

monatlich

Compute mit

COMMODORE & SCHNEIDER

12/86

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

Unabhängiges Magazin für Anwender von Commodore- und Schneider-Computern

3,80 DM
33 öS
3,80 SFr

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

C 64:

**IRQ-Directory
Joe the Ball**

C 16:

SPRITES!

VC 20:

Fight Night!

Schneider:

Tracer – Programmierhilfe

Lucky Luck

Neuer Checksummer

**Commodore:
A L L E
S T E U E R -
Z E I C H E N
J E T Z T I M
K L A R T E X T**

**Leser- + Meckerecke – Report – Software-Abo – Software-Service
Bücher – Werkstatt – Assembler-Kurs – Software-Reviews
Großer Kleinanzeigenmarkt – Galaxis – u.v.m.**

EDITORIAL

Lieber Leser!

Es weihnachtet sehr. Für den Computermarkt bedeutet das immer eine rasante Umsatzsteigerung in den Wintermonaten und eine Reihe von Neuheiten. Eine davon ist dabei, zum Mißverständnis des Jahres zu avancieren. Es handelt sich hierbei um den Schneider PC. Der Gesamteindruck in Presse und Werbung läßt nämlich vermuten, daß der Kunde einen neuen Computer erwerben kann. Und dazu noch mit dem vertrauten Namen der Firma Schneider. Tatsächlich aber ist der Schneider PC nur einer von vielen IBM-kompatiblen Computern und muß daher mit seinesgleiche verglichen werden. Wir im Verlag konnten jedenfalls die Euphorie um diesen PC nicht verstehen. Der Hardwareaufbau wurde von uns wirklich nicht als ideal empfunden. Viel zu kurze

Platinensteckplätze schränken die Erweiterungsmöglichkeiten stark ein, gelötete Bausteine erlauben nur umständliches Wechseln des Betriebssystems und die Stromversorgung aus dem Monitor erlaubt keine Variationen in der Hardwarekonfiguration. Was aber entscheidender wiegt: der günstige Preis wird auch von Konkurrenzanbietern erreicht, deren Produkte diese Nachteile nicht aufweisen. Wenn wir also in dieser Ausgabe über einen PC berichten, dann nicht um unsere Systempalette zu erweitern, sondern um unsere Leser über das Angebot aufzuklären.



Uwe Knierim
Chefredakteur

Die »andere Seite«

Hacker 8

Report

Orgatechnik in Köln 4

PC zu Homecomputerpreisen 6

Rubriken

Software-Service 32

Assemblerkurs (19) 40

NEU

Kurs: Interruptprogrammierung des C 64,	
1. Teil	42
Leser- und Meckerecke	59
Kleinanzeigen	59

Software

Commodore

Neu: Alle Grafikzeichen im Klartext	3
Ballon Hopper (VC20)	9
Fight Night (VC 20)	10
King of Kings (C 16/116)	12
Galaxis (C 16/116)	20
Sprites (C 16/116)	23
Atom (C 64)	27
Extended UMH (Grafik & Sound, C64)	35
IRQ-Directory (C 64)	39

Schneider

Alien Attack	43
Lucky Luck	48
Mouse	51
Neu: Stark verbesserte Checksummer	58
Tracer -	
Programmierhilfe für Maschinensprache	54

IMPRESSUM

"Compute mit"
ISSN 0179-6720
erscheint monatlich im
Tronic-Verlag, Am Stad 35
3440 Eschwege
Tel.: 05651/3 00 11



Redaktion:

Axel Credé (verantwortlich)
Chefredakteur: Uwe Knierim
Redakteure: Manfred Kleimann, Bernd Zimmermann, Frank Brall, Otfried Schmid, Thomas Brandt

Gesamtherstellung:

Druckhaus Dierichs Kassel, Frankfurter Str. 168,
3500 Kassel

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

Vertrieb

Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz:
Verlagsunion
6200 Wiesbaden
Friedrich-Bergius-Str. 20
Telefon: 06121/2660

Erscheinungsweise:

Erstverkaufstag von "Compute mit" jeweils Mitte des Monats.

Urheberrecht:

Alle in "Compute mit" veröffentlichten Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten.

Reproduktionen jeder Art (Fotokopien, Microfilm, Erfassung in Datenverarbeitungsanlagen usw.) bedürfen der schriftlichen Genehmigung des 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 und Bezeichnungen frei von Schutzrechten sind.

Bezugspreis:

Einzelheft 3,80 DM
Abonnement: Inland 42,- DM im Jahr (12 Ausgaben)
Ausland (Europa): 52,- DM
ohne Kassetten!

Programmierabteilung:

Montag - Freitag von 14-16 Uhr
Tel. (05651) 30013

Autoren, Manuskripte:

Der Verlag nimmt Manuskripte zur Veröffentlichung gerne entgegen.

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

Bei Zusendung von Manuskripten und Software erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger. Alle Einsendungen müssen frei von Rechten Dritter sein. Dies muß mit der Unterschrift des Einsenders bestätigt werden.

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

Kopierfähige Kassette oder Diskette mit dem Programm (Computerbezeichnung nicht vergessen!), von Drucker erstelltes Listing (keine Schreibmaschinenlistings!), evtl. Bildschirmfotos oder Hardcopies mit Demonstrationsbeispielen und ausführliche Programmbeschreibung (Erklärung der programmtechnischen Besonderheiten, Spielverlaufsbeschreibung). Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.

Anzeigenpreise:

Bitte Mediaunterlagen anfordern.

Anzeigenverwaltung:

Anzeigenleiter: Hartmut Wendt

Tronic-Verlag GmbH
Am Stad 35, 3440 Eschwege
Telefon: 05651/30011
Telefax: 05651/30011

Neu für alle Commodore Rechner C 64/VC 20 und C 16

Auf vielfachen Leserwunsch haben wir uns entschlossen, unser Druckverfahren für Commodore-Listings zu ändern. Steuerzeichen sowie alle Grafikzeichen werden zukünftig durch Klartext ersetzt. Der Ausdruck ist kompatibel zum bisherigen Druckverfahren, ein neuer Checksummer ist aus diesem Grund nicht erforderlich.

Eingabehinweise für alle Commodore Rechner!

Wer sich schon etwas näher mit den Commodore Rechnern befaßt hat, wird sicherlich wissen, daß der Grafiksatz des C 64 sehr viele ähnliche Symbole enthält. Aus diesem Grund ist es oft sehr schwer, einzelne Zeichen, wie zum Beispiel horizontale Linien, voneinander zu unterscheiden. Besonders macht sich dieser Nachteil in Listings bemerkbar, welche die Bildschirmgrafik mit PRINT-Zeilen aufbauen.

Bisher haben wir in unserem Ausdruck lediglich die Steuersequenzen wie CTRL-RVS ON oder CTRL-BLUE durch Klartext ersetzt, ab dieser Ausgabe werden jedoch auch alle Grafikzeichen durch ein Schlüsselcode ersetzt. Durch diese Methode sind Verwechslungen praktisch völlig ausgeschlossen.

Wie arbeitet unser Klartext?

Wer schon nach dem alten Verfahren Programme eingegeben hat, wird sicherlich keine Schwierigkeiten mit der Umstellung haben.

Für die neu hinzugekommenen Leser erkläre ich jedoch das Verfahren noch einmal grundlegend.

Alle sogenannten Steuerzeichen (z.B. eine Farbe) sowie ein Grafikzeichen, werden in unseren LISTINGS durch ein CODEWORT, welches die Taste bzw. Tastenkombination kennzeichnet, ersetzt. Eine Tabelle der Tastenfunktionen finden Sie in jedem Tronic-Magazin (z.B. Computronic o. Compute Mit). Das folgende Beispiel zeigt den Unterschied zwischen unserem alten und dem neuen Druckverfahren:

C64 LISTING MIT CHECKSUMMEN (C V1.0)

```
10 REM BISHERIGER AUSDRUCK                <11B>
20 PRINT " (DOHN4) |"                      <52>
30 PRINT " | | | | | | | | | | | | | | | |" <85>
40 PRINT " | | | | | | | | | | | | | | | |" <149>
50 PRINT "ENDE"                             <246>
ENDE DES LISTINGS
(LDRUCK: F.B.J)
```

C-64 LISTING + CHECKSUMMEN (C V1.0)

```
10 REM NEUER AUSDRUCK                      <162>
20 PRINT " (DOHN4 SD CY2 SP) "             <52>
30 PRINT " (SA SB SC SD SE SF SG SH SJ SK) " <85>
40 PRINT " (CA CB CC CD CE CF CG) "        <149>
50 PRINT "ENDE"                             <246>
ENDE DES LISTINGS
(LDRUCK: F.B./D.S.J)
```

Wie in dem Beispiel zu erkennen ist, werden alle Steuertasten durch die entsprechende Tastenbeschreibung markiert. Das Wort DOWN steht in diesem Fall für Cursor nach unten. Die Zahl dahinter gibt an wie oft die Taste betätigt werden muß. In unserem Beispiel müßte also die Taste CURSOR abwärts viermal betätigt werden. Die genaue Beschreibung, welche Taste gemeint ist, entnehmen Sie bitte der Klartext-Tabelle, welche am Ende dieser Beschreibung abgedruckt ist. Achtung für C16 und C64 existieren zwei verschiedene Tabellen!

Wie ebenfalls aus dem Beispiel zu erkennen ist, werden auch alle Grafikzeichen durch einen 2-Buchstaben-Code ersetzt. Der erste Buchstabe ist entweder ein »S« oder ein »C«. Das »C« steht für die »COMMODORE TASTE« und das »S« für die »SHIFT TASTE«. Der zweite Buchstabe bezeichnet die Aufschrift der Taste. Das »SA« in unserem Beispiel bedeutet also, daß die Taste SHIFT zusammen mit dem Buchstaben »A« betätigt werden muß. »C+« würde bedeuten, daß die Taste COMMODORE zusammen mit dem PLUS Zeichen gedrückt wird. Wird direkt hinter dieser Buchstabenkombination eine Zahl angegeben, so gibt diese die Anzahl der jeweiligen Zeichen an. Das Zeichen COMMODORE Y wurde also in unserem Beispiel 9 mal eingegeben. Durch diese Methode entfällt zukünftig das lästige Abzählen von mehrfachen Zeichen.

Sollten Sie einmal nicht wissen, welche Taste gemeint ist, so hilft Ihnen ein Blick in unsere Klartext-Tabelle weiter.

Um im Listing Tastenbezeichnungen von normalen Zeichen zu unterscheiden, werden alle Tastenkennzeichnungen in geschweifte Klammern gesetzt. Diese dürfen selbstverständlich nicht eingegeben werden. Auch Leerzeichen innerhalb geschweiften 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) dargestellt. Alle Zeichen außerhalb der 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 einmal an die in Klartext geschriebenen Zeichen gewöhnt hat, wird man den großen Vorteil dieser Schreibweise erkennen.

Frank Brall

Noch eine wichtige Anmerkung !

Aus technischen Gründen können drei Zeichen von unserm Ausgabegerät nicht verarbeitet werden. Diese Zeichen sind: Pfeil nach links, engl. Pfund, Pfeil nach oben

- Steht für den Pfeil nach links
- £ Steht für das englische Pfund-Symbol
- ^ Steht für den Pfeil nach oben

Klartext-Tabelle für Commodore 64 und VC 20

DOWN UP	CURSOR ABWÄRTS CURSOR HINAUF	TASTE NEBEN RECHTEN HIT-TASTE & TASTE NEBEN RECHTEM SHIFT
CLR	CLEAR SCHIRM	SHIFT-TASTE & 2. TASTE GANZ RECHTS OBEN
INST	EINFÜGEN	SHIFT-TASTE & TASTE GANZ RECHTS OBEN
HOME	CURSOR IN ECKE	2.TASTE VON GANZ RECHTS OBEN
DEL	DELETE	TASTE GANZ RECHTS OBEN
RIGHT	CURSOR RECHTS	TASTE GANZ RECHTS UNTEN
LEFT	CURSOR LINKS	SHIFT-TASTE & TASTE UNTEN RECHTS
SPACE	LEERZEICHEN	LEERTASTE (GRÖSSTE TASTE)
F1	FUNKTIONSTASTE	FUNKTIONSTASTE F1
F2	FUNKTIONSTASTE	FUNKTIONSTASTE F2
F3	FUNKTIONSTASTE	FUNKTIONSTASTE F3
F4	FUNKTIONSTASTE	FUNKTIONSTASTE F4
F5	FUNKTIONSTASTE	FUNKTIONSTASTE F5
F6	FUNKTIONSTASTE	FUNKTIONSTASTE F6
F7	FUNKTIONSTASTE	FUNKTIONSTASTE F7
F8	FUNKTIONSTASTE	FUNKTIONSTASTE F8
BLACK	SCHWARZ	CONTROL-TASTE & 1
WHITE	WEISS	CONTROL-TASTE & 2
RED	ROT	CONTROL-TASTE & 3
CYAN	TUERKIS	CONTROL-TASTE & 4
PURPLE	PURPUR	CONTROL-TASTE & 5
GREEN	GRÜN	CONTROL-TASTE & 6
BLUE	BLAU	CONTROL-TASTE & 7
YELLOW	GELB	CONTROL-TASTE & 8
RVS ON	INVERSE EIN	CONTROL-TASTE & 9
RVS OFF	INVERSE AUS	CONTROL-TASTE & 0
ORANGE	ORANGE	COMMODORE-TASTE & 1
BROWN	BRAUN	COMMODORE-TASTE & 2
LIG. RED	HELLROT	COMMODORE-TASTE & 3
DGREY	DUNKELGRAU	COMMODORE-TASTE & 4
MGREY	MITTELGRAU	COMMODORE-TASTE & 5
LIG. GREEN	HELLGRÜN	COMMODORE-TASTE & 6
LIG. BLUE	HELLBLAU	COMMODORE-TASTE & 7
HGREY	HELLGRAU	COMMODORE-TASTE & 8
CTRL ...	CTRL-ZEICHEN	CONTROL-TASTE UND DAS ... ZEICHEN
S ...	GRAFIKZEICHEN	SHIFT-TASTE UND DAS ... ZEICHEN
C ...	GRAFIKZEICHEN	COMMODORE-TASTE UND DAS ... ZEICHEN
SHIFTSPACE	UNSICHTBARER CODE	SHIFT-TASTE UND SPACE

Klartext-Tabelle für Commodore C16 und Plus 4

DOWN UP	CURSOR ABWÄRTS CURSOR HINAUF	TASTE NEBEN INST DEL; RECHTS OBEN
CLR	CLEAR SCHIRM	3. TASTE NEBEN DOWN VON RECHTS OBEN
INST	EINFÜGEN	TASTE ÜBER RETURN ZUSAMMEN MIT SHIFT
HOME	CURSOR IN ECKE	SHIFT-TASTE & TASTE GANZ RECHTS OBEN
DEL	DELETE	2. TASTE VON GANZ RECHTS OBEN
RIGHT	CURSOR RECHTS	TASTE GANZ RECHTS OBEN
LEFT	CURSOR LINKS	4. TASTE VON RECHTS OBEN
SPACE	LEERZEICHEN	5. TASTE VON RECHTS OBEN
F1	FUNKTIONSTASTE	LEERTASTE (GRÖSSTE TASTE)
F2	FUNKTIONSTASTE	FUNKTIONSTASTE F1 (VORHER DEFINIEREN)
F3	FUNKTIONSTASTE	FUNKTIONSTASTE F2 (VORHER DEFINIEREN)
F4	FUNKTIONSTASTE	FUNKTIONSTASTE F3 (VORHER DEFINIEREN)
F5	FUNKTIONSTASTE	FUNKTIONSTASTE F4 (VORHER DEFINIEREN)
F6	FUNKTIONSTASTE	FUNKTIONSTASTE F5 (VORHER DEFINIEREN)
F7	FUNKTIONSTASTE	FUNKTIONSTASTE F6 (VORHER DEFINIEREN)
F8	FUNKTIONSTASTE	FUNKTIONSTASTE F7 (VORHER DEFINIEREN)
BLACK	SCHWARZ	FUNKTIONSTASTE F8 (VORHER DEFINIEREN)
WHITE	WEISS	CONTROL-TASTE & 1
RED	ROT	CONTROL-TASTE & 2
CYAN	TUERKIS	CONTROL-TASTE & 3
PURPLE	PURPUR	CONTROL-TASTE & 4
GREEN	GRÜN	CONTROL-TASTE & 5
BLUE	BLAU	CONTROL-TASTE & 6
YELLOW	GELB	CONTROL-TASTE & 7
RVS ON	INVERSE EIN	CONTROL-TASTE & 8
RVS OFF	INVERSE AUS	CONTROL-TASTE & 9
ORNG	ORANGE	CONTROL-TASTE & 0
BRN	BRAUN	COMMODORE-TASTE & 1
YLGN	GELBGRÜN	COMMODORE-TASTE & 2
PINK	ROSA	COMMODORE-TASTE & 3
BLGRN	BLAUGRÜN	COMMODORE-TASTE & 4
LBLU	HELLBLAU	COMMODORE-TASTE & 5
DBLU	DUNKELBLAU	COMMODORE-TASTE & 6
LGRN	HELLGRÜN	COMMODORE-TASTE & 7
CTRL ...	CTRL-ZEICHEN	COMMODORE-TASTE & 8
S ...	GRAFIKZEICHEN	CONTROL-TASTE UND DAS ... ZEICHEN
C ...	GRAFIKZEICHEN	SHIFT-TASTE UND DAS ... ZEICHEN
FLASH ON	BLINKEN EIN	COMMODORE-TASTE UND DAS ... ZEICHEN
FLASH OFF	BLINKEN AUS	CONTROL-TASTE UND ...
SHIFTSPACE	UNSICHTBARER CODE	CONTROL-TASTE UND ...
		SHIFT-TASTE UND SPACE

Neues von der ORGA



Flightsimulator – PC auf dem Amiga – Commodores »Sidecar« macht's möglich.



Hervorragende Farbdigitalisierung der Firma Print-technik. Rechts das ASM-Emblem in schwarz/weiß auf dem ATARI-ST-Monitor.

Vom 16. bis zum 21. Oktober fand in Köln die 6. Orgatechnik in Köln statt. Nun wird man sich fragen, was die ASM von einer Messe für Büroorganisation erwartet. Tatsächlich wurden im Softwarebereich fast nur Produkte für die betriebliche Anwendung vorgestellt. Software für den Heimanwender war daher natürlich Mangelware. Aber die großen Konkurrenten ATARI und COMMODORE konnten es nicht lassen, ihre Gefechte auf der „Orga“ auszutragen. Und in dem Windschatten dieses Gefechtes kämpften auch einige Softwarehäuser tapfer mit.

So stellte Atari eine verbesserte Version des Textverarbeitungsprogramm „1st Word“ vor, welches auch Grafiken verarbeitet und in den Text einbaut. Niemand wird sich darüber wundern, daß ATARI diese Version „1st Word plus“ nannte. Ebensowenig ist man darüber erstaunt, daß auch

COMMODORE für seinen AMIGA ein Textverarbeitungsprogramm präsentieren konnte, daß Grafiken verarbeitet, wie auch alle erdenklichen Schriften und Schriftgrößen produziert, natürlich im Grafikmodus der jeweiligen Drucker. Leider war diese tolle Weiterentwicklung des bekannten Programms „VIZAWRITE“ auf der Messe nur als Demoversion vertreten. Die Programmierarbeiten waren bis zum Messestart noch nicht abgeschlossen. Ebenso war auf dem ATARI-Stand nur eine Demoversion des Grafik„blitters“ zu sehen, der den ST-Grafikaufbau um den Faktor 5 beschleunigen wird. Der Geschwindigkeitsgewinn war zwar beeindruckend, jedoch waren in der Grafik noch eindeutige

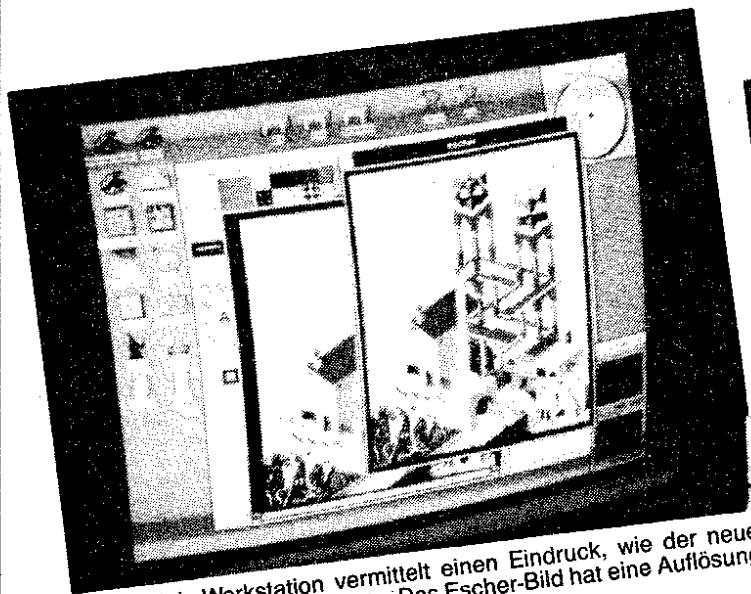
Programmierfehler zu erkennen. Ebenso weiß niemand bei ATARI, wie die Besitzer eines ST ihr Gerät werden aufrüsten können. Auch die Preisvorstellungen für den BLITTER liegen mit 300-400 DM noch weit über dem Erschwinglichen, wenn man bedenkt, daß der Umrüstsatz auch noch hinzu kommt. Der AMIGA hat natürlich ähnliche Problem. Der notwendige Speicherplatz kostet eine ganze Stange. Eine 2-Megabyte-Erweiterung kostet stolze 2000,-DM, soviel wie die ganze Grundfiguration (ohne Monitor). Allerdings hat COMMODORE aus dem Speicherplatzdesaster etwas gelernt und liefert den AMIGA ab sofort nur noch mit 512 KB-Speicher aus, die 256-K-Version war ja wohl ein schlechter Witz.

