t)
2690 NEXT t
2700 PEN 1:LOCATE 14,25:PRINT"SPACE drue
cken"
2710 EVERY 9 GOSUB 3050
2720 WHILE INKEY(47)=-1:WEND
2730 r=REMAIN(0)
2740 '
2750 ' Titelbild
2760 '

```

programme

```

2770 MODE 1:INK 0,0:PAPER 0:CLS:BORDER 1
1
2780 PRINT CHR$(22)+"1";TAB:PL0T -10,-1
0,3:FOR t=0 TO 10 STEP 2:MOVE 110+t,368-
t:PRINT "D I A M O N D H U N T E R";IN
EXT t:TABOFF:PRINT CHR$(22)+"0";
2800 PEN 2:LOCATE 16,8:PRINT CHR$(164);"
1985 by"
2810 PEN 1:LOCATE 14,10:PRINT"Andromeda-
Soft"
2820 PEN 2:LOCATE 10,13:PRINT"1...spiele
n"
2830 LOCATE 10,14:PRINT"2...Bilder erste
llen"
2840 LOCATE 10,15:PRINT"3...Bilder einla
den"
2850 LOCATE 10,16:PRINT"4...Bilder abspe
ichern"
2860 LOCATE 10,17:PRINT"5...High-Score T
abelle"
2870 LOCATE 1,24:PRINT SPACE$(10)+"Ihre
Punktzahl:";USING"#####";sc
2880 FOR t=1 TO 20:a$=INKEY$:NEXT t

```

```

2890 i=0:k1=14:k2=14:EVERY 9 GOSUB 3050
2900 FOR zv=0 TO 1 STEP 0
2910 k1=k1-1-(k1<14)
2920 k2=k2-1-(k2<14)
2930 INK 3,k1:INK 1,k2
2940 a$=INKEY$:IF VAL(a$)>0 THEN IF VAL(
a$)<6 THEN zv=1
2950 NEXT zv
2960 r=REMAIN(0):INK 1,15:INK 3,11,23
2970 IF a$="1" THEN 40
2980 IF a$="2" THEN 1500
2990 IF a$="3" THEN 2020
3000 IF a$="4" THEN 2190
3010 GOTO 2630
3020 '
3030 ' Melodie spielen
3040 '
3050 i=i+1:IF i>144 THEN i=1
3060 IF so(i,0)<>0 THEN k1=26:SOUND 130,
so(i,0),0,0,1
3070 IF so(i,1)<>0 THEN k2=26:SOUND 129,
so(i,1),0,0,1
3080 RETURN

```

CHECK V1:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN
10- 11:	371
12- 13:	29510
14- 15:	20956
16- 17:	43476
18- 19:	55528
20- 21:	16638
22- 23:	27965
24- 25:	23643
30- 40:	14846
50- 60:	17711
70- 80:	12364
90- 100:	55271
110- 120:	41239
130- 140:	39917
150- 160:	42385
170- 180:	34845
190- 200:	28838
210- 220:	6394
230- 240:	40749
250- 260:	18979
270- 280:	52687
290- 300:	75584
310- 320:	67216
330- 340:	77315
350- 360:	77419
370- 380:	102859
390- 400:	34722
410- 420:	57644
430- 440:	89026
450- 460:	95207
470- 480:	122918
490- 500:	95552
510- 520:	109417
530- 540:	1181
550- 560:	24933
570- 580:	19406
590- 600:	31580
610- 620:	39516
630- 640:	46347
650- 660:	25383
670- 680:	24215
690- 700:	47928
710- 720:	46335
730- 740:	26738
750- 760:	18551
770- 780:	33818
790- 800:	44397
810- 820:	21221

830- 840:	3392
850- 860:	69170
870- 880:	69242
890- 900:	69035
910- 920:	70163
930- 940:	68906
950- 960:	69138
970- 980:	70800
990- 1000:	68476
1010- 1020:	69234
1030- 1040:	69149
1050- 1060:	69451
1070- 1080:	37775
1090- 1100:	69746
1110- 1120:	68992
1130- 1140:	72258
1150- 1160:	69471
1170- 1180:	69392
1190- 1200:	69045
1210- 1220:	71932
1230- 1240:	69364
1250- 1260:	69660
1270- 1280:	69530
1290- 1300:	69286
1310- 1320:	34316
1330- 1340:	23892
1350- 1360:	33096
1370- 1380:	27303
1390- 1400:	16568
1410- 1420:	8477
1430- 1440:	11916
1450- 1460:	49321
1470- 1480:	35773
1490- 1500:	26431
1510- 1520:	44224
1530- 1540:	20813
1550- 1560:	27600

GESAMTCHECKSUMME: 15381