Der Softwarefluß ist bei beiden Computern nicht so, wie es sich der Anwender wünscht und wie es von den

Firmen angekündigt war. Natürlich hat die ATARI-Software aufgrund des Zeitvorteils dabei die Nase vorn. Langfristig wird es aber so aussehen, daß beide Computer nebeneinander bestehen und sich den Markt teilen werden, vor allem, wenn das Preisniveau noch mehr in Richtung Homecomputer rutscht.

Eine Frage weckte das besondere Interesse der Fachjournalisten auf der abendlichen Pressekonferenz bei ATARI. Die Frage nämlich, wie der die nächste ATARI-Computergeneration aussehen wird. Da ATARI auf der Suche nach UNIX-Programmierern ist, darf man davon ausgehen, daß ein UNIX-kompatibles oder ein UNIX-ähnliches System mit über 1000 Bildpunkten pro Y- und

TECHNIK in Köln?



Diese Unix-Workstation vermittelt einen Eindruck, wie der neue Atari einmal aussehen könnte. Das Escher-Bild hat eine Auflösung von über 1000x 1000 Bildpunkte.



Big-Boss Jack Tramiel beim abendlichen Atari-Pressetreffen im Gespräch mit Chef-Redakteur Uwe Knierim.

X-Achse auf uns zu kommen wird. Ob dieses 32-Bit-System auch ST-kompatibel sein wird, bleibt völlig offen, da sich ATARI diesbezüglich in Schweigen hüllt. Auch Jack Tramiel hatte während des Gespräches mit uns für unsere Frage nur ein verschmitztes Lächeln und ein „Which Unix?“ übrig.

Die eigentlich neue Nachricht von ATARI war dann die Ankündigung, daß ATARI im November an die Börse gehen wird und somit seine finanzielle Krise endgültig überwunden haben wird. Jack Tramiel unterstrich in unserem Gespräch noch einmal die große Bedeutung der Software für einen Rechner. Die Firma ATARI sei deshalb in einen Hardware- und einen Softwareteil gesplittet worden. In den Spielhallen gäbe es auch weiterhin Aktivitäten von

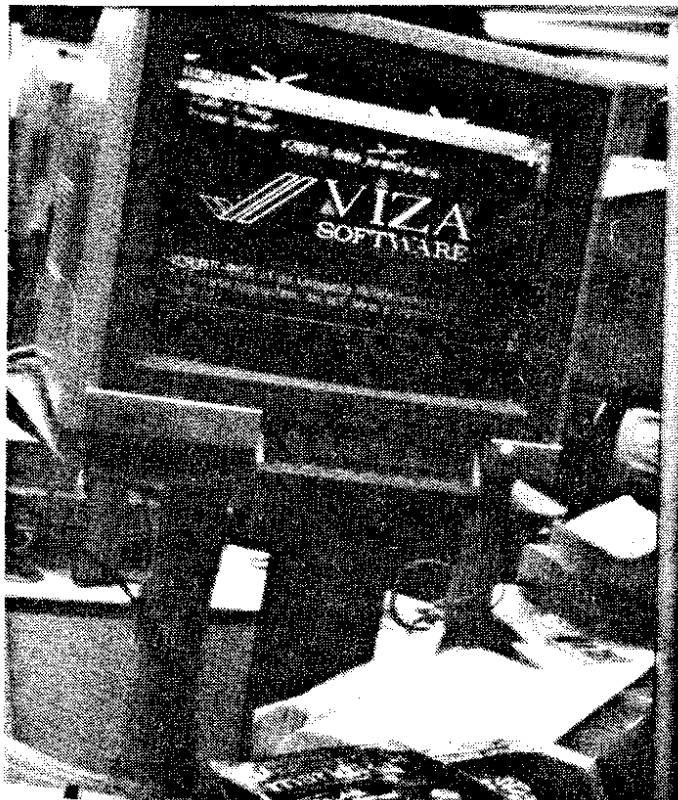
ATARI. An der Gestaltung des Hardwaremarktes habe daher auch eine Softwarezeitung einen gehörigen Anteil.

Für Programmierer präsentierte die Firma „Print Technik“ einen sehr ausgereiften Farb-Video-Digitizer auf dem ATARI-Stand. Das AMIGA-Gegenstück präsentierte auf dem Commodorestand seine fantastischen Farbbilder in einer Diashow.

Zusammenfassend kann man sagen, daß die ORGA keine Sensationen liefern konnte, da die Firmen offensichtlich auf der CeBIT in Hannover mit ihren Neuheiten vertreten sein wollen.

Ein Index für die Marktentwicklung ist die ORGA allemal gewesen.

(Uwe Knierim)



Bildschirmdarstellung = Ausdruck. VIZAWRITER für den AMIGA gab es leider nur in einer Demoversion.

"ECD Professional" - ein PC zum Homecomputer-Preis

Es ist noch nicht lange her, da kamen die ersten Homecomputer auf den Markt und leiteten eine bahnbrechende Revolution im Bereich "Neue Medien" ein.

Die Vorreiter der heutigen Homecomputer, der "PET 2001", APPLE und TRS80, waren von der Grundkonzeption eigentlich als PC entwickelt. Das technische Know-how dieser Geräte war mit einem 8k-Hauptspeicher und einem 8bit-Prozessor noch etwas dürftig. Nur der Preis von 2.000,- DM bis 3.000,- DM war für die damaligen Verhältnisse äußerst günstig und daher ist es nicht verwunderlich, daß diese "Kleinrechner" einen reißenden Absatz fanden. Die Freaks schätzten sich glücklich ein komplettes Homecomputer-System zu einem supergünstigen Preis zu bekommen.

Seit dieser Zeit sind jedoch einige Jahre ins Land gezogen. Jahre, in denen dieser Markt keine Ruhe fand. Jahre, in denen eine technische Revolution stattfand. Nimmt man die heutigen Rechner zum Vergleich, stellt man Unterschiede fest, die eigentlich nicht denkbar waren. So findet man heute bereits Rechner, die mit 128k-Hauptspeicher, hochauflösender Grafik etc. zu einem Preis von unter 800,- DM angeboten werden. Und wer heute einen Computer kaufen will, der braucht nicht einmal mehr in ein Fachgeschäft zu eilen. Nein! Der

kommt bei seinen täglichen Einkäufen im Supermarkt automatisch an den Computerständen vorbei und kann sich dort von diesem neuen Medium faszinieren lassen. Diesen Supermärkten und einem ständig steigenden Hersteller-Angebot an Rechner ist es zu verdanken, daß dieser Wirtschaftsbereich in den letzten Jahren fast 3stellige Zuwachsraten erwirtschaften konnte und das Preis-/Leistungsverhältnis ständig verbessert wurde. Da ist es dann auch eine ganz natürliche Sache, wenn die Grenze zwischen den Home- und Personalcomputern immer mehr verwischt.

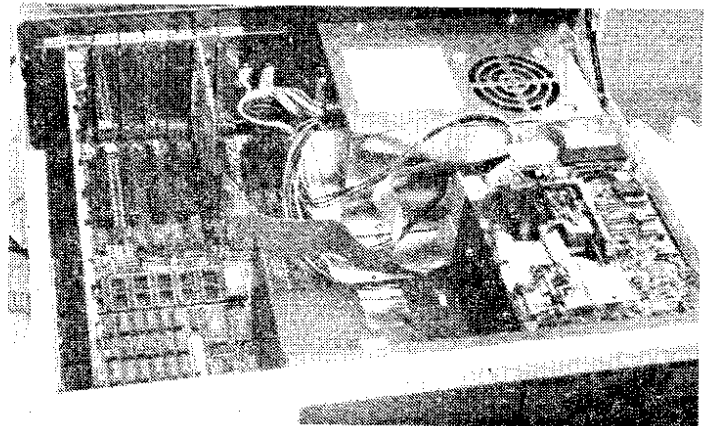
Denn auch im PC-Bereich hat sich einiges in Punkto Preis-/Leistungsverhältnis getan. So werden PC's angeboten, die in der Preisklasse eines größeren Homecomputers liegen. Da stellt sich für den Freak natürlich die Frage, welches System lege ich mir zu? Wir versuchen diese entscheidende Frage und einige andere mehr an dieser Stelle zu beantworten. Da ist z.B. noch das Problem, wie kompatibel sind die Rechner, oder welche technischen Möglichkeiten stecken in den PC's, oder welche Vorteile bringt der Einsatz dieses Rechners gegenüber einem Commodore 128 oder Schneider CPC 6128?

In den letzten Monaten wurde schon einiges zu diesem Thema veröffentlicht. Leider oft jedoch mit gegensätzlichen Behauptun-

gen. Deshalb sind wir selbst einmal diesem Thema auf den Grund gegangen und haben uns mit IBM-kompatiblen Rechner beschäftigt. Jeder in der Szene weiß, daß es auch in diesem Bereich eine Vielzahl von Anbietern gibt, und das jeder dieser Anbieter das Beste versucht, sein System zu verkaufen. Innerhalb dieses Riesengebotes fiel uns ein System

ganz besonders auf, der "ECD Professional" der Fa. ECD Computertechnik. Dieses System, inzwischen in größerer Stückzahl in unseren Redaktionen im Einsatz, haben wir ein wenig genauer unter die Lupe genommen und anhand dieses Systems wollen wir ein paar Daten und Antworten weitergeben.

Spitzenqualität Made in Taiwan



Ein Blick ins Innere des ECD zeigt eine saubere und solide Verarbeitung der einzelnen Bausteine

Genau wie der Großteil aller angebotenen PC's wird auch der "ECD Professional" im Land der untergehenden Sonne produziert. Das hat natürlich den Vorteil, das Spitzentechnologie zu einem günstigen Preis angeboten werden kann. Jedes kleine Kind weiß heute, daß die Japaner führend auf dem Gebiet der Mikroelektronik

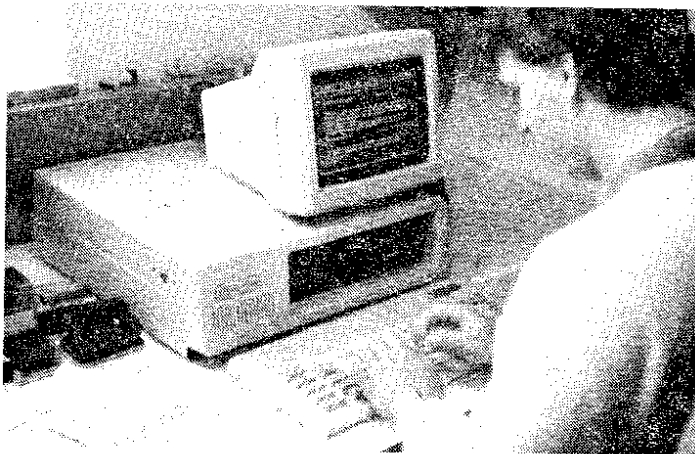
sind, und das in Billiglohnländern auch die Herstellungskosten der Chips, Platinen usw. um einiges niedriger liegen, als in unseren Breitengraden. Da ist es schon eine Notwendigkeit für die Konkurrenzfähigkeit eines Importeurs hier zu kaufen, zumal auch die großen Markenanbieter in diesen Ländern produzieren lassen.

Saubere Verarbeitung

Der ECD wird in einem stabilen Metall-Gehäuse, ähnlich dem Original IBM-Design, geliefert. Zwei seitlich angebrachte Druckknöpfe sorgen für eine schnelle und einfache Erreichung des Innenlebens dieses Systems. Ein Blick in diesen Bereich zeigt uns ein ausgesprochen aufgeräumtes und übersichtliches Innenleben. Ein Schaltnetzteil von 150 Watt sorgt dafür, daß bis zu sechs Erweiterungskarten angeschlossen und betrieben werden können. Ein

äußerst geräuscharmer Ventilator verhindert einen Hitzestau. Das Gehäuse wurde so konstruiert, daß man insgesamt vier Laufwerke oder aber 2 Laufwerke in Verbindung mit einer Festplatte installieren kann.

Auf der übersichtlich angeordneten Hauptplatine verrichtet ein 8088 Intel-Prozessor seinen Dienst. Neben ihm ist ein freier Steckplatz für den arithmetischen Coprozessor 8087 vorhanden.



Der ECD im Dauertest unserer Redaktion.

Die Programmierung
des TED-Chips

**Ein kompletter
Kurs,
um Sound
und Grafik des
C 16/116/plus 4**

**optimal zu
programmieren!**

Erscheint im

**»Compute mit«
Sonderheft 4**

**Termin:
15.12.1986**



Auch die wichtigsten DIP-Schalter zum Verändern der Konfiguration sind schnell gefunden und leicht erreichbar. Unser Testgerät war bereits mit einer 640k-Byte RAM-Erweiterung und 2 Diskettenlaufwerken ausgerüstet. Ein Anschluß an der Geräterückseite erlaubt zusätzlich den Betrieb zwei externer Laufwerke. Die eingebaute Videokarte liefert RGB- oder FBAS-Signale und eignet sich dadurch sowohl für einen Farbmonitor, als auch für einen günstigeren Monochrom-Monitor.

ASCII- oder DIN-Tastatur

Je nach Wahl gehören zum Lieferumfang des Rechners eine ASCII-Tastatur oder eine DIN-Tastatur. Mit 83 Tastsen, von denen NumLock, CapsLock und ScrollLock mit LED-Anzeige versehen sind, ist eine angemessene Ausstattung vorhanden. Sie ist damit in den Funktionen mit der Original-Tastatur von IBM zu vergleichen. Fehler in den Funktionen konnten auch nach längeren Belastungszeiten nicht festgestellt werden, so daß man von einer robusten und soliden Verarbeitung sprechen kann. Übrigens wird auch bei einigen Markenfabrikaten der höheren Preisklassen die gleiche Tastatur mitgeliefert.

Kompatibilität

Die wohl wichtigste Frage für den Anwender stellt sich in der Kompatibilität und der Anpassung der Hardware an die marktorientierte Software. Um das Ergebnis gleich vorwegzunehmen, die von uns getestete Software, und das waren immerhin über 100 Programme, liefen ausnahmslos ohne Beanstandungen. Es traten keinerlei Probleme bei der Handhabung der Programme mit dem "ECD Professional" auf. Lediglich die Arbeitsgeschwindigkeit der Floppy-Laufwerke war etwas langsamer als bei den Originalen. Offensichtlich mußte hier aus urheberrechtlichen Gründen das Diskettenbetriebssystem geändert werden. Insider werden aber mit Hilfe einer kleinen technischen Veränderung auch dieses Problem lösen können, wodurch dann der Rechner 100% kompatibel gemacht werden kann.

Günstiger Preis

Sicherlich werden Sie sich nun fragen, was muß ich für all diese technischen Details zahlen? Nun wir werden Ihnen auch diese Frage beantworten. Der Grundpreis für ein "ECD-Professional" mit einem Laufwerk bestückt mit 640 KB, der Color-Grafik-Karte, 150 W-Netzteil, DIN- oder ASCII-

Tastatur beträgt inkl. MwSt. ea. 1.775,- DM. Hinzu kommen nur noch die Versandkosten. Wir sind der Meinung, daß ist für soviel Spitzentechnik ein supergünstiger Preis. Jeder der sich mit dem Gedanken trägt in das Profilager umzusteigen und einen PC anschaffen will, der sollte dieses Angebot genau studieren und sich überzeugen lassen. Es muß ja nicht immer das teure Original sein, was man kaufen sollte.

Fazit:

An dieser Stelle ein Fazit zu ziehen ist schon relativ einfach. Ich war erstaunt und überrascht von soviel Qualität und Leistung zu einem solch günstigen Preis. Deshlab empfehle ich, in dem riesigen Hardware-Angebot genau zu selektieren, was man für seinen Bedarf benötigt und wie man seine Probleme am besten mit Hilfe des Computers realisieren kann.

Jede Anwendung verlangt von einem Rechner unterschiedliche Aufgaben. Wer nur Spiele mit dem Computer erleben will, der sollte auch in Zukunft nur auf den 8bit-Homecomputer mit 64k zurückgreifen. Für diese Ansprüche reicht ein solches System längst aus. Wer aber mit dem Computer arbeiten und komplexe Aufgaben lösen will, der sollte sich schon über die entsprechende

Hardware Gedanken machen. Inzwischen gibt es im Homecomputer-Bereich ebenfalls eine Vielzahl an guten Anwendungsprogrammen (Textverarbeitung, Tabellenkalkulation usw.). Allerdings muß bei solchen Programmen immer wieder auf die geringe Speicherkapazität der Hardware hingewiesen werden. Da ist es schon von Vorteil, wenn man auf den freien Speicherplatz eines PC's zurückgreifen kann. Ein weiterer Vorteil gegenüber dem Homecomputer ist natürlich auch das nimmer endenwollende Software-Angebot für MS-DOS Rechner. Hier wird man auch in Zukunft auf eine Riesenauswahl an Software stoßen und wie bereits erwähnt, gab es bei dem Test dieser Software auf dem "ECD Professional" keinerlei Probleme. Entscheiden Sie also beim Kauf Ihrer Hardware zunächst nach den Aufgabenstellungen und dann nach den Preis-/Leistungs-vorteilen, die sich bei den unterschiedlichen Systemen ergeben.

Sollten Sie sich für den Kauf eines PC's entschließen, verweisen wir auf unseren Testbericht und die

Fa.
ECD Computertechnik GmbH
Haindlfinger Str. 20
8051 Zolling/Palzing
Tel.: (08161) 66996

Hartmut Wendt/Frank Brall



HACKER



Für die VC-20-Grundversion:

Ballon Hopper

BALLON HOPPER ist ein Spiel für den VC 20 in der Grundversion, bei dem es darum geht, die Container am unteren Bildrand zu dem Transporter zu befördern. Unglücklicherweise ist aber gerade heute eine Flugtagveranstaltung und diverse Ballons stören den Kleintransporter gewaltig. Kollidieren Sie mit einem der Ballone, so müssen Sie sich schon einen neuen Transporter kaufen. Für 3 allerdings reicht Ihr Geld nur. Hat man acht Container ordnungsgemäß verladen, erhält man einen neuen Auftrag. Der ist natürlich noch schwerer.

Hinweise zum Programm
Das Programm besteht aus zwei Teilen, dem Loader und dem eigentlichen Spiel. Der Loader erzeugt die Grafikzeichen und das Maschinenprogramm, sowie noch eine kurze Spielanleitung.

Der "Hopper" wird während des Spiels mit den folgenden Tasten gesteuert:
Oben - F 1
Rechts - C
Links - Z
Unten - F 5

VC20 LISTING + CHECKSUMMEN (V V1.0)

```

10 POKES2,27:POKE56,27:CLR:A=7168:PRINT "CLEAR B
LACK DOHNS RIGHTS"BITTE<SPACE>WARTEN" <31>
20 READB:IFB>-1THENPOKEA,B:A=A+1:GOTO20 <252>
30 DATA7,30,62,124,124,252,248,248,224,120,124,6
2,62,63,31,31 <143>
40 DATA248,248,248,124,124,60,30,14,31,31,31,62,
62,60,120,112 <155>
50 DATA6,7,2,2,2,3,3,3,96,224,64,64,64,192,192,1
92 <4>
60 DATA0,3,7,31,63,97,109,97,0,192,224,248,252,1
34,182,134 <160>
70 DATA127,63,13,24,48,120,204,204,254,252,176,2
4,12,30,51,51 <203>
80 DATA127,97,111,97,125,97,127,0,254,134,182,18
2,182,134,254,0 <163>
90 DATA127,100,117,117,117,116,127,0,254,70,86,8
6,86,70,254,0 <208>
100 DATA0,126,126,126,126,126,126,0,-1 <172>
110 FDRA=7424T07431:POKEA,0:NEXT <64>
120 A=6912 <196>
130 READB:IFB>-1THENPOKEA,B:A=A+1:GOTO130 <25>
140 DATA169,0,133,0,166,0,189,60,3,72,189,61,3,1
68,104 <248>
150 DATA170,24,32,10,229,169,8,160,29,32,30,203,
166,0 <242>
160 DATA189,60,3,201,2,240,4,201,18,208,8,189,62
,3 <22>
170 DATA73,1,157,62,3,166,0,189,61,3,201,0,240,4
180 DATA201,20,208,8,189,63,3,73,1,157,63,3,166,
0 <215>
190 DATA189,62,3,201,2,240,13,201,1,208,6,222,60
,3 <7>
200 DATA24,144,3,254,60,3,189,63,3,201,2,240,13
<31>
    
```

```

210 DATA201,1,208,6,222,61,3,24,144,3,254,61,3,1
89,60,3 <202>
220 DATA72,189,61,3,168,104,170,24,32,10,229,169
,21 <180>
230 DATA160,29,32,30,203,165,0,197,1,208,1,96,16
5,0 <87>
240 DATA24,105,4,133,0,76,4,27,32,253,206,32,158
,215,138 <174>
250 DATA72,32,253,206,32,158,215,104,168,24,32,1
0,229,32,253,206,76,160,202,-1 <109>
260 A=7432 <74>
270 READB:IFB>-1THENPOKEA,B:A=A+1:GOTO270 <24>
280 DATA32,32,17,157,157,32,32,17,157,157,32,32,
0 <183>
290 DATA64,65,17,157,157,66,67,17,157,157,68,69,
0,-1 <192>
300 PRINT "CLEAR RED"***<SPACE>BALLON<SPACE2>HOP
PER<SPACE>***" <253>
310 PRINT "DOHNS GREEN"REDIENUNGSANWEISUNG:<SPAC
E2 DOHNS BLUE SPACE>SIE<SPACE>MUESSEN<SPACE>DIE<S
PACE>KON-<SPACE DOHNS SPACE2>TAINER<SPACE>AM<SPAC
E>UNTEREN<SPACE>" <232>
320 PRINT "DOHNS BILDRAND<SPACE>DURCH<SPACE>ANDOK
-<SPACE DOHNS KEN<SPACE>ZU<SPACE>IHREM<SPACE>TRAN
SPOR-<DOHNS TER<SPACE>(OBERER<SPACE>BILDRAND)" <78>
330 PRINT "DOHNS SPACE6"BEFOERDERN.":PRINT "DOHNS
SPACE8 RVSON CYAN"TASTE<RVSOFF"> <80>
340 BETA$:IFA$=""THEN340 <240>
350 PRINT "CLEAR RED SPACE3"ABER<SPACE>VORSICHT<
SPACE>!" :PRINT "DOHNS PURPLE"HEUTE<SPACE>IST<SPAC
E>FLUGTAG<SPACE>UND<SPACE DOHNS SPACE>EINE<SPACE>
KOLLISION<SPACE>MIT" <176>
360 PRINT "DOHNS SPACE" EINEM<SPACE>BALLON<SPACE>K
OSTET<SPACE2 DOHNS SPACE6>EIN<SPACE>LEBEN." <86>
370 PRINT "DOHNS BLACK SPACE" SIND<SPACE>ACHT<SPAC
E>KONTAINER<SPACE2 DOHNS SPACE2>ABGELIEFERT<SPACE
>WORDEN,<SPACE DOHNS SPACE>BEGINNT<SPACE>DIE<SPAC
E>NAECHSTE" <77>
380 PRINT "DOHNS SPACE2" (SCHWERERE) <SPACE>RUNDE." <110>
390 PRINT "DOHNS SPACE8 RVSON CYAN"TASTE<RVSOFF">
" <175>
400 BETA$:IFA$=""THEN400 <4>
410 PRINT "CLEAR BLUE"GESTEUERT<SPACE>WIRD<SPACE
>MIT.":PRINT "DOHNS BLACK"TAB(10)"F1" <23>
420 PRINTTAB(10)"(S-)" :PRINTTAB(8)"Z(S* S* S)C"
:PRINTTAB(10)"(S-)" :PRINTTAB(10)"F5" <159>
425 PRINT "DOHNS GREEN" IN<SPACE>DEN<SPACE>BONUS<S
PACE>RUNDEN<SPACE3>GENIEBT<SPACE>EIN<SPACE>TASTE
N-<SPACE3>DRUCK<SPACE>UM<SPACE>AUF<SPACE>DIE<SPA
CE>KON-" <214>
427 PRINT "TAINER<SPACE>NIEDERZURASEN. (BLACK)" <121>
430 PRINT "DOHNS NUN<SPACE>WIRD<SPACE>DAS<SPACE>S
PIELPRO-<DOHNS SPACE2>GRAMM<SPACE>GELADEN<SPACE>!"
<146>
440 POKE631,131:POKE198,1:END <254>
500 REM ***** <206>
510 REM BALLON HOPPER <122>
520 REM (C) 1986 BY <165>
530 REM THOMAS KOLBE <212>
540 REM WILDERMANNSTR. 62 <181>
550 REM 4350 RECKLINGHAUSEN <154>
560 REM ***** <11>
ENDE DES LISTINGS
    
```

Teil 2

VC20 LISTING + CHECKSUMMEN (V V1.0)

```

10 AT=7061:TT=1:LI=3 <82>
20 PRINT "CLEAR":POKE36869,255:A$="" <54>
30 PRINT "BLACK"NNN<SPACE16>NNN" <161>
40 SYSAT,0,21,"N(CRED)JK<SPACE2>JK<SPACE2>JK<SPAC
E2>JK<SPACE2>JK<SPACE2 BLACK>NN<GREEN>LM<BLACK>N
N<GREEN>LM<BLACK>NN<GREEN>LM<BLACK>NN<GREEN>LM<B
LACK>NN<GREEN>LM<BLACK>NN<HOME>" <38>
50 POKE185,14:POKE38905,0 <5>
60 X=0:Y=2:A=RND(0) <91>
70 FDRA=OTOTT:POKEB28+A*4,RND(TI)*15+3:POKEB29+A
*4,RND(1)*19+1 <26>
80 POKEB30+A*4,RND(1)*3:POKEB31+A*4,RND(1)*3 <28>
90 IFPEEK(B30+A*4)=2ANDPEEK(B31+A*4)=2THENB0 <52>
    
```

programme

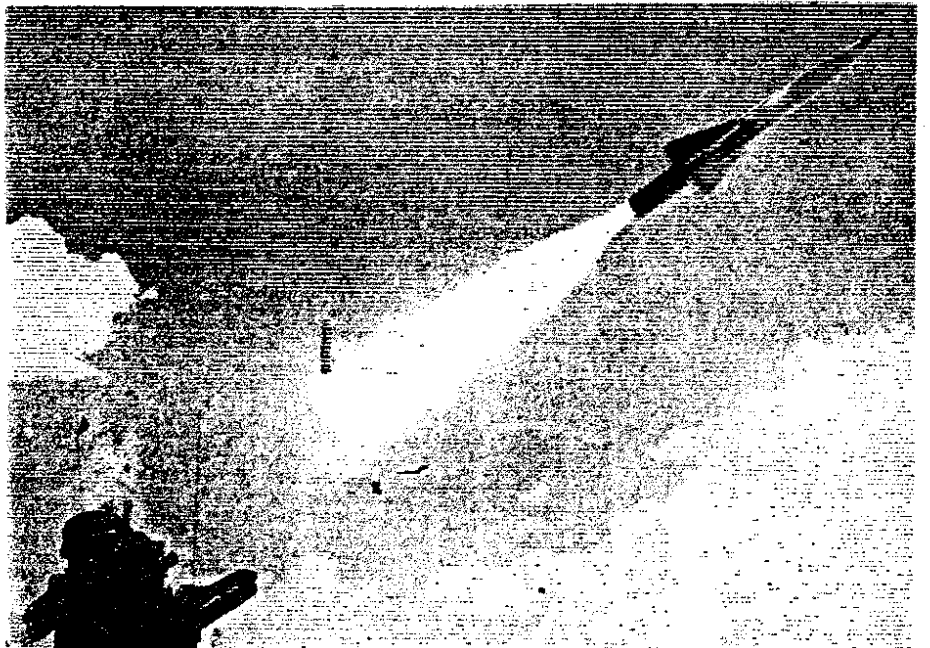
```

100 NEXTA:POKE1,TT*4
110 SYSAT,X,Y,"FG(DOWN LEFT2)HI":GOSUB700
120 AX=PEEK(197):IFAX=33ANDX>0THENGOSUB600:X=X-1
130 IFAX=34ANDX<20AND(NOT(X=19ANDY=19ANDS>0))TH
ENGOSUB600:X=X+1
140 IFAX=39ANDY>2THENGOSUB600:Y=Y-1
150 IFAX=55ANDY<19-SXTHENGOSUB600:Y=Y+1
160 SYSAT,X,Y,"(BLACK)FG(DOWN LEFT2)HI(BLUE)":IF
SX>0THENGOSUB600:Y=Y-1
170 SYS6912:B=7680+X+22*Y
180 IFPEEK(B)<60RPEEK(B+1)<60RPEEK(B+22)<60RPEEK
(B+23)<6THEN1000
190 IFPEEK(B+44)=10THENSX=1
200 IFPEEK(B+66)=12ANDPEEK(B+44)=32THENSX=2:SYSAT
T,X,Y+3,"(SPACE2 HOME)"
210 IFY>20RSX=0THEN120
220 SYSAT,X,Y+2,"(SPACE2 HOME)":ONSXGOSUB520,570
: SX=0:GOSUB800
230 SYSAT,3,1,A#:PRINT"(HOME RVSON BLUE)SCORE:"S
C:PRINT"(HOME)"TAB(13)"(GREEN RVSON)LI:"LI"(BLAC
K)"
240 IFLEN(A#)=24THEN400
250 GOTO120
400 POKE36878,15:FORA=1TO10:FORS=128TO250STEP10:
POKE36876,S:NEXTS,A
410 POKE36876,0
420 FORA=21TO0STEP-3:SYSAT,3,1,LEFT$(A#,A)"(SPAC
E2)":POKE36876,240
425 SC=SC+TT*10:PRINT"(HOME BLUE RVSON)SCORE:"SC
430 FORS=1TO100:NEXTS:POKE36876,0:FORS=1TO100:NE
XTS,A
440 TT=TT+1:LI=LI+1:IFTT/3=INT(TT/3)THENGOSUB200
0
450 GOTO20
500 SYSAT,X,Y+2,"(RED)JK(BLUE HOME)":RETURN
520 A#=A#+"(RED)JK":SC=SC+50:RETURN
550 SYSAT,X,Y+2,"(GREEN)LM(BLUE HOME)":RETURN
570 A#=A#+"(GREEN)LM":SC=SC+100:RETURN
590 END
600 SYSAT,X,Y,"(SPACE2 DOWN LEFT2 SPACE2)":IFSX>
0THENSYSAT,X,Y+2,"(SPACE2 HOME)"
610 RETURN
700 POKE36878,15:FORA=1TO2:RESTORE:FORB=1TO8:REA
DH,0
710 POKE36876,H:FORS=1TO10:NEXT:POKE36876,0:FORS=
1TO10:NEXTB,A
720 DATA225,50,195,100,195,50,209,600,217,50,209
,50,217,50,225,600
730 FORB=15TO0STEP-.4:POKE36878,B:POKE36876,244:
A=SIN(1234)
<111> 740 POKE36876,0:A=SIN(1234):NEXTB:POKE36878,15
<16> 750 RETURN
<252> 800 POKE36876,175:POKE36875,207:POKE36874,225:FO
RA=15TO0STEP-.2
<182> 810 POKE36878,A:NEXTA:POKE36876,0:POKE36875,0:PO
KE36874,0:RETURN
<227> 1000 POKE36878,15:FORA=1TO2:FORS=128TO250:POKE36
876,S:POKE36876,0:NEXTS,A
<147> 1010 GOSUB600:X=0:Y=2
<164> 1020 LI=LI-1:IFLI>0THEN230
<223> 1030 FORA=250TO128STEP-.5:POKE36876,A:NEXT:POKE3
6876,0
<228> 1040 SYSAT,3,10,"(RVSON CYAN CA S#14 CS)":SYSAT,
3,11,"(RVSON CYAN S- SPACE2 BLACK)GAME(SPACE2)OV
ER(CYAN SPACE2 S-)"
<86> 1050 SYSAT,3,12,"(RVSON CZ S#14 CX)"
<219> 1060 POKE198,0:WAIT198,1:RUN
<248> 2000 POKE36878,15:FORB=1TO10:PRINT"(CLEAR DOWN4
RIGHT4 RED RVSON)BONUS(SPACE)RUNDE(SPACE)!"
<48> 2005 POKE36876,128+B*10
<53> 2010 FORA=1TO300:NEXT:PRINT"(CLEAR)":FORA=1TO300
: NEXTA,B
<219> 2020 POKE36876,0:AT=7061:POKE36869,255:POKE36878
,15
<59> 2030 PRINT"(CLEAR)":FORA=0TO20STEP2:SYSAT,A,18,"
(BLUE)EA(DOWN LEFT2)BC(DOWN LEFT2)DE(HOME)":NEXT
A
<188> 2040 A=INT(RND(1)*11)*2:SYSAT,A,18,"(SPACE2 DOWN
LEFT2 RED)JK(DOWN LEFT2 GREEN)LM(HOME)"
<37> 2050 A=0:POKE198,0
<249> 2060 SYSAT,A,0,"(BLACK)FG(DOWN LEFT2)HI(HOME)":P
OKE36876,240:B=SIN(1234):POKE36876,0
<172> 2065 GETA#:IFA#<>"THEN2090
<88> 2070 SYSAT,A,0,"(SPACE2 DOWN LEFT2 SPACE2 HOME)"
: A=A+2:IFA>20THENA=0
<184> 2080 GOTO2060
<64> 2090 FORB=0TO18:SYSAT,A,B,"FG(DOWN LEFT2)HI(HOME
)":POKE36876,150:D=SIN(123):POKE36876,0
<61> 2100 SYSAT,A,B,"(SPACE2 DOWN LEFT2 SPACE2 HOME)"
: NEXTB
<108> 2110 IFPEEK(7680+A+22*(B+1))=12THEN2500
<190> 2115 SYSAT,5,10,"(BLUE RVSON)KEIN(SPACE)BONUS(SP
ACE)!"
<64> 2120 FORA=240TO128STEP-.5:POKE36876,A:NEXT:POKE3
6876,0:RETURN
<208> 2500 SC=SC+50*TT:SYSAT,5,10,"(BLUE RVSON)BONUS:"
50*TT
<46> 2510 FORI=1TO10:FORA=140TO200:POKE36876,A:NEXTA,
I:POKE36876,0:RETURN
<166> ENDE DES LISTINGS
<17>

```

FIGHT NIGHT

**Beschützen Sie
die Erde vor
fremden Raumschiffen**



Für VC-20 (Grundversion)

In der ersten Stufe müssen Sie den Feind von den Energievorräten abhalten. Wenn Sie Ihre Energievorräte an die Ausserirdischen verloren haben, beginnt die 2. Stufe, in der Sie die Angreifer mindestens 30 Sekunden von der Erde fernhalten müssen. Haben Sie dies geschafft, geraten Sie wieder in den ersten Level.

Insgesamt dreimal können Sie Ihr Können unter Beweis stellen, falls Sie nicht schon vorher ausgeschieden sind.

Hinweise

Das Spiel besteht aus Vorprogramm und Hauptprogramm. Wichtig: Am Ende des Ladevorgangs muß die Datasette ausgeschaltet werden.

VC20 LISTING + CHECKSUMMEN (V V1.0)

```

0 REM HEINZ KROOS,NEUE REIHE 26,4795 B0KE <6>
1 POKE1,0:POKE2,29:PRINT "CLEAR BLACK";:POKE368 <172>
79,8:FORI=7168TO7327:READA:POKEI,A:NEXT
16 FORA=7328TO7423:POKEA,0:NEXT:FORI=7424TO7657: <147>
READA:POKEI,A:NEXT
30 DATA255,127,63,63,63,127,255,255,1,7,143,223, <105>
255,255,255,255
40 DATA128,192,248,252,254,255,255,255 <235>
60 DATA7,15,31,31,31,63,255,255,255,255,189,153, <86>
153,189,255,255
80 DATA0,0,1,3,31,63,255,255,1,3,231,255,255,255 <174>
,255,255
90 DATA128,192,192,192,248,254,255,255 <85>
110 DATA255,255,255,255,255,255,255,255,1,15,31, <156>
63,63,127,255,255
120 DATA0,0,1,3,7,31,255,255 <90>
130 DATA128,128,128,192,192,224,240,255 <96>
140 DATA135,207,255,255,255,255,255,255 <96>
160 DATA0,0,0,1,1,3,7,31,128,192,192,224,240,240 <121>
,249,255
170 DATA0,0,0,0,126,255,255,255 <46>
180 DATA16,56,124,56,16,56,56,124 <78>
190 DATA24,24,24,24,126,255,255,255 <42>
200 DATA0,3,103,254,254,254,103,3 <66>
210 DATA0,192,230,127,127,127,230,192 <1>
340 DATA169,150,133,251,169,29,133,252 <244>
350 DATA169,0,133,98,133,99,160,128 <44>
360 DATA177,251,201,17,208,9,230,99 <253>
370 DATA208,2,230,98,24,144,101,201 <15>
380 DATA21,144,10,177,251,56,233,1 <99>
390 DATA145,251,24,144,87,201,18,208 <14>
400 DATA38,136,177,251,201,20,144,22 <219>
410 DATA160,149,177,251,201,21,240,14 <148>
420 DATA160,128,169,21,145,251,136,169 <207>
430 DATA18,145,251,24,144,54,160,128 <27>
440 DATA169,19,145,251,24,144,45,201 <150>
450 DATA19,208,41,200,177,251,201,20 <205>
460 DATA144,28,160,151,177,251,201,21 <253>
470 DATA240,20,160,128,169,21,145,251 <237>
480 DATA200,169,19,145,251,230,251,208 <225>
490 DATA2,230,252,24,144,6,169,18 <152>
500 DATA160,128,145,251,230,251,208,2 <81>
510 DATA230,252,165,251,201,127,240,3 <15>
520 DATA76,14,29,165,252,201,31,240 <231>
530 DATA3,76,14,29,162,144,56,76 <32>
540 DATA73,220,162,0,189,0,30,157 <239>
550 DATA0,26,189,0,31,157,0,27 <134>
560 DATA232,208,241,96,162,0,189,0 <137>
570 DATA26,157,0,30,168,185,212,29 <2>
580 DATA157,0,150,189,0,27,157,0 <30>
590 DATA31,168,185,212,29,157,0,151 <239>
600 DATA232,208,227,96,5,5,5,5 <24>
610 DATA7,5,5,5,5,5,5,5,5,1,2,1,1,1,1,1 <15>

```

```

700 POKE36869,255:FORA=1TO32:PRINT"TTTTTTTTTTTT"; <219>
:NEXT
710 PRINT"TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT <122>
TTTTT@TTTTTTTTTTTTTTTTJHKTTTTTH";
720 PRINT"ABCHNEFGTTTTTIIHHGTOHHHHHHHHHHGTTTIIHH <83>
HHLHHHHHHHHHHHGGIHHHHHHHHHH";
730 PRINT"HHHHHHHHHHHHHHHHHHHH";:POKEB185,8:POK <56>
E38905,6
740 SYS7586:PRINT"(CLEAR)":SYS7604 <78>
800 PRINT"(HOME WHITE DOWN2 RIGHT)SSSSSSSSSSSSSS <229>
SSSSS(RIGHT3)S(DOWN LEFT)S(RIGHT16)S(RIGHT4)S(RI
GHT11)R(RIGHT2)SSSS"
810 PRINT"(RIGHT)S(RIGHT3)R(RIGHT7)R(RIGHT4)S(RI <236>
GHT4)SSS(RIGHT2)R(RIGHT4)S(RIGHT4)S(RIGHT3)R(RI
GHT2)SSSS(RIGHT)RRRR(RIGHT)S"
820 PRINT"(RIGHT)S(RIGHT3)R(RIGHT2)S(RIGHT)SS(RI <68>
GHT)RR(RIGHT)R(RIGHT)S(RIGHT4)S(RIGHT3)R(RIGHT2)
SSSS(RIGHT)R(RIGHT2)R(RIGHT)S(RIGHT4)S(RIGHT3)R(R
IGHT5)S(RIGHT)R(RIGHT2)R(RIGHT)SS"
830 PRINT"(RIGHT)S(RIGHT3)R(RIGHT5)S(RIGHT7)SS(R <15>
IGHT11)SS"
835 GOSUB6000 <127>
840 FORI=1TO40:FORA=7680TO8054STEP22:POKEA,20:NE <178>
XT:Q=USR(0):NEXT
850 A$="(RVSON WHITE)ENERGIEVORAE=(RVSOFF YELL <7>
ON)DTTTTT(UP)D"
860 B$="(WHITE)TTTTTTTTTTTTTTTTTTTT(UP)T" <53>
865 PRINT"(DOWN3)"A$:GOSUB6000 <238>
870 FORA=1TO18:PRINTB$:;PRINT"(UP)"A$;:FORI=1TO1 <166>
00:NEXT:NEXT
880 PRINT"(RVSON DOWN2)BASIS=":FORA=8130TO77568T <9>
EP-22:POKEA,17:POKEA+22,20:FORB=1TO100:NEXT:NEXT
889 GOSUB6000:PRINT"(CLEAR)":POKE36869,240:POKE6 <20>
31,131:POKE198,1:END
6000 FORA=1TO3000:NEXT:RETURN <18>
ENDE DES LISTINGS