Teil 2:

CHECK V1:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN
5- 6:	1174
7- 8:	28671
9- 10:	386
20- 30:	4149
40- 50:	61172
60- 70:	19908

80- 90:	9519
100- 110:	96504
120- 130:	17704
140- 150:	32576
160- 170:	57007
180- 190:	20125
200- 210:	19091
220- 230:	78301
240- 250:	43432
260- 270:	70503
280- 290:	27785
300- 310:	22909
320- 330:	8945
340- 350:	45774
360- 370:	106633
380- 390:	57858
400- 410:	3873
420- 430:	39341
440- 450:	16486
460- 470:	68781
480- 490:	49170
500- 510:	12910
520- 530:	41813
540- 550:	32429
560- 570:	62382
580- 590:	16804
600- 610:	38004
620- 630:	42551
640- 650:	32623
660- 670:	56878
680- 690:	95544
710- 730:	74975
740- 750:	6052
760- 770:	53606
762- 765:	77098
770- 780:	75274
790- 800:	29642
810- 820:	43552
830- 840:	586
850- 860:	49293
870- 880:	41793
890- 900:	586
910- 920:	38327
930- 940:	30028
950- 960:	63261
970- 980:	37785
990- 1000:	44083
1010- 1020:	26079
1030- 1040:	14167
1050- 1060:	52608
1070- 1080:	10871
1090- 1100:	14371
1110- 1120:	91679
1130- 1140:	68748
1150- 1160:	49852
1170- 1180:	25194
1185- 1187:	34648
1190- 1200:	46951
1210- 1220:	31651
1230- 1240:	30640
1250- 1260:	18803
1270- 1280:	25183
1290- 1300:	38224
1310- 1320:	586
1330- 1340:	48879
1350- 1360:	25493
1370- 1380:	43170
1390- 1400:	586
1410- 1420:	45425
1430- 1440:	15526
1450- 1460:	46752
1470- 1480:	32652
1490- 1500:	13943
1510- 1520:	21294
1530- 1540:	24300
1550- 1560:	53294
1570- 1580:	47109
1590- 1600:	68400
1610- 1620:	67152
1630- 1640:	46284
1650- 1660:	10563
1670- 1680:	38705
1690- 1700:	18127
1710- 1720:	63038
1730- 1740:	38504
1750- 1760:	20395
1770- 1780:	62416
1790- 1800:	62528
1810- 1820:	48358
1830- 1840:	81014
1850- 1860:	77875
1870- 1880:	53533
1890- 1900:	48366
1910- 1920:	40355
1930- 1940:	48997
1950- 1960:	78018
1970- 1980:	32030
1990- 2000:	18232
2010- 2020:	45876
2030- 2040:	101098
2050- 2060:	45092
2070- 2080:	93023
2090- 2100:	60456
2110- 2120:	86526
2130- 2140:	4667
2150- 2160:	2483
2170- 2180:	25056
2190- 2200:	25348
2210- 2220:	60286
2230- 2240:	71662
2250- 2260:	11167
2270- 2280:	54491
2290- 2300:	33349
2310- 2320:	26260
2330- 2340:	2234
2350- 2360:	10026
2370- 2380:	3084
2390- 2400:	77951
2410- 2420:	92903
2430- 2440:	62234
2450- 2460:	43716
2470- 2480:	29509
2490- 2500:	105098
2510- 2520:	18238
2530- 2540:	44070
2550- 2560:	54253
2570- 2580:	38755
2590- 2600:	1525
2610- 2620:	34644
2630- 2640:	54317
2650- 2660:	13163
2670- 2680:	106936
2690- 2700:	31456
2710- 2720:	11644
2730- 2740:	5078
2750- 2760:	9143
2770- 2780:	51841
2800- 2810:	54889
2820- 2830:	62643
2840- 2850:	81716
2860- 2870:	57730
2880- 2890:	66611
2900- 2910:	34706
2920- 2930:	36688
2940- 2950:	16692
2960- 2970:	23890
2980- 2990:	21911
3000- 3010:	10241
3020- 3030:	17009
3040- 3050:	44477
3060- 3070:	35042
3080- :	201

GESAMTCHECKSUMME: 48671

Was gibts Neues auf dem Software-Markt?