```

Teil 2

VC20 LISTING + CHECKSUMMEN (V V1.0)

```

0 PRINT"(CLEAR WHITE)":;G=.17:H=7702:TI$="000000 <171>
":Y=8052:GOSUB1:GOTO22
1 POKE36869,255 <151>
2 POKE36879,8:POKE36878,15 <100>
20 SYS7604:PRINT"(HOME)":;FOREO=1TO32:PRINT"TTTT <55>
TTTTTT";:NEXT:RETURN
22 IFY>8052THENPOKEY-15,20:POKEY+30705,1:POKEY+7 <209>
,20:POKEY+30727,1:POKEY+29,5
30 IFEA=1THENPRINT"(HOME WHITE DOWN8 RIGHT)R(RIG <153>
HT)R(RIGHT)R(RIGHT)R(RIGHT)R(RIGHT)R(RIGHT)R(RIG
HT)R(RIGHT)R(RIGHT)R(RIGHT4)R(RIGHT)R(RIGHT)R(RI
GHT)R(RIGHT)R(RIGHT)R(RIGHT)R(RIGHT)R"
90 B=8152 <161>
91 POKE38421,0:POKE7701,14 <187>
99 POKE36874,135 <198>
100 GOSUB1200 <153>
101 IFRND(1)<GTHENPOKEH,19 <254>
102 IF TI$>"000014"THENGOTO500 <243>
110 GOSUB1300 <166>
140 IFPEEK(Y)=19THEN600 <218>
150 GOSUB1400 <209>
280 IFF>1THENQ=USR(0):IFPEEK(Y)=19THEN600 <151>
300 GOTO100 <104>
400 POKE36877,210 <202>
410 FORA=B-22TO8009STEP-22:POKEA,16:POKEA,20:NEX <138>
T:Q=USR(0):IFPEEK(Y)=19THEN600
411 POKE36877,0:FORR=AT07823STEP-22:POKER,16:POK <69>
ER,20:NEXT:Q=USR(0):IFPEEK(Y)=19THEN600
412 IFF>4THENPOKE7746,19 <162>
420 FORA=RT07679STEP-22:POKEA,16:POKEA,20:NEXT:R <222>
ETURN
500 F=F+.5:H=H+44:POKE36874,135+F*5:IFF>8THENPOK <116>
E36874,201
502 IFH>7966THENH=7966:G=G+.1:F=F-.5 <252>
503 G=G+.05 <153>
510 SI=SI+VAL(TI$):TI$="000000" <97>
599 GOTO100 <148>
600 POKE36874,127:POKE36877,127 <106>
610 PRINT"(HOME DOWN)":;FORA=1TO34:PRINT"TTTTTT <37>
TTTT";:NEXT

```


programme

```

620 FORI=YTO7702STEP-21:POKEI+1,4:POKEI,19:POKEI
-1,21:POKEI+20,20:POKEI+22,20
623 POKEI+21,20:POKE36874,(Y-1)/4+127
630 FORZ=1TO100:NEXT:NEXT
635 EW=EW+1:IFEW=5THEN700
640 POKE36874,0:IFEW>7THEN900
650 IFY>8052THEN700
699 Y=Y+16:PRINT" (CLEAR WHITE)";:GOSUB1:GOTO22
700 TI$="000000":PRINT" (CLEAR WHITE)";:IFEW<5THE
NGOSUB950:GOTO702
701 GOSUB1100
702 POKE36869,255:GOSUB1:POKE8053,20:POKE8075,20
:POKE8097,5:POKE8069,20:POKE8091,6
703 Y=8078:POKE36874,195:B=B152
704 IFEW=5THEN707
705 PRINT" (HOME YELLOW DOWN3)S (DOWN)S (DOWN)S (DOW
N)S (DOWN)S (DOWN)S (UP)S (UP)S (UP)S (UP)S (UP)S (DOWN)
S (DOWN)S (DOWN)S (DOWN)S (DOWN)S (UP)S (UP)S (UP)S (UP)
S (UP)S":TI$="000000"
706 GOTO710
707 PRINT" (HOME DOWN7 LEFT BLACK)N (WHITE)STSTSTS
TSTSTRTRTRTRTRT (BLACK)N"
710 GOSUB1200
720 IFTI$>"000030"GOTO800
730 GOSUB1300
760 IFPEEK(Y)=19THEN900
765 GOSUB1400
788 Q=USR(0):IFPEEK(Y)=19THEN900
790 GOTO710
800 Y=8052:POKE36874,0:POKE36877,0
810 IFEW=5THENFDOEQ=7834TD7855:IFPEEK(E0)<>20THE
N900

```

```

<115>
<90>
<101>
<86>
<107>
<95>
<27>
<10>
<241>
<216>
<157>
<135>
<23>
<14>
<207>
<253>
<50>
<20>
<102>
<58>
<190>
<99>
<218>
<65>

```

```

B11 IFEW=5THENNEXT
820 PRINT" (CLEAR WHITE)";:EW=EW+1:GOSUB1:GOTO22
900 POKE36874,0:POKE36877,0
910 SI=SI+VAL(TI$):POKE36869,240
920 PRINT" (CLEAR DOWN7 RIGHT)DU (SPACE)HAST"SI"SE
KUNDEN (SPACE DOWN RIGHT3) DURCHBEHALTEN"
930 GETA$:IFA$=""THEN930
940 RUN
950 POKE36869,240:PRINT" (CLEAR WHITE DOWN6 RIGHT
2)HALTE (SPACE)NUN (SPACE)30 (SPACE)SEC.":PRINT" (RI
GH2 DOWN2)AUS"
955 FOREY=1TO2500:NEXT
960 PRINT" (CLEAR WHITE)";:RETURN
1100 POKE36869,240:POKE36874,0
1101 PRINT" (CLEAR DOWN8 RIGHT)SCHIESSE (RIGHT)IN (
RIGHT)30 (RIGHT)SEC. (RIGHT3 DOWN)ALLES (RIGHT)AB":
FDOEQ=1TO1800:NEXT:RETURN
1200 POKE37154,127:J=(PEEK(37152)AND128)+(PEEK(3
7151)AND28):POKE37154,255:RETURN
1300 C=B:B=8152:IFJ=140THENB=8079
1313 IFJ=28THENB=8117
1314 IFC<>BTHENPOKEC,15:POKEB,17
1320 Q=USR(0):RETURN
1400 M=N:IFPEEK(37137)<102THENN=N+1:IFN<2THENGOS
UB400:GOTO1420
1410 IFM=NTHENN=0
1420 PRINT" (RVSON HOME WHITE)"VAL(TI$)+SI
1430 M=N:IFPEEK(37137)<102THENN=N+1:IFN<2THENGOS
UB400:GOTO1450
1440 IFM=NTHENN=0
1450 RETURN
ENDE DES LISTINGS
<145>
<193>
<177>
<204>
<79>
<123>
<57>
<128>
<149>
<127>
<156>
<205>
<163>
<33>
<20>
<137>
<147>
<66>
<111>
<49>
<210>
<141>
<61>

```

Spannende Action für Ihren C 16/116

King of Kings

Auf dem Planeten Galactica regiert der Tyrannenherrscher Thong. Um die Bevölkerung zu retten, sind Sie in den Palast des Herrschers eingedrungen. Dieser hat 9 Räume, die von Soldaten bewacht werden. Diese und noch einige andere Hindernisse müssen Sie überwinden. Haben Sie alle geschafft, so ist das Volk von seinem Tyrannen befreit und ernennt Sie zum König der Könige.

Spieleanleitung

Am Anfang des Bildes befinden Sie sich am linken, unteren Bildschirmviertel. Der Raum, in dem Sie sich befinden, gliedert sich in 3 Etagen. Diese müssen Sie durchqueren und den Schlüssel rechts oben berühren. Dadurch errei-

chen Sie den nächsten Level. Welche Gefahren im Lauf des Spieles auftreten werden, sollen Sie selber austüfeln, aber bedenken Sie, daß der Bonuszähler unbarmherzig die Zeit herunter zählt.

Steuerung

Shift - Springen nach oben

Komma - nach links

Punkt - nach rechts

SHIFT in Verbindung mit PUNKT oder KOMMA: nach rechts oder links springen

A = Aufzug (bringt Sie eine Etage höher).

Punkte: 70 Punkte für die Überwindung einer Etage.

Für einen kompletten Level, wird Ihnen der Bonus angerechnet, als KING OF KINGS erhalten Sie nochmal 5000 Punkte, eine kleine Überraschung, und mit einem weiteren Bonusleben beginnen Sie wieder mit einem etwas schnelleren Level 1.

Tips zum Abtippen

Das Spiel belegt den Speicherplatz von \$1000 - \$3FFF0. Es muß im TEDMON-Monitor eingegeben werden. Geben Sie Monitor und RETURN ein - nun können Sie mit dem Befehl M beginnen, das Listing einzugeben. Wenn man erst einen Teil abgetippt hat, kann man ihn mit S"KOK",1,

1000,Endadresse + 1 abspeichern. Hat man das ganze Spiel eingetippt, geht man zurück in das Basic und gibt folgende Pokes ein: POKE 45,240: POKE46, 63.

Nun kann man das ganze Spiel mit SAVE"KOK" absaven. Der Programmname darf nicht länger als 3 Zeichen sein, da sonst falsche Daten in dem Spiel auftauchen.

Wenn Sie nun das Spiel laden und LIST eingeben, sehen Sie eine kleine Basiczeile: 1986 SYS 8128 BY HARTMUT OTT.

Das Spiel kann nun mit RUN gestartet werden.

```

>1000 00 00 00 c2 07 9e 20 30 >11f8 ea e8 e0 05 d0 eb f0 0b >13f0 88 88 bd 50 10 aa 20 6b
>1008 31 32 33 20 42 59 20 48 >1200 a2 05 bd 7e 10 9d 83 10 >13f8 3f 20 9b 3f a0 00 a6 d7
>1010 41 52 54 4d 55 54 20 4f >1208 ca d0 f7 a2 28 a9 01 9d >1400 bd 52 10 c9 03 90 15 c9
>1018 54 54 00 00 00 00 00 00 >1210 ff 07 ca d0 fa a9 02 8d >1408 03 f0 16 c9 08 f0 12 c9
>1020 00 00 00 00 00 00 00 00 >1218 00 0c a9 0f 8d 01 0c 8d >1410 04 f0 13 c9 07 f0 0f a9
>1028 00 00 00 00 00 00 00 00 >1220 0e 0c 8d 1e 0c a9 0e 8d >1418 00 4c 28 14 a9 78 4c 28
>1030 00 00 00 00 00 00 00 00 >1228 02 0c a9 15 8d 03 0c a9 >1420 14 a9 50 4c 28 14 a9 28
>1038 26 00 26 00 26 00 26 00 >1230 13 8d 04 0c 8d 0c 0c 8d >1428 85 d3 c4 d3 f0 09 98 18
>1040 00 00 00 00 00 00 00 00 >1238 1c 0c a9 03 8d 0d 0c 8d >1430 69 20 a8 c4 d3 d0 f7 c0
>1048 00 00 00 00 00 00 00 00 >1240 1d 0c a9 12 8d 0f 0c 8d >1438 78 f0 13 a9 0e 91 d8 c8
>1050 03 18 06 10 18 08 17 18 >1248 1f 0c a9 05 8d 10 0c 8d >1440 91 d8 c8 91 d8 98 18 69
>1058 03 1d 18 00 00 00 00 00 >1250 20 0c a9 08 8d 1a 0c a9 >1448 26 a8 c0 78 d0 ed 20 98
>1060 00 00 00 00 00 00 00 00 >1258 09 8d 1b 0c a2 05 bd 7e >1450 3f a0 00 c4 d3 f0 17 a2
>1068 00 00 00 00 00 00 00 00 >1260 10 18 69 30 9d 11 0c bd >1458 03 20 a0 1b c8 ca d0 f9
>1070 00 00 00 00 00 00 00 00 >1268 03 10 18 69 30 9d 21 0c >1460 ea 98 18 69 25 a8 c4 d3
>1078 00 00 02 05 02 00 18 00 >1270 ca d0 eb a2 03 bd 79 10 >1468 d0 ed c0 78 f0 21 a2 3c
>1080 00 00 00 00 00 00 00 00 >1278 18 69 30 9d 05 0c ca d0 >1470 8a 91 d8 c8 e8 e8 e8 8a
>1088 00 00 00 05 00 00 00 02 >1280 f4 60 a2 78 a9 20 9d 00 >1478 91 d8 c8 e8 e8 e8 8a 91
>1090 a2 0f a9 08 9d 79 10 ca >1288 0c ca d0 fa 60 a2 5f a9 >1480 d9 8a 38 e9 05 aa 98 18
>1098 d0 fa a9 18 8d 07 ff a9 >1290 00 9d 1f 10 ca d0 fa 8d >1488 69 26 a8 c0 78 d0 e1 a6
>10a0 1b 8d 06 ff a9 c0 8d 12 >1298 09 10 8d 8a 10 85 d6 85 >1490 d7 fe 52 10 bd 52 10 c9
>10a8 ff a9 30 8d 13 ff a9 32 >12a0 da a2 03 a9 04 9d 79 10 >1498 09 d0 05 a9 00 9d 52 10
>10b0 8d 17 ff a9 71 8d 18 ff >12a8 ca d0 fa a2 10 bd b7 32 >14a0 a5 d7 18 69 03 85 d7 c9
>10b8 60 a2 05 a9 00 9d 7e 10 >12b0 9d c7 32 ca d0 f7 8e 8d >14a8 1e f0 03 4c b3 13 60 a9
>10c0 ca d0 fa 8d 8c 10 a9 05 >12b8 10 8e 7e 10 a9 27 85 db >14b0 03 85 d7 a4 d7 c8 b1 da
>10c8 8d 8b 10 a9 c1 8d 15 ff >12c0 ad 8c 10 f0 0f a8 a5 da >14b8 d0 03 4c d8 15 a5 d7 38
>10d0 8d 19 ff a9 00 8d 8c 10 >12c8 18 69 53 90 02 e6 db 85 >14c0 e9 03 0a 85 d3 a6 d3 bd
>10d8 ea ea ea 60 a9 00 85 d7 >12d0 da 88 d0 f2 20 48 3f a2 >14c8 21 10 c9 00 d0 19 ad 00
>10e0 a5 d7 0a aa b5 e5 d0 03 >12d8 28 a0 00 b1 da 9d 77 00 >14d0 ff c9 05 b0 0f a4 d7 a6
>10e8 4c b7 11 a8 b5 e4 aa 20 >12e0 9d 8f 09 9d a7 0a 9d bf >14d8 d3 b1 da 9d 20 10 c8 b1
>10f0 6b 3f a2 03 a0 00 20 b0 >12e8 0b a9 73 9d 77 0c 9d 8f >14e0 bd 9d 21 10 4c 82 15 a8
>10f8 1b ea ea ea ea a5 d7 >12f0 0d 9d a7 0e 9d bf 0f ca >14e8 bd 20 10 aa 20 6b 3f a2
>1100 0a aa b5 e4 c9 24 f0 04 >12f8 d0 df a0 03 20 6b 3f 20 >14f0 4d 20 05 18 ea ea ea ea
>1108 c9 01 d0 09 a5 d7 aa b5 >1300 9b 3f a0 00 b1 da 91 d8 >14f8 ea ea ea ea ea ea ea ea
>1110 d0 49 01 95 d0 a5 d7 0a >1308 a0 27 91 d8 20 93 3f a0 >1500 ea ea ea ea ea ea ea ea
>1118 aa b5 e5 05 d3 a5 d7 0a >1310 00 a9 73 91 d8 a0 27 91 >1508 ea a6 d3 e0 0b 90 09 de
>1120 0a 0a 38 e5 d7 18 69 04 >1318 d8 ea a5 d8 18 69 28 90 >1510 20 10 de 20 10 4c 1e 15
>1128 c5 d3 f0 07 18 69 05 c5 >1320 02 e6 d9 85 d8 c9 e8 d0 >1518 fe 20 10 fe 20 10 bd 21
>1130 d3 d0 09 a5 d7 aa b5 d0 >1328 d6 a0 01 b1 da 85 d4 c0 >1520 10 a8 bd 20 10 aa 20 6b
>1138 49 02 95 d0 a5 d7 aa b5 >1330 b1 da 85 d5 a0 4b a2 03 >1528 3f a6 d3 bd 20 10 f0 1c
>1140 d0 49 04 95 d0 a5 d7 aa >1338 b1 da 95 cf 80 80 80 ca >1530 c9 ff f0 18 c9 27 f0 14
>1148 b5 d0 29 03 a8 a5 d7 0a >1340 d0 f6 a0 46 a2 00 b1 da >1538 c9 28 f0 10 d0 18 ea ea
>1150 aa c0 00 f0 0f c0 01 f0 >1348 95 e4 e8 c0 b1 da 95 e4 >1540 ea ea ea ea ea ea ea ea
>1158 12 c0 02 f0 15 f6 e4 f6 >1350 e8 ea c8 c8 c0 4f d0 ee >1548 ea ea ea ea a6 d3 a9 00
>1160 e5 4c 76 11 d6 e4 d6 e5 >1358 a9 4e 85 d3 a4 d3 ea b1 >1550 9d 21 10 4c 82 15 20 9b
>1168 4c 76 11 f6 e4 d6 e5 4c >1360 da aa c8 b1 da a8 20 6b >1558 3f a0 00 a9 00 91 d8 c8
>1170 76 11 d6 e4 f6 e5 a5 d7 >1368 3f 20 9b 3f a0 00 a9 31 >1560 91 d8 a0 28 91 d8 c8 91
>1178 0a aa b5 e5 a8 b5 e4 aa >1370 91 d8 c8 91 d8 c6 d9 a5 >1568 d8 20 93 3f a0 00 a9 4d
>1180 20 6b 3f 20 9b 3f a0 00 >1378 d8 30 e9 18 b0 02 c6 d9 >1570 91 d8 c8 a9 4f 91 d8 a0
>1188 a9 0c 91 d8 c8 a9 09 91 >1380 85 d0 20 93 3f a0 00 a9 >1578 28 a9 4e 91 d8 c8 a9 50
>1190 d8 c8 a9 0c 91 d8 20 93 >1388 20 91 d8 c8 91 d8 a5 d3 >1580 91 d8 a5 d7 38 e9 03 0a
>1198 3f a5 d7 aa a0 51 b5 d0 >1390 18 69 02 85 d3 c9 52 d0 >1588 18 69 02 c5 d3 f0 07 e6
>11a0 29 04 f0 02 a0 54 98 a0 >1398 c3 a0 52 b1 da 85 e3 a0 >1590 d3 e6 d3 4c c5 14 a4 d7
>11a8 00 91 d8 c8 18 69 01 91 >13a0 1b b1 da 99 35 10 c8 c0 >1598 b1 da aa c8 b1 da a8 20
>11b0 d8 c8 18 69 01 91 d8 e6 >13a8 45 d0 f6 20 00 34 60 a9 >15a0 6b 3f 20 9b 3f ea a9 09
>11b8 d7 a5 d7 c9 03 f0 03 4c >13b0 00 85 d7 a6 d7 bd 51 10 >15a8 91 d8 c8 91 d8 a0 28 91
>11c0 e8 10 60 a2 03 de 79 10 >13b8 a8 c0 00 d0 03 4c a0 14 >15b0 d8 c8 91 d8 20 93 3f a4
>11c8 bd 79 10 c9 ff d0 18 a9 >13c0 bd 50 10 aa 20 6b 3f 20 >15b8 d7 a9 45 c0 00 90 02 a9
>11d0 09 9d 79 10 ca d0 ee a9 >13c8 9b 3f a0 00 a9 00 91 d8 >15c0 49 a0 00 91 d8 18 69 02
>11d8 01 8d 8a 10 a9 00 8d 7a >13d0 c8 91 d8 c8 91 d8 20 93 >15c8 c8 91 d8 38 e9 01 a0 28
>11e0 10 8d 7b 10 8d 7c 10 a2 >13d8 3f a0 00 a9 74 91 d8 c8 >15d0 91 d8 18 69 02 c8 91 d8
>11e8 00 bd 7f 10 dd 84 10 f0 >13e0 a9 75 91 d8 c8 a9 76 91 >15d8 e6 d7 e6 d7 a5 d7 c9 0f
>11f0 03 90 18 b0 0b ea ea ea >13e8 d8 a6 d7 bd 51 10 a8 88 >15e0 f0 03 4c b3 14 60 a9 0f

```

programme

```

>15e8 85 d7 a4 d7 c8 b1 da d0
>15f0 03 4c 3c 17 a5 d7 38 e9
>15f8 0f 0a 85 d3 a6 d3 bd 39
>1600 10 d0 24 ad 02 ff c9 05
>1608 b0 14 a4 d7 b1 da 9d 38
>1610 10 c8 b1 da 9d 39 10 e0
>1618 0c 90 06 de 38 10 4c de
>1620 16 fe 38 10 4c de 16 a8
>1628 bd 38 10 aa 20 6b 3f a5
>1630 d3 c9 0b b0 06 20 27 18
>1638 4c 40 16 a2 22 20 05 18
>1640 ea ea ea ea ea ea ea ea
>1648 ea ea ea ea ea ea a6 d3
>1650 e0 0b 90 06 de 38 10 4c
>1658 5d 16 fe 38 10 bd 39 10
>1660 a8 bd 38 10 aa 20 6b 3f
>1668 a2 00 a0 00 20 7f 16 c8
>1670 20 7f 16 a0 28 20 7f 16
>1678 c8 20 7f 16 4c 87 16 b1
>1680 d8 c9 73 d0 01 e8 60 e0
>1688 00 f0 0a a6 d3 a9 00 9d
>1690 39 10 4c de 16 20 9b 3f
>1698 a0 00 a9 0c 91 d8 c8 91
>16a0 d8 a0 28 91 d8 c8 91 d8
>16a8 20 93 3f a5 d3 a0 00 c9
>16b0 0b 90 17 a9 22 91 d8 c8
>16b8 a9 24 91 d8 a0 28 a9 23
>16c0 91 d8 c8 a9 25 91 d8 4c
>16c8 de 16 a9 1d 91 d8 c8 a9
>16d0 1f 91 d8 a0 28 a9 1e 91
>16d8 d8 c8 a9 21 91 d8 a5 d7
>16e0 38 e9 0f 0a 18 69 02 c5
>16e8 d3 f0 07 e6 d3 e6 d3 4c
>16f0 fc 15 a4 d7 b1 da aa c8
>16f8 b1 da 38 20 6b 3f 20 9b
>1700 3f a9 0d 91 d8 c8 91 d8
>1708 a0 28 91 d8 c8 91 d8 20
>1710 93 3f a6 d7 a0 00 e0 14
>1718 90 12 a9 2f 91 d8 a0 28
>1720 a9 3a 91 d8 c8 a9 3b 91
>1728 d8 4c 3c 17 c8 a9 2d 91
>1730 d8 a0 28 a9 2c 91 d8 c8
>1738 a9 2e 91 d8 e6 d7 e6 d7
>1740 a5 d7 c9 1b f0 03 4c ea
>1748 15 60 ea ea ea ea ea ea
>1750 a9 00 85 d7 a5 d7 0a 0a
>1758 05 d3 a6 d3 bd 6e 10 29
>1760 7f d0 03 4c f9 17 e6 d7
>1768 a5 d7 0a 0a 0a 38 e5 d7
>1770 18 69 03 a8 c6 d7 bd 6e
>1778 10 29 7f aa 20 6b 3f a0
>1780 00 20 9b 3f a9 05 91 d8
>1788 20 93 3f a9 77 91 d8 a6
>1790 d3 c8 98 dd 71 10 d0 e9
>1798 e6 d7 a5 d7 0a 0a 0a 38
>17a0 e5 d7 18 69 03 a8 c6 d7
>17a8 bd 6e 10 29 7f 18 7d 70
>17b0 10 aa 20 6b 3f a6 d3 20
>17b8 9b 3f a9 09 91 d8 20 93
>17c0 3f a9 73 91 d8 c8 98 dd
>17c8 6f 10 d0 eb bd 70 10 f0
>17d0 10 bd 70 10 ea ea 18 7d
>17d8 6f 10 ea ea dd 71 10 d0
>17e0 08 bd 6e 10 49 80 9d 6e
>17e8 10 bd 6e 10 29 80 d0 06
>17f0 de 70 10 4c f9 17 fe 70
>17f8 10 e6 d7 a5 d7 c9 03 f0
>1800 03 4c 54 17 60 a0 00 8a
>1808 d1 d8 d0 04 a9 20 91 d8
>1810 c8 e8 e8 8a d1 d8 d0 04
>1818 a9 20 91 d8 ca c0 29 f0
>1820 05 a0 28 4c 07 18 60 a0
>1828 00 a9 1d d1 d8 d0 04 a9
>1830 20 91 d8 c8 a9 1f d1 d8
>1838 d0 04 a9 20 91 d8 a0 28
>1840 a9 1e d1 d8 d0 04 a9 20
>1848 91 d8 c8 a9 21 d1 d8 d0
>1850 04 a9 20 91 d8 60 ea ea
>1858 ea ea ea ea ea ea ea ea
>1860 ea d4 aa a5 d5 a8 20 6b
>1868 3f 20 9b 3f a5 d6 d0 55
>1870 a5 c6 c9 0a d0 08 4c 00
>1878 1f ea ea ea ea ea ad 43
>1880 05 c9 01 f0 03 4c c5 18
>1888 a5 c6 c9 2f f0 17 c9 2c
>1890 f0 22 a9 04 85 d6 a9 07
>1898 8d 89 10 4c c5 18 a9 03
>18a0 85 d6 4c c5 18 a9 01 85
>18a8 d6 ad 8d 10 09 02 8d 8d
>18b0 10 4c c0 18 a9 02 85 d6
>18b8 ad 8d 10 29 fd 8d 8d 10
>18c0 a9 07 8d 89 10 20 93 3f
>18c8 a0 00 20 e6 18 c8 20 e6
>18d0 18 a0 28 20 e6 18 c8 20
>18d8 e6 18 a0 50 20 e6 18 c8
>18e0 20 e6 18 4c f6 18 b1 d8
>18e8 c9 5b 90 04 c9 73 90 05
>18f0 a9 01 8d 8a 10 60 a0 00
>18f8 a9 20 91 d8 c8 91 d8 a0
>1900 28 91 d8 c8 91 d8 a0 50
>1908 91 d8 c8 91 d8 a6 d6 e0
>1910 03 d0 07 a0 78 91 d8 c8
>1918 91 d8 a5 d6 d0 03 4c 94
>1920 19 c9 01 f0 36 c9 02 f0
>1928 32 c9 04 f0 2e 20 aa 1a
>1930 a0 00 b1 d8 aa 20 65 26
>1938 4c 20 1f ea a9 73 a0 78
>1940 91 d8 c8 91 d8 a9 00 85
>1948 d6 ea ea 4c 94 19 c6 d5
>1950 a9 0a 8d 7d 10 20 c0 1a
>1958 4c c8 19 ad 89 10 c9 04
>1960 f0 0b c9 04 90 05 c6 d5
>1968 4c 6d 19 e6 d5 e0 04 f0
>1970 13 c9 07 f0 0f c9 01 f0
>1978 0b e0 01 f0 05 e6 d4 4c
>1980 84 19 c6 d4 ce 89 10 f0
>1988 03 4c c0 19 a9 00 85 d6
>1990 4c c0 19 ea a5 c6 c9 2f
>1998 f0 07 c9 2c f0 10 4c c0
>19a0 19 c6 d4 ad 8d 10 09 02
>19a8 8d 8d 10 4c b3 19 e6 d4
>19b0 ad 8d 10 29 fd 8d 8d 10
>19b8 ad 8d 10 49 01 8d 8d 10
>19c0 a5 d4 d0 04 a9 01 85 d4
>19c8 a5 d4 c9 26 d0 04 a9 25
>19d0 85 d4 a5 d4 aa a5 d5 a8
>19d8 20 6b 3f a5 d6 d0 14 a0
>19e0 78 b1 d8 c9 73 f0 d0 07 c8
>19e8 b1 d8 c9 73 f0 05 a9 01
>19f0 8d 8a 10 a0 00 20 11 1a
>19f8 c8 20 11 1a a0 28 20 11
>1a00 1a c8 20 11 1a a0 50 20
>1a08 11 1a c8 20 11 1a 4c 1d
>1a10 1a b1 d8 c9 20 f0 05 a9
>1a18 01 8d 8a 10 60 20 9b 3f
>1a20 a0 00 a9 09 91 d8 c8 91
>1a28 d8 a0 28 91 d8 c8 91 d8
>1a30 a0 50 a9 08 91 d8 c8 91
>1a38 d8 a5 d6 c9 03 d0 09 a0
>1a40 78 a9 31 91 d8 c8 91 d8
>1a48 20 93 3f ad 8d 10 d0 02
>1a50 a2 6d c9 01 d0 02 a2 67
>1a58 c9 02 d0 02 a2 61 c9 03
>1a60 d0 02 a2 5b 0a a0 00 91
>1a68 d8 c8 18 69 03 91 d8 a0
>1a70 28 38 e9 02 91 d8 c8 18
>1a78 69 03 91 d8 a0 50 38 e9
>1a80 02 91 d8 c8 18 69 03 91
>1a88 d8 a5 d6 c9 03 d0 09 a0
>1a90 78 a9 73 91 d8 c8 91 d8
>1a98 a5 d4 c9 25 d0 0b 4c a0
>1aa0 1f ea ea ea a9 02 8d 8a
>1aa8 10 60 a5 d8 38 e9 a0 b0
>1ab0 02 c6 d9 85 d8 60 ea ea
>1ab8 ea ea ea ea ea ea ea ea
>1ac0 ad 83 10 18 6d 7d 10 c9
>1ac8 0a 90 19 38 e9 0a a0 a2
>1ad0 04 fe 7e 10 bd 7e 10 c9
>1ad8 0a d0 08 a9 00 9d 7e 10
>1ae0 ca d0 ee 98 8d 83 10 60
>1ae8 a9 00 8d 8d 10 20 6b 3f
>1af0 ad 8e 10 0a 0a 2e 8d 10
>1af8 0a 2e 8d 10 85 da a9 d0
>1b00 ea 18 6d 8d 10 85 db a9
>1b08 00 8d 8d 10 a9 80 8d 8e
>1b10 10 ac 8d 10 b1 da aa a0
>1b18 00 20 9b 3f ad 8f 10 91
>1b20 d8 20 93 3f 8a 2d 8e 10
>1b28 f0 07 a9 73 91 d8 4c 35
>1b30 1b a9 20 91 d8 4e 8e 10
>1b38 c8 c0 08 d0 dc a5 d8 18
>1b40 69 28 90 02 e6 d9 85 d8
>1b48 ee 8d 10 ad 8d 10 c9 08
>1b50 0d ba 60 ea ea ea ea ea
>1b58 ea ea a2 28 a9 02 9d 27
>1b60 08 ca d0 fa a2 28 bd 6f
>1b68 1b 9d 27 0c ca d0 f7 60
>1b70 03 12 05 01 14 05 04 20
>1b78 01 0e 04 20 04 05 13 09
>1b80 07 0e 05 04 20 02 19 20
>1b88 20 20 08 01 12 14 0d 15
>1b90 14 20 20 0f 20 14 20 14
>1b98 ea ea ea ea ea ea ea ea
>1ba0 b1 d8 c9 3c 90 08 c9 45
>1ba8 b0 04 a9 20 91 d8 60 ea
>1bb0 b1 d8 c9 51 90 08 c9 57
>1bb8 b0 04 a9 20 91 d8 c8 ca
>1bc0 d0 ee 60 ea ea ea ea ea
>1bc8 20 48 3f a9 00 85 d3 a6
>1bd0 d3 bd 3c 1c 8d 8f 10 bd
>1bd8 3b 1c 8d 8e 10 bd 3a 1c
>1be0 a8 bd 39 1c aa 20 e8 1a
>1be8 a5 d3 18 69 04 85 d3 c9
>1bf0 20 d0 dc a9 00 8d 0e ff
>1bf8 ad 12 ff 29 fc 09 02 8d
>1c00 12 ff a9 13 8d 11 ff a2
>1c08 00 a0 00 c8 d0 fd e8 d0
>1c10 f8 ad 11 ff 18 69 01 c9
>1c18 19 f0 05 8d 11 ff d0 e7
>1c20 a9 00 8d 11 ff a2 00 a9
>1c28 00 a0 00 c8 d0 fd 18 69
>1c30 01 d0 f6 e8 e0 05 d0 ef
>1c38 60 04 05 07 01 0c 05 01
>1c40 02 14 05 0d 05 1c 05 05
>1c48 03 04 0e 0f 07 0c 0e 16
>1c50 06 14 0e 05 10 1c 0e 12
>1c58 0d ea a9 0a 8d 7d 10 ad

```


>1c60	7a 10 f0 23 20 c0 1a 20	>1e88	ea ea ea ea ea ea ea ea	>20b0	00 00 00 00 00 00 00 00
>1c68	c0 1a 20 c0 1a 20 c0 1a	>1e90	ea ea ea ea ea ea ea ea	>20b8	00 00 00 00 00 00 00 00
>1c70	20 c0 1a 20 c0 1a 20 c0	>1e98	ea ea ea ea ea ea ea ea	>20c0	00 00 00 00 00 00 00 00
>1c78	1a 20 c0 1a 20 c0 1a 20	>1ea0	ea ea ea ea ea ea ea ea	>20c8	00 00 00 00 00 00 00 00
>1c80	c0 1a ce 7a 10 d0 dd ad	>1ea8	ea ea ea ea ea ea ea ea	>20d0	00 00 00 00 00 00 00 00
>1c88	7b 10 f0 08 20 c0 1a ce	>1eb0	ea ea ea ea ea ea ea ea	>20d8	00 00 00 00 00 00 00 00
>1c90	7b 10 d0 f8 ad 7c 10 8d	>1eb8	ea ea ea ea ea ea ea ea	>20e0	00 00 00 00 00 00 00 00
>1c98	7d 10 4c 90 26 60 ea ea	>1ec0	ea ea ea ea ea ea ea ea	>20e8	00 00 32 00 00 00 00 00
>1ca0	20 48 3f a9 00 85 d3 a6	>1ec8	ea ea ea ea ea ea ea ea	>20f0	35 35 35 35 35 35 35 35
>1ca8	d3 bd 14 1d 8d 8f 10 bd	>1ed0	ea ea ea ea ea ea ea ea	>20f8	35 35 35 35 35 35 35 35
>1cb0	13 1d 8d 8e 10 bd 12 1d	>1ed8	ea ea ea ea ea ea ea ea	>2100	35 35 35 35 35 35 35 35
>1cb8	a8 bd 11 1d aa 20 e3 1a	>1ee0	ea ea ea ea ea ea ea ea	>2108	35 35 35 35 35 35 35 35
>1cc0	a5 d3 18 69 04 85 d3 c9	>1ee8	ea ea ea ea ea ea ea ea	>2110	35 35 35 00 00 00 00 00
>1cc8	20 d0 dc a9 00 4c 35 1d	>1ef0	ea ea ea ea ea ea ea ea	>2118	00 00 63 63 63 63 63 63
>1cd0	ad 12 ff 29 fc 09 03 8d	>1ef8	ea ea ea ea ea ea ea ea	>2120	63 63 63 63 63 63 63 63
>1cd8	12 ff a9 00 85 d3 a9 18	>1f00	a0 78 b1 d8 c9 31 d0 0a	>2128	63 63 63 63 63 63 63 63
>1ce0	8d 11 ff a6 d3 bd 0a 1d	>1f08	c8 b1 d8 c9 31 d0 03 4c	>2130	63 63 63 00 00 00 00 00
>1ce8	a8 c0 01 d0 05 a9 10 8d	>1f10	9e 18 4c 7e 18 ea ea ea	>2138	00 00 00 00 00 00 00 00
>1cf0	11 ff a2 00 a9 00 18 69	>1f18	ea ea ea ea ea ea ea ea	>2140	00 00 71 71 71 71 71 71
>1cf8	08 d0 fb e3 d0 f6 88 d0	>1f20	e0 73 90 07 e0 78 b0 03	>2148	71 71 71 71 71 71 71 71
>1d00	e8 e6 d3 a5 d3 c9 07 d0	>1f28	4c 3c 19 4c 4e 19 ea ea	>2150	71 71 71 71 71 71 71 71
>1d08	d5 60 05 05 0a 0a 05 05	>1f30	ea ea ea ea ea ea ea ea	>2158	71 71 71 71 71 71 00 00
>1d10	0a 08 03 07 04 10 03 05	>1f38	ea ea ea ea ea ea ea ea	>2160	04 04 04 04 04 04 04 04
>1d18	06 18 03 14 05 00 0c 12	>1f40	ea ea ea ea ea ea ea ea	>2168	04 04 04 04 04 04 04 04
>1d20	0a 08 0c 05 03 10 0c 01	>1f48	ea ea ea ea ea ea ea ea	>2170	04 04 04 04 04 04 04 04
>1d28	04 18 0c 04 0f 20 0c 19	>1f50	ea ea ea ea ea ea ea ea	>2178	04 04 04 04 04 04 04 04
>1d30	02 ea ea ea ea 8d 0e ff	>1f58	ea ea ea ea ea ea ea ea	>2180	04 04 04 04 04 04 04 04
>1d38	a2 0e a9 03 9d 55 0b a9	>1f60	ea ea ea ea ea ea ea ea	>2188	04 04 04 04 04 04 04 04
>1d40	05 9d a5 0b ca d0 f3 a9	>1f68	ea ea ea ea ea ea ea ea	>2190	04 04 04 04 04 04 04 04
>1d48	0c 0d 56 0f 8d a6 0f 8d	>1f70	ea ea ea ea ea ea ea ea	>2198	04 04 04 04 04 04 04 04
>1d50	5e 0f a9 05 8d 58 0f 8d	>1f78	ea ea ea ea ea ea ea ea	>21a0	04 04 04 04 04 04 04 04
>1d58	5c 0f 8d ac 0f a9 16 8d	>1f80	ea ea ea ea ea ea ea ea	>21a8	04 04 04 04 04 04 04 04
>1d60	5a 0f 8d aa 0f a9 09 8d	>1f88	ea ea ea ea ea ea ea ea	>21b0	04 04 04 04 04 04 04 04
>1d68	a8 0f a9 13 8d ae 0f ad	>1f90	ea ea ea ea ea ea ea ea	>21b8	04 04 04 04 04 04 04 04
>1d70	8c 10 18 69 31 8d 62 0f	>1f98	ea ea ea ea ea ea ea ea	>21c0	04 04 04 04 04 04 04 04
>1d78	ad 8b 10 18 69 30 8d b2	>1fa0	a5 d5 c9 06 f0 07 c9 07	>21c8	04 04 04 04 04 04 04 04
>1d80	0f 4c d0 1c ea ea ea ea	>1fa8	f0 03 4c a9 1a 4c a4 1a	>21d0	04 04 04 04 04 04 04 04
>1d88	ea ea a2 26 a0 08 20 6b	>1fb0	ea ea ea ea ea ea ea ea	>21d8	04 04 04 04 04 04 04 04
>1d90	3f 20 9b 3f a9 f1 a0 00	>1fb8	ea ea ea ea ea ea ea ea	>21e0	77 77 77 77 77 77 77 77
>1d98	91 d8 a0 28 91 d0 20 93	>1fc0	a9 80 8d 47 05 4c 95 25	>21e8	77 77 77 77 77 77 77 77
>1da0	3f a0 00 a9 59 91 d8 a0	>1fc8	ea ea ea ea ea ea ea ea	>21f0	77 77 77 77 77 00 00 46
>1da8	28 a9 5a 91 d8 ee 7e 10	>1fd0	ea ea ea ea ea ea ea ea	>2200	46 77 77 00 00 00 00 00
>1db0	ad 7e 10 c9 1c f0 01 60	>1fd8	ea ea ea ea ea ea ea ea	>2208	00 00 00 00 00 00 00 00
>1db8	a9 00 8d 7e 10 a2 0f kd	>1fe0	ea ea ea ea ea ea ea ea	>2210	00 00 00 00 00 00 00 00
>1dc0	c7 32 e8 9d c7 32 ca ca	>1fe8	ea ea ea ea ea ea ea ea	>2218	00 00 00 00 00 00 00 00
>1dc8	d0 f5 a9 00 8d c8 32 60	>1ff0	ea ea ea ea ea ea ea ea	>2220	00 00 00 00 00 00 00 00
>1dd0	ea ea ea ea ea ea ea ea	>2000	63 63 63 00 32 32 32 32	>2228	00 00 00 00 00 00 00 00
>1dd8	ea ea ea ea ea ea ea ea	>2008	32 32 32 32 32 32 32 32	>2230	63 63 63 63 63 63 63 63
>1de0	ea ea ea ea ea ea ea ea	>2010	32 32 32 32 32 32 32 32	>2238	63 63 63 63 63 63 63 63
>1de8	ea ea ea ea ea ea ea ea	>2018	32 00 00 00 00 00 00 00	>2240	63 63 63 63 63 63 63 63
>1df0	ea ea ea ea ea ea ea ea	>2020	00 00 00 00 00 00 00 00	>2248	63 63 63 63 63 63 63 63
>1df8	ea ea ea ea ea ea ea ea	>2028	00 00 00 00 00 00 00 00	>2250	00 00 00 00 00 00 00 00
>1e00	20 5c 1e ea a9 20 05 db	>2030	00 00 00 00 00 00 00 00	>2258	63 63 63 63 63 63 63 63
>1e08	a9 08 85 d9 a9 f5 85 d8	>2038	00 00 00 00 00 00 00 00	>2260	63 63 63 63 63 63 63 63
>1e10	a0 00 b1 da 91 d8 c8 c0	>2040	00 00 00 00 00 00 00 00	>2268	63 63 63 63 63 63 63 63
>1e18	1e d0 f7 a5 da 18 69 1e	>2048	00 00 00 00 00 00 00 00	>2270	63 63 63 63 63 63 63 63
>1e20	90 02 e6 db 85 da a5 d8	>2050	77 77 77 77 77 77 77 77	>2278	63 63 63 63 63 63 63 63
>1e28	18 69 1e 90 02 e6 d9 85	>2058	00 00 00 00 00 00 00 00	>2280	00 00 00 00 00 00 00 00
>1e30	d8 c9 89 d0 db a5 d9 c9	>2060	00 00 00 00 00 00 00 00	>2288	00 00 00 00 00 00 00 00
>1e38	0f f0 07 a9 0c 85 d9 4c	>2068	00 00 00 00 00 00 00 00	>2290	00 00 00 00 20 20 20 20
>1e40	0c 1e a2 00 a0 00 a9 00	>2070	00 00 00 00 00 00 00 00	>2298	14 08 05 20 0b 09 0e 07
>1e48	18 69 01 d0 fb c8 d0 f6	>2078	77 77 77 77 77 77 77 77	>22a0	20 0f 06 20 0b 20 09 20
>1e50	e8 e0 05 d0 ef a5 c6 c9	>2080	77 77 77 77 77 77 77 77	>22a8	0e 20 07 20 13 20 20 20
>1e58	04 d0 fa 60 20 48 3f a9	>2088	77 77 77 77 77 77 77 77	>22b0	20 20 20 20 20 20 20 20
>1e60	00 85 da a9 00 8d 15 ff	>2090	77 77 77 00 00 00 00 00	>22b8	20 20 20 20 20 20 20 20
>1e68	8d 19 ff 60 ea ea ea ea	>2098	00 77 77 00 00 00 00 00	>22c0	20 20 20 20 20 20 20 20
>1e70	ea ea ea ea ea ea ea ea	>20a0	77 77 77 77 77 00 00 00	>22c8	20 20 20 20 20 20 20 20
>1e78	ea ea ea ea ea ea ea ea	>20a8	00 00 00 00 00 00 00 00	>22d0	20 20 20 20 20 20 20 20
>1e80	ea ea ea ea ea ea ea ea				