ACTIVISION präsentiert eine mysteriöse Neuheit:

HACKER

„Diskette einlegen und Programm laden – alles andere müssen Sie selbst herausfinden.“

Das ist so ungefähr alles, was in der Anleitung als Hinweis zu finden ist. Darüber hinaus erfährt der User lediglich, daß er sich bei dem Versuch, eine Mailbox anzuwählen, in der Nummer geirrt hat und ganz plötzlich in ein wildfremdes Computernetz geraten ist. Auf dem Bildschirm ist lediglich LOG ON zu sehen. Was nun? Was tun? Das Passwort ist nicht bekannt. Der Name der Firma auch nicht. Es gibt auch keinerlei Hinweise, Regeln oder sonstige Backgroundinformationen. Aber es reizt natürlich, mehr über das fremde und geheimnisvolle Computersystem herauszufinden. Aber wie? Das muß man – wie ein echter Hacker – schon selbst herausfinden.

Der Anwender steckt nun mitten in einer für ihn unklaren Situation und muß sich ganz auf sein eigenes Geschick und seine

eigene Logik verlassen.

Entwickelt wurde HACKER von Steve Cartwright, der schon einige Spiele-Hits erdacht hat.

Erhältlich ist HACKER für Atari 800 XL, C-64, Schneider CPC 464, Sinclair Spectrum und Apple II.

Technische Besonderheiten

- ★ Mit Joystick und Keyboard zu spielen
- ★ Farbige Grafik mit Voll-Animation
- ★ Thematisch der echten „Hackerei“ angelehnt, wobei die Realität jedoch im Sinne des Spielens stark überzogen wurde.
- ★ Wie auch bei echten Hackern, ist es nicht möglich, eine Zwischenspeicherung durchzuführen. Wenn der Spieler einen Security-Check nicht passieren kann, wird er aus dem System herausgeworfen und muß von vorn beginnen.



WAY OF THE EXPLODING FIST Schneider CPC

Ein echter Renner. Lange Zeit Nummer Eins in den Charts. Fantastische Animation, Spielwitz, Action, die Spaß macht. Die Rede ist von „THE WAY OF THE EXPLODING FIST“ (Spectrum/C-64/Schneider). MELBOURNE HOUSE ist ein Meisterwerk gelungen, mit dem eigentlich nur noch IMAGINE'S „Ye Ar Kung-Fu“ mithalten kann.

Wir haben uns die Schneider-Version des „Exploding Fist“ zu Gemüte geführt. Herrlich die Graphic, der fernöstliche „Sound“

und der genaue Bewegungsablauf der kämpfenden Personen.

Sie treten gegen den Computer an:

Man beginnt als Neuling (Novice); hat man den Gegner bezwungen, erreicht man den Ersten Dan. Wer sehr gut „trainiert“, der bringt es zum Großen Meister, der schafft den „Weg zur explodierenden Faust“.

Hersteller: MELBOURNE HOUSE. Vertrieb: Weeske Software, Potsdamer Ring 10, 7150 Backnang

ACTIVISION veröffentlicht weitere Adaptionen von

Mindshadow

Im Herbst, genauer gesagt im Oktober und November, veröffentlichte ACTIVISION weitere Adaptionen ihres erfolgreichsten Adventures als Cassettenversionen für den C-64 und Schneider CPC 464.

MINDSHADOW ist fast schon ein „lebendiger Roman“, in dem hervorragende Grafiken mit einem spannenden Abenteuer voller Intrigen verbunden werden.

WER BIST DU?
WO BIST DU?
WAS WIRST DU JETZT TUN?

Du kannst Dich an nichts mehr erinnern. Du befindest Dich an einem wüstenähnlichen Strand. Aber an welchem?

Du willst Deine Identität um jeden Preis herausfinden. Dabei führt Dich die Reise rund um die Welt und bringt Dich der Wahrheit immer näher. Einer Wahrheit voller Intrigen und Gefahren.