programme

>2950 00 00 00 00 11 08 01 16	>2b78 10 10 10 10 10 10 10	>2da0 20 09 13 14 20 14 08 05
>2953 00 00 25 0f 25 16 00 00	>2b80 35 35 32 10 10 10 10	>2da8 20 0e 05 17 20 03 12 0f
>2956 0b 0a 05 15 11 07 00 00	>2b88 00 48 48 00 10 10 10	>2db0 17 0e 20 06 0f 12 20 19
>2958 00 00 00 00 00 00 00 00	>2b90 00 48 48 00 10 10 10	>2db8 0f 15 20 20 20 20 20
>2970 00 00 00 00 00 00 00 00	>2b98 00 48 48 10 10 10 10	>2dc0 20 20 20 20 20 20 20
>2978 00 00 00 00 00 00 00 00	>2ba0 10 10 10 10 10 10 10	>2dc8 20 20 20 20 20 20 20
>2980 00 00 0a 03 04 09 00 00	>2ba8 35 10 32 10 10 10 00	>2dd0 20 20 20 20 20 20 20
>2988 00 00 00 00 00 00 00 00	>2bb0 48 48 48 48 00 10 00	>2dd8 20 20 20 20 20 20 20
>2990 00 00 00 1d 13 07 11 01	>2bb8 48 48 48 48 00 10 00	>2de0 20 20 20 20 20 20 20
>2998 55 01 15 01 0b 00 00 00	>2bc0 48 48 48 10 10 10 10	>2de8 20 20 20 20 20 20 20
>29a0 00 00 00 00 00 00 00 01	>2bc8 10 10 10 10 10 10 10	>2df0 20 20 20 20 20 79 78 20
>29a8 00 01 0f 00 00 00 00 00	>2bd0 35 10 32 10 10 10 00	>2df8 20 20 20 20 20 79 78 20
>29b0 00 00 00 0e 0a 04 14 0a	>2bd8 48 48 48 48 48 00 00	>2e00 20 20 20 20 20 79 78 20
>29b8 07 11 11 03 13 11 02 1d	>2be0 48 48 48 48 48 00 00	>2e08 20 20 20 20 20 20 20
>29c0 11 00 19 18 04 00 00 00	>2be8 40 48 48 10 10 10 10	>2e10 20 20 20 20 10 20 0b 20
>29c8 00 00 00 00 00 00 00 00	>2bf0 10 10 10 10 10 10 10	>2e18 20 20 20 20 20 79 7f 78
>29d0 00 00 00 00 00 00 00 00	>2bf8 35 10 10 10 10 10 48	>2e20 20 20 20 20 79 7f 7f 78
>29d8 00 05 04 09 11 01 06 06	>2c00 48 48 48 48 48 48 48	>2e28 20 20 20 20 79 7f 7f 20
>29e0 00 00 00 00 00 00 21 18	>2c08 48 48 48 48 48 48 48	>2e30 20 20 20 20 20 20 20
>29e8 08 11 03 ea ea ea ea	>2c10 48 48 48 10 10 10 10	>2e38 20 20 20 20 12 20 05 20
>29f0 20 48 3f a9 90 85 da a9	>2c18 10 10 10 10 10 10 10	>2e40 20 20 20 79 7f 7f 7f 7f
>29f8 2a 85 db a9 08 85 d9 a9	>2c20 35 10 32 32 10 10 29 29	>2e48 78 20 20 79 7f 7f 7f 7f
>2a00 f5 85 d0 a0 00 b1 da 91	>2c28 29 29 29 29 29 29 29	>2e50 78 20 20 79 7f 7f 7f 20
>2a08 d8 c8 c0 1e d0 f7 a5 da	>2c30 29 29 29 29 29 29 29	>2e58 20 20 20 20 20 20 20
>2a10 18 69 1e 90 02 e6 db 85	>2c38 29 29 29 10 10 10 10	>2e60 20 20 20 20 05 20 19 20
>2a18 da a5 d3 18 69 1e 90 02	>2c40 10 10 10 10 10 10 10	>2e68 20 20 79 7f 7f 7f 7f 7f
>2a20 e6 d9 85 d3 c9 89 d0 db	>2c48 10 10 32 32 10 10 5f 5f	>2e70 7f 78 79 7f 7f 7f 7f 7f
>2a28 a5 d9 c9 0f f0 07 a9 0c	>2c50 5f 5f 5f 5f 5f 5f 5f	>2e78 7f 78 79 7f 7f 7f 7f 20
>2a30 85 d9 4c ff 29 ea ea ea	>2c58 5f 5f 5f 5f 5f 5f 5f	>2e80 20 20 20 20 20 20 20
>2a38 ea ea ea ea ea ea ea ea	>2c60 5f 5f 5f 10 10 10 10	>2e88 20 20 20 20 13 20 20 20
>2a40 ea ea ea ea ea ea ea ea	>2c68 10 10 10 10 10 10 10	>2e90 20 20 7f 7f 7f 7f 7f 7f
>2a48 ea ea ea ea ea ea 60 a2	>2c70 35 10 10 10 10 10 2e 2e	>2e98 7f 7f 7f 7f 7f 7f 7f 7f
>2a50 00 a0 00 c8 d0 fd e8 d0	>2c78 2e 2e 2e 2e 2e 2e 2e	>2ea0 7f 7f 7f 7f 7f 7f 7f 20
>2a58 f8 20 4e 34 20 f0 29 a9	>2c80 2e 2e 2e 2e 2e 2e 2e	>2ea8 20 20 20 20 20 20 20
>2a60 04 85 e3 20 00 34 a2 00	>2c88 2e 2e 2e 10 10 10 10	>2eb0 20 20 20 20 13 20 14 20
>2a68 a9 00 a0 00 c8 d0 fd 18	>2c90 10 10 10 10 10 10 10	>2eb8 20 20 7f 7f 7f 7f 7f 7f
>2a70 69 01 d0 f6 e8 e0 05 d0	>2c98 35 10 32 32 10 10 6d 6d	>2ec0 7f 7f 7f 7f 7f 7f 7f 7f
>2a78 ef a5 c6 c9 40 f0 fa 20	>2ca0 6d 6d 6d 6d 6d 6d 6d	>2ec8 7f 7f 7f 7f 7f 7f 7f 20
>2a80 4e 34 60 ea ea ea ea ea	>2ca8 6d 6d 6d 6d 6d 6d 6d	>2ed0 20 20 20 20 20 20 20
>2a88 ea ea ea ea ea ea ea ea	>2cb0 6d 6d 6d 10 10 10 10	>2ed8 20 20 20 20 20 20 0f 20
>2a90 32 32 10 10 10 00 00 00	>2cb8 10 10 10 10 10 10 10	>2ee0 20 20 7f 7f 7f 7f 7f 7f
>2a98 00 00 00 00 00 00 00 00	>2cc0 35 10 32 32 10 63 6b 6b	>2ee8 7f 7f 7f 7f 7f 7f 7f 20
>2aa0 00 00 00 00 00 00 00 00	>2cc8 6b 6b 6b 6b 6b 6b 6b	>2ef0 7f 7f 7f 7f 7f 7f 7f 20
>2aa8 00 00 00 10 10 10 10 10	>2cd0 6b 6b 6b 6b 6b 6b 6b	>2ef8 20 20 20 20 20 20 20 20
>2ab0 10 10 10 10 10 10 10 10	>2cd8 6b 6b 6b 10 10 10 10	>2f00 20 20 20 20 01 20 20 20
>2ab8 10 10 10 71 71 71 71 71	>2ce0 10 10 10 10 10 10 10	>2f08 20 20 7f 7f 7f 7f 7f 7f
>2ac0 71 71 71 71 71 71 71 71	>2ce8 10 10 10 10 5a 29 29	>2f10 7f 7f 7f 7f 7f 7f 7f 7f
>2ac8 71 71 71 71 71 71 71 71	>2cf0 29 29 29 29 29 29 29	>2f18 7f 7f 7f 7f 7f 7f 7f 20
>2ad0 71 71 71 71 10 10 10 10	>2cf8 29 29 29 29 29 29 29	>2f20 20 20 20 20 20 20 20
>2ad8 10 10 10 10 10 10 10 10	>2d00 29 29 29 10 10 32 10 10	>2f28 20 20 20 20 0e 20 07 0f
>2ae0 10 10 63 63 63 63 63 63	>2d08 10 10 35 35 35 35 35 35	>2f30 20 20 7f 7f 7f 7f 7f 7f
>2ae8 63 63 63 63 63 63 63 63	>2d10 35 35 35 35 35 35 35	>2f38 7f 7f 7f 7f 7f 7f 7f 7f
>2af0 63 63 63 63 63 63 63 63	>2d18 35 35 35 35 35 35 35	>2f40 7f 7f 7f 7f 7f 7f 7f 20
>2af8 63 63 63 63 10 10 10 10	>2d20 35 35 35 35 20 20 20	>2f48 20 20 20 20 20 20 20 20
>2b00 10 10 10 10 10 10 10 10	>2d28 20 19 0f 15 20 08 01 16	>2f50 20 20 20 20 19 20 0f 0e
>2b08 77 77 77 77 77 77 77 77	>2d30 05 20 04 05 13 14 12 0f	>2f58 20 20 7f 7f 7f 7f 7f 7f
>2b10 77 77 77 77 77 77 77 77	>2d38 19 05 04 20 14 08 05 20	>2f60 7f 7f 7f 7f 7f 7f 7f 7f
>2b18 77 77 77 77 77 77 77 77	>2d40 20 20 20 20 20 20 20	>2f68 7f 7f 7f 7f 7f 7f 7f 20
>2b20 77 77 77 77 77 77 10 10	>2d48 20 20 20 20 20 20 0d	>2f70 20 20 20 20 20 20 20 20
>2b28 10 10 10 10 10 10 10 10	>2d50 01 04 20 0b 09 0e 07 20	>2f78 20 20 20 20 20 20 20 20
>2b30 10 10 10 10 10 10 10 10	>2d58 0f 06 20 07 01 0c 01 03	>2f80 20 20 7b 7b 7b 7c 7c 7c
>2b38 10 10 10 10 10 10 10 10	>2d60 14 09 03 01 20 01 0e 04	>2f88 7d 7d 7d 7e 7e 7e 7d 7d
>2b40 10 10 10 10 10 10 10 10	>2d68 20 20 20 20 20 20 20	>2f90 7d 7c 7c 7c 7b 7b 7b 20
>2b48 10 10 10 10 10 10 10 10	>2d70 20 20 20 20 20 20 13	>2f98 20 20 20 20 20 20 20 20
>2b50 10 10 10 10 10 10 10 10	>2d78 01 16 05 04 20 03 09 13	>2fa0 20 20 20 20 20 20 20 20
>2b58 10 10 10 10 10 10 10 10	>2d80 20 10 05 0f 10 0c 05 20	>2fa8 20 20 20 20 20 20 20 20
>2b60 10 00 00 10 10 10 10 10	>2d88 06 12 0f 0d 20 03 09 0d	>2fb0 20 20 20 20 20 20 20 20
>2b68 10 00 00 10 10 10 10 10	>2d90 20 20 20 20 20 20 20	>2fb8 ea ea ea ea ea ea ea ea
>2b70 10 00 00 10 10 10 10 10	>2d98 20 20 20 20 08 05 12 05	>2fc0 ea ea ea ea ea ea ea ea

programme

```

>2fc8 ea ea ea ea ea ea ea ea
>2fd0 ea ea ea ea ea ea ea ea
>2fd8 ea ea ea ea ea ea ea ea
>2fe0 ea ea ea ea ea ea ea ea
>2fe8 ea ea ea ea ea ea ea ea
>2ffa0 ea ea ea ea ea ea ea ea
>2ffa8 ea ea ea ea ea ea ea ea
>3000 66 00 7e 66 7e 66 66 00
>3008 7e 7e 66 66 7e 66 66 00
>3010 7c 66 66 7e 66 66 7c 00
>3018 7e 7e 60 60 60 7e 7e 00
>3020 7c 66 66 66 66 66 7c 00
>3028 7e 60 60 78 60 60 7e 00
>3030 7e 60 60 78 60 60 60 00
>3038 3e 7e 60 7e 66 7e 3e 00
>3040 66 66 66 7e 66 66 66 00
>3048 18 18 18 18 18 18 18 00
>3050 7e 06 06 06 06 6c 38 00
>3058 66 6e 78 70 78 6e 66 00
>3060 60 60 60 60 60 7e 7e 00
>3068 c6 ee fe d6 c6 c6 c6 00
>3070 66 76 76 7e 6e 6e 66 00
>3078 3c 7e 66 66 66 7e 3c 00
>3080 7c 66 66 7c 60 60 60 00
>3088 3c 66 66 66 66 3c 0e 00
>3090 7c 66 66 7c 78 6c 66 00
>3098 3c 66 60 3c 06 66 3c 00
>30a0 7e 7e 18 18 18 18 18 00
>30a8 66 66 66 66 66 7e 7e 00
>30b0 66 66 66 66 66 3c 18 00
>30b8 c6 c6 c6 d6 fe ee c6 00
>30c0 66 66 3c 18 3c 66 66 00
>30c8 66 66 66 3c 0c 18 30 00
>30d0 7e 7e 0c 18 30 7e 7e 00
>30d8 66 00 3c 66 66 66 3c 00
>30e0 66 00 66 66 66 7e 7e 00
>30e8 05 14 10 53 43 4f 4f 4f
>30f0 4f 5f 1f 1f 0f 01 01 01
>30f8 00 00 f0 fc fc ef ef ff
>3100 00 00 00 00 00 00 00 00
>3108 fc f0 fc ff ff 04 04 45
>3110 00 00 0f 3f 3f fb fb ff
>3118 3f 0f 3f ff ff 10 10 51
>3120 50 14 04 c5 c1 f1 f1 f1
>3128 f1 f5 f4 f4 f0 40 40 40
>3130 18 18 18 18 18 18 18 18
>3138 00 00 00 ff ff 00 00 00
>3140 18 18 18 1f 1f 00 00 00
>3148 18 18 18 f8 f8 00 00 00
>3150 00 00 00 f8 f8 18 18 18
>3158 00 00 00 1f 1f 18 18 18
>3160 03 0f 0f 3c f0 f0 c0 c0
>3168 0c 0d 3d 3d fd fd fd fd
>3170 fd fd fd fd 3d 3d 0d 0c
>3178 30 70 7c 7c 7f 7f 7f 7f
>3180 7e 66 6e 7e 76 66 7e 00
>3188 18 18 18 18 18 18 18 00
>3190 7e 06 06 7e 60 60 7e 00
>3198 7e 06 06 1e 06 06 7e 00
>31a0 60 60 66 7e 7e 06 06 00
>31a8 7e 60 60 7e 06 06 7e 00
>31b0 7e 60 60 7e 66 66 7e 00
>31b8 7e 7e 06 0c 18 30 60 00
>31c0 3c 66 66 7e 66 66 3c 00
>31c8 7e 66 66 7e 06 06 7e 00
>31d0 7f 7f 7f 7f 7c 7c 70 30
>31d8 c0 f0 fc 3c 0f 0f 03 03
>31e0 20 20 20 20 20 20 1f 1f
>31e8 20 20 20 20 20 20 23

>31f0 23 23 23 20 20 20 20 20
>31f8 3f f3 ff 30 3f 03 ff ff
>3200 35 3d 3d 3d 3d 3f ff f0
>3208 f0 c0 c0 f0 f0 30 f0
>3210 f0 3c fc 30 f0 00 fc fc
>3218 7f f3 f3 f3 f3 f0 fc 3f
>3220 3f 0f 0f 3c 3c 30 30 3c
>3228 00 00 00 aa aa ff ff ff
>3230 ff ff ff aa aa 00 00 00
>3238 00 00 00 a8 a9 fd fd fd
>3240 fd fd fd a9 a8 00 00 00
>3248 00 00 00 2a 6a 6f 6f 6f
>3250 6f 6f 6f 6a 2a 00 00 00
>3258 00 00 00 aa aa ff ff ff
>3260 ff ff ff aa aa 00 00 00
>3268 00 0f 3f ff ff fd f7 f7
>3270 f7 f7 f7 fd ff 3f 0f 00
>3278 00 f0 fc ff 7f ff ff ff
>3280 ff ff ff ff 7f fc f0 00
>3288 c0 f0 fc 3f 0f 03 00 00
>3290 82 aa 28 ff f7 f7 ff 3c
>3298 03 0f 3f fc f0 c0 00 00
>32a0 00 00 00 0f 3f ff f0 c0
>32a8 82 aa 28 ff df df ff 3c
>32b0 00 00 00 f0 fc ff 0f 03
>32b8 3c ff c3 c3 c3 ff 3c 20
>32c0 20 20 20 20 28 28 20 28
>32c8 00 00 00 00 00 00 3c ff
>32d0 c3 c3 c3 ff 3c 20 20 20
>32d8 0f 3e 3e 0f 0f 03 03 0f
>32e0 0d 05 17 5f 4f 0f 0f 03
>32e8 02 02 03 0f 0c 0c 3c 3c
>32f0 f0 f0 f0 f0 f0 c0 c0 f0
>32f8 f0 f0 f4 f5 f1 f0 f0 c0
>3300 80 a0 20 28 08 08 28 28
>3308 0f 3e 3e 0f 0f 03 03 0f
>3310 0f 0f 1f 5f 4f 0f 0f 03
>3318 02 02 02 0a 08 08 28 28
>3320 f0 f0 f0 f0 f0 c0 c0 f0
>3328 70 50 d4 f5 f1 f0 f0 c0
>3330 80 b0 30 3c 0c 0c 3c 0c
>3338 0f 0f 0f 0f 0f 03 03 0f
>3340 0f 0f 1f 5f 4f 0f 0f 03
>3348 02 0a 08 28 20 20 28 28
>3350 f0 bc bc f0 f0 c0 c0 f0
>3358 70 50 d4 f5 f1 f0 f0 c0
>3360 80 80 c0 f0 30 30 3c 3c
>3368 0f 0f 0f 0f 0f 03 03 0f
>3370 0d 05 17 5f 4f 0f 0f 03
>3378 02 0e 0c 3c 30 30 3c 3c
>3380 f0 bc bc f0 f0 c0 c0 f0
>3388 f0 f0 f4 f5 f1 f0 f0 c0
>3390 80 80 80 a0 20 20 28 28
>3398 e7 e7 e7 00 7e 7e 7e 00
>33a0 00 f0 64 64 67 64 64 00
>33a8 00 00 00 00 bd a5 bd 00
>33b0 00 00 00 0f e9 2f 21 0f
>33b8 cc cc cc cc cc ff ff
>33c0 80 c0 e0 f0 f8 fc fe ff
>33c8 01 03 07 0f 1f 3f 7f ff
>33d0 3c ff ff eb eb ff ff 3c
>33d8 ff ff ff ff ff ff ff 55
>33e0 ff ff ff ff ff ff ff 55
>33e8 ff ff ff ff 55 55 55 55
>33f0 ff ff ff ff 55 55 55 55
>33f8 ff bb bb aa bb bb ff ff
>3400 ea ea ea ea ea ea ea a5
>3408 e3 0a 0a aa bd 36 34 85
>3410 dd bd 37 34 85 de bd 38

>3418 34 85 e0 bd 39 34 85 e1
>3420 a9 00 85 df 85 e2 a9 38
>3428 8d 11 ff a9 5e 8d 14 03
>3430 a9 34 8d 15 03 60 45 35
>3438 39 36 85 37 4f 38 33 39
>3440 2f 3a d3 3a d3 3b ab 3c
>3448 73 3d ea ea ea ea a9 0e
>3450 8d 14 03 a9 ce 8d 15 03
>3458 a9 00 8d 11 ff 60 a5 df
>3460 c9 00 d0 3c a0 01 b1 dd
>3468 d0 0f a5 e3 0a 0a aa bd
>3470 36 34 85 dd bd 37 34 85
>3478 de a0 00 b1 dd 0a aa bd
>3480 ea 34 8d 0e ff ad 12 ff
>3488 29 fc 1d e9 34 8d 12 ff
>3490 c8 b1 dd 85 df a5 dd 18
>3498 69 02 90 02 e6 de 85 dd
>34a0 c6 df a5 e2 c9 00 d0 3c
>34a8 a0 01 b1 e0 d0 0f a5 e3
>34b0 0a 0a aa bd 38 34 85 e0
>34b8 bd 39 34 85 e1 a0 00 b1
>34c0 e0 0a aa bd ea 34 8d 0f
>34c8 ff ad 10 ff 29 fc 1d e9
>34d0 34 8d 10 ff c8 b1 e0 85
>34d8 e2 a5 e0 18 69 02 90 02
>34e0 c6 e1 85 e0 c6 e2 4c 0e
>34e8 ce 03 fe 00 08 00 40 00
>34f0 76 00 a9 00 d9 01 06 01
>34f8 31 01 59 01 7f 01 a3 01
>3500 c5 01 e6 02 04 02 20 02
>3508 3b 02 54 02 6d 02 83 02
>3510 99 02 ac 02 c0 02 d2 02
>3518 e3 02 f3 03 02 03 10 03
>3520 1e 03 2a 03 36 03 42 03
>3528 4c 03 56 03 60 03 69 03
>3530 71 03 79 03 81 03 88 03
>3538 8f 03 95 03 9b 03 a1 03
>3540 a6 03 ab ea ea 1b 80 1e
>3548 10 1c 10 1b 10 1c 10 1e
>3550 30 1b 10 17 40 00 40 23
>3558 60 25 10 23 10 22 10 20
>3560 10 1e 10 1d 10 1e 30 1b
>3568 10 17 40 00 40 1c 30 19
>3570 10 16 20 19 20 1b 20 1c
>3578 20 1e 30 1b 10 23 40 00
>3580 40 25 10 23 10 22 10 23
>3588 10 22 10 20 10 1f 10 20
>3590 10 1e 10 1c 10 1b 10 1c
>3598 10 1b 30 1c 08 1b 08 19
>35a0 20 1e 20 1d 20 1c 20 1b
>35a8 50 17 10 1b 10 1e 10 1c
>35b0 10 19 10 1c 10 20 10 1e
>35b8 30 1b 10 17 10 00 03 17
>35c0 0d 19 10 1b 10 1c 10 1e
>35c8 10 20 10 22 10 23 10 22
>35d0 10 23 10 22 10 25 10 23
>35d8 10 22 10 23 10 22 10 20
>35e0 10 1e 10 1d 10 1e 10 1b
>35e8 10 1e 10 1b 10 17 40 00
>35f0 40 1c 10 19 10 1c 10 19
>35f8 10 16 10 19 10 18 10 19
>3600 10 1a 10 1b 10 1c 10 1d
>3608 10 1e 10 1b 10 1e 10 1b
>3610 10 23 50 27 10 22 10 23
>3618 10 1f 10 20 10 1c 10 19
>3620 10 17 2e 00 04 17 0e 1b
>3628 10 19 10 17 10 19 10 00
>3630 04 19 3c 17 7f 00 01 00
>3638 00 0b 10 12 10 0f 10 12