Mit Hilfe des mysteriösen Condors kommst Du dem Verräter immer näher ...

Technische Besonderheiten:

- ★ über 80 Bilder in hochauflösender Grafik
- ★ Der enorme Wortschatz erlaubt eine großzügige Kommunikation – bis hin zu kompletten Sätzen.
- ★ Mit ausführlichem Demo-Programm
- ★ Help-Modus – es werden Hinweise gegeben, wenn der Spieler nicht mehr weiterkommt. Allerdings stellen diese Hinweise keine Lösungen dar.
- ★ Hint-Book erscheint demnächst

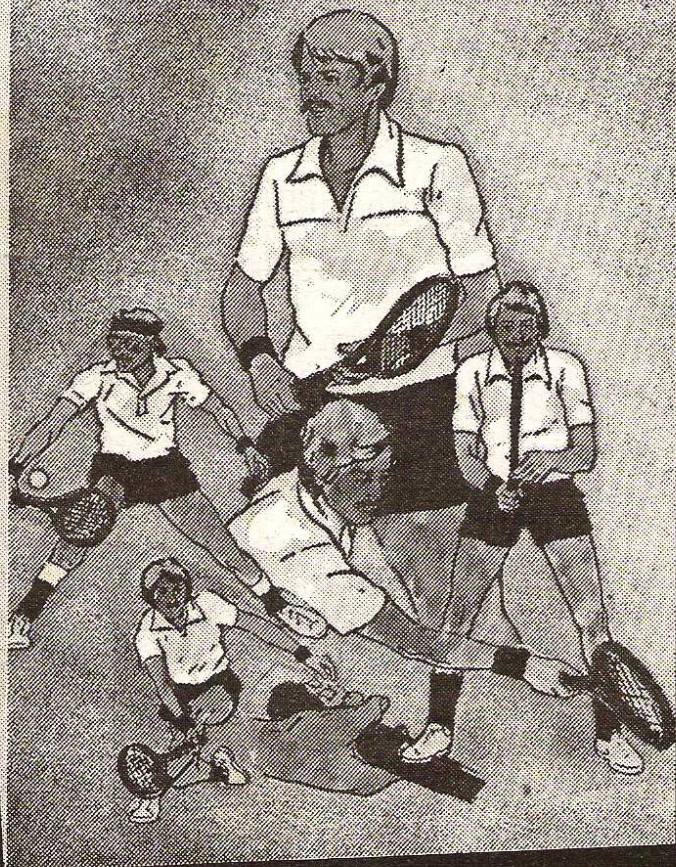
Hersteller:
Activision Home Computer Software
Postfach 70 06 80
2000 Hamburg 76

NEU!

THE GAMES

Commodore 64 DM 29,50

Tennis



Das TENNIS-Fieber hat uns alle ergriffen. Nicht zuletzt durch die großartigen Erfolge von Boris Becker ist der „Weiße Sport“ zu einem Volkssport geworden! In unserem Game treten zwei Tennis-Cracks gegeneinander an; sie schenken sich nichts; um jeden Ball wird gekämpft. Das Spiel basiert auf den Grundregeln des internationalen Tennis. Hart umkämpfte Tie-Breaks sind ebenso an der Tagesordnung wie ein zermürbendes Fünfsatz-Spiel. Sie haben hierbei die Möglichkeit, Ballgeschwindigkeit und Ballwinkel zu variieren. Aufgeschlagen wird per „Knopfdruck“. Ein Superspiel in 3 D-Effekt.

ZX-Spectrum DM 29,50

Witchcraft



Der alte Magier will einen Zaubertrank brauen. Dazu benötigt er allerlei Zutaten. Also ist er unterwegs, um Spinnen, Kraken, Pilze und andere Gegenstände einzusammeln. Doch Hexen, Fledermäuse und anderes Ungetier versuchen ihn daran zu hindern. Der Magier ist aber nicht hilflos! Die Monster kann er mit Zaubersternen abwehren. Hat er eine Zutat gefunden, so muß er sie in den Topf werfen. Ein Spaß für die ganze Familie.

THE GAMES - Software

★ SUPER! ★

Postfach · D-3444 Wehretal 1 · Tel. 056 51/406 43