```

programme

```
>3640 10 0b 10 12 10 0f 10 12 >3868 0c 10 0c 14 0c 17 0c 14 >3a90 20 0b 00 0d 00 00 20 06
>3648 10 0d 10 12 10 10 10 12 >3870 0c 10 0c 15 0c 19 0c 15 >3a98 20 0a 20 0d 20 12 20 0d
>3650 10 0b 10 12 10 0f 10 12 >3878 0c 10 0c 15 0c 19 0c 15 >3aa0 20 06 20 12 20 05 20 0d
>3658 10 0b 10 12 10 0f 10 12 >3880 0c 10 0c 15 0c 19 0c 15 >3aa8 20 08 20 0d 20 05 20 0d
>3660 10 0b 10 12 10 0f 10 12 >3888 0c 10 0c 15 0c 19 0c 15 >3ab0 20 08 20 0d 20 06 20 0d
>3668 10 0b 10 14 10 10 10 14 >3890 0c 10 0c 14 0c 17 0c 14 >3ab8 20 0a 20 0d 20 06 20 0d
>3670 10 0b 10 14 10 10 10 14 >3898 0c 10 0c 14 0c 17 0c 14 >3ac0 20 0a 20 0d 20 0b 00 0d
>3678 10 0b 10 11 10 0e 10 11 >38a0 0c 0f 0c 12 0c 17 0c 12 >3ac8 40 00 02 0d 3e 0a 00 00
>3680 10 0b 10 12 10 0f 10 12 >38a8 0c 0f 0c 12 0c 17 0c 12 >3ad0 80 00 00 17 20 1b 20 1e
>3688 10 0b 10 12 10 0f 10 12 >38b0 0c 10 18 14 18 12 18 10 >3ad8 20 1b 20 17 20 1e 20 1b
>3690 10 0b 10 12 10 0f 10 12 >38b8 18 00 02 10 2e 0f 18 00 >3ae0 20 17 20 23 20 22 05 23
>3698 10 0d 10 12 10 10 10 12 >38c0 18 10 0c 14 0c 17 0c 14 >3ae8 05 22 36 1e 20 25 20 22
>36a0 10 0d 10 12 10 10 10 12 >38c8 0c 10 0c 14 0c 17 0c 14 >3af0 20 1e 20 25 20 22 20 1e
>36a8 10 0d 10 12 10 10 10 12 >38d0 0c 00 09 10 0c 14 0c 17 >3af8 20 23 20 1e 20 1b 20 21
>36b0 10 0b 10 12 10 0f 10 12 >38d8 0c 14 0c 10 0c 14 0c 17 >3b00 20 1e 20 1b 20 21 20 1e
>36b8 10 08 10 0f 10 0b 10 0f >38e0 0c 14 0c 10 0c 15 0c 19 >3b08 20 1b 20 20 20 1c 20 19
>36c0 10 03 10 0b 10 06 10 0b >38e8 0c 15 0c 10 0c 15 0c 19 >3b10 20 16 20 19 20 1c 20 20
>36c8 10 04 40 0b 40 08 40 0b >38f0 0c 15 0c 00 09 10 0c 15 >3b18 20 1e 20 1c 20 1e 20 1b
>36d0 40 0a 40 00 40 0b 10 12 >38f8 0c 19 0c 15 0c 10 0c 15 >3b20 20 17 20 14 20 17 20 1b
>36d8 10 0f 10 12 10 0b 10 12 >3900 0c 19 0c 15 0c 10 0c 14 >3b28 20 1e 20 1c 20 1b 20 1c
>36e0 10 0f 10 12 10 0d 10 12 >3908 0c 17 0c 14 0c 11 0c 14 >3b30 20 19 20 16 20 12 20 16
>36e8 10 10 10 12 10 0b 10 12 >3910 0c 19 0c 14 0c 12 0c 15 >3b38 20 19 20 1c 20 1b 20 19
>36f0 10 0f 10 12 10 0b 10 12 >3918 0c 19 0c 15 0c 0f 0c 12 >3b40 20 1b 05 19 05 1b 16 1c
>36f8 10 0f 10 12 10 0b 10 12 >3920 0c 17 0c 12 0c 10 10 09 >3b48 20 1e 20 20 20 1b 20 23
>3700 10 0f 10 12 10 0b 10 14 >3928 18 0b 19 12 19 14 30 00 >3b50 20 20 20 1b 20 23 20 19
>3708 10 10 10 14 10 0b 10 14 >3930 30 00 00 1b 10 19 10 18 >3b58 20 1b 20 1d 20 1e 20 19
>3710 10 10 10 14 10 0b 10 11 >3938 10 19 10 1e 60 19 20 16 >3b60 20 22 20 1e 20 19 20 22
>3718 10 0e 10 11 10 0b 10 12 >3940 20 19 20 1e 20 22 20 19 >3b68 20 17 05 16 05 17 76 19
>3720 10 0f 10 12 10 0b 10 12 >3948 04 25 60 22 10 1e 10 19 >3b70 20 17 20 16 20 17 20 14
>3728 10 0f 10 12 10 0b 10 12 >3950 20 00 20 1e 10 20 10 1e >3b78 20 16 20 17 20 16 20 14
>3730 10 0f 10 12 10 0d 10 12 >3958 10 1d 10 1b 20 00 20 1e >3b80 20 16 20 12 20 19 20 1b
>3738 10 10 10 12 10 0d 10 12 >3960 10 20 10 1e 10 1d 10 1b >3b88 20 1c 20 1b 05 19 05 1b
>3740 10 10 10 12 10 0d 10 12 >3968 20 00 20 2a 10 27 10 2a >3b90 16 1c 20 1b 20 19 20 1b
>3748 10 10 10 12 10 0b 10 12 >3970 10 27 10 25 10 2a 10 29 >3b98 20 17 20 20 20 22 20 23
>3750 10 0f 10 12 10 08 10 0f >3978 10 27 10 25 10 23 10 22 >3ba0 20 1d 05 1b 05 1d 16 1e
>3758 10 0b 10 0f 10 03 10 0b >3980 10 20 10 1e 40 00 20 16 >3ba8 20 1d 20 1b 20 1d 20 19
>3760 10 06 10 0b 10 04 10 0d >3988 20 17 20 00 02 17 1e 19 >3bb0 20 23 20 25 20 27 20 25
>3768 10 08 10 0d 10 06 10 0f >3990 10 17 10 16 10 17 10 20 >3bb8 20 23 20 22 20 20 20 22
>3770 10 0b 10 0f 10 06 10 10 >3998 60 17 20 16 20 00 02 16 >3bc0 20 1e 20 19 20 1e 20 1d
>3778 10 0d 10 10 10 00 04 10 >39a0 1e 17 10 16 10 14 10 16 >3bc8 20 1e 05 1d 05 1e 36 00
>3780 3c 0f 80 00 00 23 18 21 >39a8 10 1e 60 16 20 00 02 16 >3bd0 80 00 00 00 60 00 c0 06
>3788 0c 20 0c 1e 0c 1c 0c 00 >39b0 1e 14 20 17 20 1b 20 00 >3bd8 20 0a 20 0d 20 0a 20 0d
>3790 02 1c 16 00 02 1c 16 00 >39b8 02 1b 1e 11 20 14 20 19 >3be0 20 0d 20 0a 20 06 20 0d
>3798 02 1c 46 28 18 27 0c 25 >39c0 20 17 00 16 40 00 20 16 >3be8 20 0f 05 10 05 0f 36 0b
>37a0 0c 23 0c 21 0c 00 02 21 >39c8 20 17 10 14 10 16 10 17 >3bf0 20 12 20 0f 20 0b 20 12
>37a8 16 00 02 21 16 00 02 21 >39d0 10 19 10 1b 10 1d 10 1e >3bf8 20 0f 20 0b 20 10 20 14
>37b0 46 25 18 00 02 25 0a 23 >39d8 10 20 10 1e 10 1d 10 1b >3c00 20 10 20 0d 20 10 20 0d
>37b8 0c 21 0c 20 0c 00 02 20 >39e0 10 19 10 17 10 16 10 14 >3c08 20 0a 20 0b 20 0d 20 0f
>37c0 16 23 18 00 02 23 0a 20 >39e8 10 16 10 12 10 14 10 16 >3c10 20 12 20 0f 20 0b 20 0f
>37c8 0c 20 0c 1e 0c 00 02 1e >39f0 10 17 10 19 10 1b 10 1d >3c18 20 0b 20 08 20 0a 20 0b
>37d0 16 21 18 00 02 21 0a 20 >39f8 10 1e 10 20 10 22 10 23 >3c20 20 0d 20 10 20 0d 20 0a
>37d8 0c 1e 0c 1c 0c 1b 18 1c >3a00 10 25 10 27 10 29 10 2a >3c28 20 0d 20 0a 20 06 20 08
>37e0 18 20 30 1e 18 23 18 21 >3a08 10 00 10 1b 10 20 10 23 >3c30 20 0a 20 0b 05 0a 05 0b
>37e8 0c 20 0c 1e 0c 1c 0c 00 >3a10 10 27 10 25 10 23 10 22 >3c38 76 0a 20 00 20 06 20 08
>37f0 02 1c 16 00 02 1c 16 1b >3a18 10 20 10 1e 10 1d 10 1b >3c40 20 05 20 0a 05 08 05 0a
>37f8 03 1c 03 1e 03 1c 48 28 >3a20 10 19 10 17 10 16 10 14 >3c48 76 08 20 06 20 05 20 06
>3800 18 27 0c 25 0c 23 0c 21 >3a28 10 12 00 00 00 00 00 00 >3c50 20 03 20 08 20 0a 20 0b
>3808 0c 00 02 21 16 00 02 21 >3a30 40 06 20 0d 20 0a 20 0d >3c58 20 0d 20 08 20 11 20 0d
>3810 16 20 03 21 03 23 03 21 >3a38 20 06 20 0d 20 0a 20 0d >3c60 20 08 20 11 20 12 05 10
>3818 48 25 18 00 02 25 0a 23 >3a40 20 06 24 0d 20 0a 20 0d >3c68 05 12 76 06 20 08 20 0a
>3820 0c 21 0c 20 0c 00 02 20 >3a48 20 06 20 0d 20 0a 20 0d >3c70 20 08 20 06 20 0b 05 0a
>3828 16 23 18 00 02 23 0a 21 >3a50 20 06 20 0f 20 0b 20 0f >3c78 05 0b 76 08 20 0a 20 0b
>3830 0c 20 0c 1e 0c 00 02 1e >3a58 20 06 20 0f 20 0b 20 0f >3c80 20 0a 20 08 20 0d 05 0b
>3838 16 21 18 00 02 21 0a 20 >3a60 20 06 20 0d 20 0a 20 0d >3c88 05 0d 76 0b 20 0a 20 08
>3840 0c 21 0c 1e 0c 1c 18 1b >3a68 20 0a 20 0d 20 06 20 0d >3c90 20 06 20 05 20 06 20 08
>3848 18 1c 30 00 30 00 00 00 >3a70 20 05 20 0d 20 08 20 0d >3c98 20 0a 20 0b 20 0d 20 0f
>3850 18 10 0c 14 0c 17 0c 14 >3a78 20 05 20 0d 20 08 20 0d >3ca0 20 0d 40 01 20 06 40 00
>3858 0c 10 0c 14 0c 17 0c 14 >3a80 20 06 20 0d 20 0a 20 0d >3ca8 80 00 00 1f 0c 21 0c 23
>3860 0c 10 0c 14 0c 17 0c 14 >3a88 20 06 20 0d 20 0a 20 0d >3cb0 0c 21 24 1a 18 21 0c 23
```

programme

```

>3cb8 0c 24 0c 23 24 1a 18 1c >3dd8 18 0a 18 13 18 0b 18 13 >3ee8 20 00 02 26 0e 25 10 23
>3cc0 0c 00 02 1c 16 1b 0c 1c >3dd8 18 0e 18 02 18 07 18 0e >3ef0 10 21 10 2a 10 26 10 21
>3cc8 0c 1e 0c 1f 18 1a 18 00 >3de0 18 02 18 0e 18 07 18 0e >3ef8 10 23 10 1f 10 1c 10 19
>3cd0 02 1a 22 17 0c 18 0c 19 >3de8 18 02 18 0e 18 0b 18 0f >3f00 10 1a 20 00 60 00 00 02
>3cd8 0c 1a 18 1f 18 1e 18 1c >3df0 18 12 18 17 18 04 18 0b >3f08 20 0e 20 02 20 0e 20 02
>3ce0 18 1a 0c 1f 18 1e 18 1c >3df8 18 10 30 0c 30 0d 30 0e >3f10 20 0e 20 02 20 0e 20 02
>3ce8 0c 1a 0c 17 0c 1a 18 1f >3e00 30 10 18 04 18 09 30 02 >3f18 20 0e 20 04 20 0e 20 15
>3cf0 18 1e 18 1c 18 1a 0c 1e >3e08 30 13 18 0e 18 07 18 00 >3f20 20 09 20 0b 20 0d 20 02
>3cf8 18 1c 18 1a 0c 15 0c 17 >3e10 18 00 00 a9 a3 85 dd a9 >3f28 20 09 20 02 20 0c 20 07
>3d00 0c 18 18 1e 18 1c 18 1a >3e18 3e 85 de a9 07 85 e0 a9 >3f30 20 0e 20 06 20 11 20 09
>3d08 18 18 0c 1e 18 1c 18 1a >3e20 3f 85 e1 a9 00 85 df 85 >3f38 20 12 20 09 20 13 20 12
>3d10 0c 1e 0c 21 0c 24 18 00 >3e28 e2 a9 38 8d 11 ff a5 df >3f40 20 00 60 00 00 ea ea ea
>3d18 02 24 2e 25 18 26 3c 17 >3e30 d0 33 a0 01 b1 dd d0 06 >3f48 a9 0c 85 d9 a9 28 85 d8
>3d20 0c 18 0c 19 0c 1a 18 1f >3e38 a9 00 8d 11 ff 60 a0 00 >3f50 a0 00 a9 20 91 d8 c8 d0
>3d28 18 1e 18 1c 18 1a 0c 1f >3e40 b1 dd 0a aa bd ea 34 8d >3f58 fb e6 d9 a5 d9 c9 0f d0
>3d30 18 1e 18 1c 0c 1a 0c 17 >3e48 0e ff ad 12 ff 29 fc 1d >3f60 f1 a9 20 91 d8 c8 c0 c0
>3d38 0c 23 18 00 02 23 2e 21 >3e50 e9 34 8d 12 ff c8 b1 dd >3f68 d0 f9 60 a9 0c 85 d9 a9
>3d40 18 1f 48 1c 18 21 18 00 >3e58 85 df a5 dd 18 69 02 90 >3f70 00 85 d8 c0 00 f0 0e a5
>3d48 02 21 16 22 18 00 02 22 >3e60 02 e6 de 85 dd c6 df a5 >3f78 d8 18 69 28 90 02 e6 d9
>3d50 16 23 0c 00 02 23 16 26 >3e68 e2 d0 27 a0 00 b1 e0 0a >3f80 85 d8 88 d0 f2 e0 00 f0
>3d58 0c 24 18 23 18 00 02 23 >3e70 aa bd ea 34 8d 0f ff ad >3f88 09 e6 d8 d0 02 e6 d9 ca
>3d60 0a 00 02 23 16 1f 0c 21 >3e78 10 ff 29 fc 1d e9 34 8d >3f90 d0 f7 60 a5 d9 18 69 04
>3d68 0c 1f 0c 21 18 1f 48 00 >3e80 10 ff c8 b1 e0 85 e2 a5 >3f98 85 d9 60 a5 d9 38 e9 04
>3d70 18 00 00 07 0c 09 0c 0b >3e88 e0 18 69 02 90 02 e6 e1 >3fa0 85 d9 60 a9 48 8d 11 ff
>3d78 0c 09 24 02 18 09 0c 0b >3e90 85 e0 c6 e2 a2 00 a0 00 >3fa8 a9 00 8d 0f ff ad 10 ff
>3d80 0c 0d 0c 0b 24 02 18 04 >3e98 c8 d0 fd e8 e0 06 d0 f6 >3fb0 29 fc 09 01 8d 10 ff a9
>3d88 0c 00 02 04 16 03 0c 04 >3ea0 4c 2e 3e 1e 10 23 10 21 >3fb8 28 85 d8 a9 0c 85 d9 a0
>3d90 0c 06 0c 07 18 02 18 00 >3ea8 10 1f 10 1e 10 1c 10 1a >3fc0 00 a9 20 91 d8 c8 c0 28
>3d98 02 02 16 00 30 07 18 0e >3eb0 10 1e 10 1c 10 1a 10 17 >3fc8 d0 f9 a2 00 a0 00 c8 d0
>3da0 18 02 18 0e 18 07 18 0e >3eb8 10 15 20 26 20 23 10 21 >3fd0 fd e8 e0 25 d0 f6 a5 d8
>3da8 18 02 18 0e 18 07 18 0e >3ec0 10 23 10 21 10 1e 10 1a >3fd8 18 69 28 90 02 e6 d9 85
>3db0 18 02 18 0e 18 09 18 0e >3ec8 10 1c 10 1e 10 1c 70 15 >3fe0 d8 c9 e8 d0 da a9 00 8d
>3db8 18 02 18 0e 18 09 18 0e >3ecd0 20 1e 10 1d 10 1e 10 1f >3fe8 11 ff 60 ea ea ea ea ea
>3dc0 18 02 18 0e 18 09 18 0e >3ed8 10 21 10 20 10 21 10 22 >3ff0 00 00 00 00 00 00 00 00
>3dc8 18 02 18 0e 18 09 18 12 >3eed 10 23 10 22 10 23 10 26 >3ff8 00 00 00 00 00 00 00 00

```

Galaxis C 16/116

Vier Raumschiffe sind im Weltall vermisst. Die Bodenkontrollstation Galaxis startet die Suchaktion. Nur durch Konzentration, genaues Registrieren eingefangener Funksignale und durch logische Schlussfolgerungen kann die Suchaktion erfolgreich verlaufen. Auf Ihren Raumsektor-Bildschirm ist der Weltraumbereich sichtbar, in dem sich die vier vermissten Raumschiffe befinden müssen. Jeder Punkt zeigt die Position einer fest verankerten Wel-

Traumboje an und kann zugleich die Position eines der gesuchten Raumschiffe sein. Alle Linien von einem Punkt aus sind die Suchstrahlen dieser Raumboje. Bei Abruf melden die Raumbojen per Funksignal die Anzahl der Suchkontakte zu den RAumschiffen. Bei genauer Auswertung der Funksignale können die Raumschiffe geortet werden. Aber sie Zeit ist begrenzt und die Energie der Bodenkontrollstation reicht nur für 18 Kontaktaufnahmen

mit den Raumbojen aus. Deshalb muß die Suche gezielt und schnell erfolgen, damit eine Hilfsaktion für die Raumschiffbesatzung möglich ist.

Ladeanweisung:

Nachdem Sie das Programm eingetippt haben, speichern Sie es zuerst, da das Programm nach Starten mit RUN überschrieben wird. Am Ende jeder DATA-Zeile steht zum Erkennen von

Tippfehlern eine Prüfsumme (zusätzlich zum Tronic-Checksummer). Eventuelle Prüfsummenfehler werden mit der Zeilennummer ausgegeben. Treten Fehler auf, muß das eingetippte Programm neu eingeladen werden.

Wenn Sie das fehlerfreie Programm starten, wird ein Maschinenprogramm erzeugt, welches nach ca. 2 Minuten auf den ge-

wünschten Massenspeicher (Zeile 2130 beachten) abgesaved wird. Dieses Maschinenprogramm kann nun jederzeit mit LOAD (DLOAD) geladen und gestartet werden.

Steuerung

Zur Steuerung benötigen Sie ein Joystick in Port 2. Nach Pro-

grammstart können Sie durch Drücken des Joysticks nach oben den Schwierigkeitsgrad einstellen. Nach Betätigen der Feuertaste beginnt das Spiel. Die einzelnen Raumbojen werden mit dem Joystick angesteuert. Die gerade angepeilte Raumboje macht sich durch Blinken bemerkbar. Durch Drücken der Feuertaste wird die

Boje aktiviert und gibt die Anzahl der Sichtkontakte auf dem Bildschirm aus. Punkte, auf denen sich keine Raumschiffe befinden, können sie mit der Stern-Taste markieren. Aufgepaßt: Im Funk-schatten eines gefundenen Raumschiffes kann sich ein weiteres Raumschiff befinden. Haben Sie alle Raumschiffe ge-

funden, erscheint auf dem Counter-Monitor Ihre Punktzahl. Je kürzer die Suchzeit und je weniger Raumbojen sie aktiviert haben, desto größer wird Ihre Punktzahl sein. Haben Sie nicht alle Raumschiffe gefunden, können Sie sich mit der HELP-Taste die Lösung zeigen lassen. Ein neues Spiel beginnt mit RETURN.

C-16 LISTING + CHECKSUMMEN (OC V1.0)

```
1 REM *****
2 REM * GALAXIS *
3 REM *-----*
4 REM * FOR YOUR COMMODORE 16/116/PLUS4 *
5 REM *-----*
6 REM * 1986 BY JDACHIM STOLTE *
7 REM *****
8
9 :
1001 DATA 000B1001009E34313132000000000000A99320
D2FFA909BD15FFBD19FF20FB16 9DB <59>
1002 DATA A93CB13FFA9C08D12FF20F01320001AA200A9
779D00089D00097D000AA9569D C38 <230>
1003 DATA F80ACAD0EDA97358527A97CB531A909B532A900
B541B52F20FB10EAA9028575A2 ED3 <154>
1004 DATA 03A9099560CA10FBA909856020F01A20DB16A2
03A02E9B9D6B0F8BA97F9D6B0B D33 <235>
1005 DATA CA10F3EAEAEAEAEAA9778516200019A571C93C
F006A93CB5711004A9BCB571A4 052 <13>
1006 DATA 71A2029B9D1D0FA97A9D1D0B8BCA10F3207013
20201920701420E4FFC92A00CC CE5 <93>
1007 DATA A5318533A5321869048534A000A92F91334CBC
10EAA900856BA203B560D002E6 CRC <6>
1008 DATA 6CA10F7A569C904D0034C7D19A565C900D003
4C7D1960EAA900852DA90C852E D63 <190>
1009 DATA A900852B8510B511A90CB52CA200A000BD0020
912B8C8C004D0F518A52B692B C77 <77>
1010 DATA 852B9002E62CE00CD0E4E610A510C906F00B3B
A72B8E924B002C62CD0CFA18A5 F6C <204>
1011 DATA 2D6904852D9002E62EA52DB52BA52E852CA900
B510E611A511C909D0AFAEA200 CC0 <189>
1012 DATA A000B90C209D000CB910209DD00EE8C8C004D0
EEE024D0E8EAA900852BA90CB5 EFD <188>
1013 DATA 2CA200A000B971420912B18A52B6927B52B9002
E62CC8C003D0EB18A52B6903B5 C02 <168>
1014 DATA 2B9002E62CEBE006D09EAA924852BA90CB52C
A200A000B91720912B18A52B69 D52 <179>
1015 DATA 27852B9002E62CC8C003D0EB18A52B6903B52B
9002E62CEBE006D09EAA948BD F08 <25>
1016 DATA F40EAEAEAEAEAEAEAEAEAEAEAEAEAEAEAEAEAE
A000A200A92599FB0E9A92679C3 669 <203>
1017 DATA 0FC8E8E005D0F9EAC8C8C8C002030E6EAEAEAE
A000A200A92799220F994A0F99 1AB <162>
1018 DATA 720F999A0FA972992B0F99500F997B0F99A00F
EAC8C8C8C8C8C8C8C8E004D0 08A <100>
1019 DATA D6EAA200A9219DFA0EA9229D000FA9249DC20F
A9239DC80FEBE8E8E8E8E8E8E8 203 <142>
1020 DATA E02030E060EAEAEAEAEAEAEAEAEAEAEAA000B9
1A2099230FB971F20994B0FB924 1A4 <97>
1021 DATA 2099730FB972920999B0FCBC005D0E3A000B92E
2099230BB93320994B0BB93820 C3A <37>
1022 DATA 99730BB93D20999B0B8C005D0E360EAA000B9
4220992B0FB972099530FB94C BA4 <58>
1023 DATA 20997B0FB97512099A30FCBC005D0E3A000B956
20992B0BB975B2099530BB96Q20 CFA <136>
1024 DATA 997B0BB9652099A30B8C005D0E360EAA000B9
6A2099730FB976F20995B0FB974 E64 <240>
1025 DATA 2099B30FB9792099A90FCBC005D0E3A000B97E
2099730BB98320995B0BB98820 DBA <125>
1026 DATA 99830BB98BD2099A90B8C005D0E360EAA000B9
922099730FB9772099630FB99C F24 <0>
1027 DATA 20998B0FB97A12099B30FCBC005D0E3A000B9A6
2099730BB98B2099630BB9B020 E7A <56>
1028 DATA 998B0BB97852099B30B8C005D0E360EAA90124
2FD00BA923852BA90FB52C20BB E33 <244>
1029 DATA 13A902242FD00BA928852BA90FB52C20BB13A9
04242FD00BA9733852BA90FB52C A94 <212>
1030 DATA 20BB13A90B242FD00BA9733852BA90FB52C20BB
13A0EAEAEAA20BAD00FF6A9D47 D74 <199>
1031 DATA 3DCAD00F6A52B852DA90B852EA200A000A92991
2BA971912DC8C005D0F318A52B E96 <220>
```

```
1032 DATA 6928852B852DEBE004D0E360EAEAEAEAA00FF
A20495D0CA10FBA90085102039 09D <139>
1033 DATA 14EAEAC90AB0F7B52B203914EAEAC907B0F70A
0A0A0A18652B852BA510F00CA6 DA1 <63>
1034 DATA 10CAB520C52BF0D6CA10F7A52BA6109520E610
E003D0C760EAA9E025D4092085 05B <164>
1035 DATA D418A204B5D075D095D5CA10F718A204B5D075
D595D0CA10F718A202B5D075D7 1E7 <80>
1036 DATA 95D0CA10F7A5D060EAEAEAEAEAEAEAA9FD8D
0BFFAD0BFFA000A2004AB0018B 40E <88>
1037 DATA 4A0001C84A8001CA4A8001E8B6D0B4D12908D0
034C1E158572EAA000A5169131 D97 <59>
1038 DATA A5D0F02D3018A527290FC909F023E627A5311B
690485319002E632CD114A527 C88 <220>
1039 DATA 290FF00DC627A5313BE9048531B002C632A5D1
F03B3021A52729F04A4A4A4C97 D45 <120>
1040 DATA 06F02DA527186910B527A53118697885319002
E6324C1015A52729F0F012A527 B7F <65>
1041 DATA 38E910B527A53138E9788531B002C632200019
4CE015EAEAEAEAEAEAEAEAEAEAE 06D <244>
1042 DATA A527C520D009A52F0901852F4C3C16C521D009
A52F0902B52F4C4216C522D009 A70 <225>
1043 DATA A52F0904852F4C4816C523D009A52F0908852F
4C4E1620D015EAEAA5318533A5 B55 <63>
1044 DATA 32B534A9008514EAEAEAEAEAA527B52B207013
A90185D0A90085D1209016A9FF F47 <226>
1045 DATA B5D120B516A90085D0209016A9FFB5D020B516
A90085D1209016A901A901B5D1 E5C <240>
1046 DATA 20B516A90085D0209016A901B5D020B5162050
1AA52B8527A5338531A5348532 BBA <64>
1047 DATA A000A5341869048534A51469B0913360A527C5
6AF004856AC66520201A60EAEAE 444 <29>
1048 DATA A572F00EA000A51609B09131200B1B200B1B60
EAEAEAEAEAEAEAEAEAEAEAEAEAE 183 <134>
1049 DATA 207013A5318533A5328536A5168515A527B519
20A014206716EAEAEAEAEAEAEAE 60 DD1 <239>
1050 DATA EAEAEAEAEAEAEAEAEAEAEAEAEAEAEAEAEAEAE
14E617CA10F360EAEAEAEAEAEAE 0E1 <29>
1051 DATA 511620B0124C511620F0124C5116203013A531
B533A5321869048534A000A92A 94A <67>
1052 DATA 91334C801AEAEAA9008518A519290F8510A527
290FC510F002E618A51929F085 C74 <204>
1053 DATA 10A52729F0C510F002E61860EAEAEAEAA53385
31A5348532A528527200016A5 E34 <127>
1054 DATA 18F00A202616A517D0034C9C16EAEAEAEAE60EA
EAEAA5288527A5338531A53485 F26 <195>
1055 DATA 32200016A518C902300A202616A517D0034CC1
164CA016EAA205BDE01F9D420F AB2 <25>
1056 DATA BDE81F9D6A0F9D209DBA0FBDF01F9DE20FCA
10E5A912856560EAA9018581EA 020 <164>
1057 DATA A993A200A96E9D00089D0009EAA9779D500ACA
D0EFA900852485238520852285 D95 <110>
1058 DATA D085118513A9D085128514A9EF85168518A90C
B517B519AD07FF29F0BD07FFAD E72 <239>
1059 DATA 06FF29F0BD06FF208B18A620BDC020D006E622
A920C620B523A202A5240A8524 D5B <108>
1060 DATA 18A5230A85239002E624CA10EEA52385118530
A52418651285128531A9008524 B00 <218>
1061 DATA 207021EAEAA9808515A907B521A000B1112515
C515D006A9A091163005A92091 C69 <219>
1062 DATA 16EAEAEAEAA51118690185119002E612A51618
692885169002E617EAC621A521 D3B <164>
1063 DATA 10C9A9078510AD07FF29F0180510BD07FFA216
A0168BD0FDCAD0F8C610A510D0 F5A <153>
1064 DATA E5A200A5DBF0FCB9C9079CB0CB9F10C99F00C
B9190D99180DB9410D99400B89 F7C <64>
1065 DATA 690D996B0DB9910D99900DB9B90D9980DB9E1
0D99E00DC8C027D0C7EAA53085 FA5 <235>
1066 DATA 11A5318512A5188516A5198517EAEAEAA5154A
B515F0034CB917EAA5138511A5 448 <155>
1067 DATA 148512E620A522C905F0034CA17A5D0D0034C
1517A908BD07FFA918BD06FFA9 CE9 <202>
```


C-16

SPRITES —

Jetzt auch für den Textbildschirm

Das vorliegende Programm erzeugt spriteähnliche Shapes, die auch im Textmodus funktionieren. Diese definierten Grafiken lassen sich Punkt für Punkt auf dem Textbildschirm fortbewegen.

Diese Grafiken sind frei definierbar. Sie müssen in den dafür vorgesehenen Spritespeicher gepoket

werden. Einige Grundformen sind bereits vorhanden. Die Routine ist gedacht, um in Spielen die Figuren genauer bewegen zu können als in dem groben 8x8-Raster der Zeichen.

Allerdings gehen durch die Anwendung der Sprites die oberen 32 Zeichen der Poke-Code-

Tabelle verloren (nicht die Reverse-Codes)!

Auf 1000 Positionen des Bildschirms lassen sich die "Sprites" postieren.

Ein weiterer Vorteil: Sie können sowohl den Großschrift- als auch den Kleinschriftmodus verwenden.

Dazu ist nur eine kleine Änderung notwendig, die im Programm erläutert wird.

Insgesamt stehen 6 verschiedene Sprites zu Verfügung (0-5).

Alles weitere wird im Programm selbst erläutert. Bitte beachten Sie die sehr lange Ladezeit!

C-16 LISTING + CHECKSUMMEN (DC V1.0)

```

10 REM *****
20 REM * COMMODORE C 16 *
30 REM *
40 REM * SPRITES #1 *
50 REM *
60 REM *21.04.86/19.05.86*
70 REM *****
80 REM WRITTEN 1986 BY NIKI LOCHNER
90 SCNCLR
100 REM ---SPEICHER BEGRENZEN---
110 AD=DEC("3547")
120 HB=INT(AD/256)
130 LB=AD-256*HB
140 POKE 51, LB:POKE 52, HB
150 POKE 55, LB:POKE 56, HB
160 CLR
170 REM =====
180 REM = DATA'S FUER ZSV =
190 REM =====
200 DATA A9,00 :REM LDA #00 ;
210 DATA B5,D0 :REM STA #D0 ; INITIALISIE
REN DER
220 DATA B5,D2 :REM STA #D2 ;
230 DATA A9,D0 :REM LDA #D0 ; VEKTOREN
240 DATA B5,D1 :REM STA #D1 ;
250 DATA A9,38 :REM LDA #38 ; D0-D1: D000
260 DATA B5,D3 :REM STA #D3 ; D1-D2: 3800
270 DATA A2,08 :REM LDX #08 ; X UND
280 DATA A0,00 :REM LDY #00 ; Y BELEGEN
290 DATA B1,D0 :REM LDA (#D0),Y ; UMLEGEN
300 DATA 91,D2 :REM STA (#D2),Y
310 DATA 88 :REM DEY ;
320 DATA D0,F9 :REM BNE #0712 ;
330 DATA E6,D1 :REM INC #D1 ; D1 UND
340 DATA E6,D3 :REM INC #D3 ; D3 UM 1 ERH
UEHEN
350 DATA CA :REM DEX ;
360 DATA D0,F2 :REM BNE #0712 ;
370 DATA 60 :REM RTS ; ZURUECK IN'
S BASIC
380 DATA EA :REM NOP ;
390 REM +++EINLESEN+++
400 FOR I=1792 TO 1825
410 READ A#
420 A=DEC(A#)
430 PR=PR+A
440 POKE I,A
450 NEXT I
460 IF PR<>5666 THEN PRINT"?DATA{SPACE}ERROR{SPA
CE}IN{SPACE}430-610":END:ELSE PR=0

```

Teil 1

```

470 SYS 1792
480 REM ---ZUSATZPOKE'S FUER ZSV---
490 POKE 65298,PEEK(65298) AND 251
500 POKE 65299,PEEK(65299) AND 3 OR 56
510 POKE 740,56
520 REM =====
530 REM = DATA'S UNTERPROGRAMME =
540 REM =====
550 DATA EA :REM NOP ; UMLEGEN DER
ADRESSE
560 DATA A5,DE :REM LDA #DE ;
570 DATA 18 :REM CLC ; IN #E2-E3 F
UER LSV, RSV
580 DATA 69,01 :REM ADC #01 ;
590 DATA B0,05 :REM BCS #3554 ;
600 DATA B5,DE :REM STA #DE ;
610 DATA A0,09 :REM LDY #09 ;
620 DATA 60 :REM RTS ;
630 DATA E6,DF :REM INC #DF ;
640 DATA A9,00 :REM LDA #00 ;
650 DATA 4C,4F,35 :REM JMP #354F ;
660 DATA EA :REM NOP ;
670 DATA 20,47,35 :REM JSR #3547 ; LINKS-VERSC
HIEBUNGS-
680 DATA 88 :REM DEY ;
690 DATA 30,67 :REM BMI #35C9 ; ROUTINE (LS
V)
700 DATA A9,08 :REM LDA #08 ;
710 DATA 38 :REM SEC ;
720 DATA E5,DC :REM SBC #DC ;
730 DATA AA :REM TAX ;
740 DATA B9,F7,03 :REM LDA #03F7,Y ;
750 DATA 0A :REM ASL ;
760 DATA CA :REM DEX ;
770 DATA D0,FC :REM BNE #356B ;
780 DATA 91,DE :REM STA (#DE),Y ;
790 DATA 4C,5F,35 :REM JMP #355F ;
800 DATA EA :REM NOP ;
810 DATA 20,47,35 :REM JSR #3547 ; RECHTS-VERS
CHIEBUNGS-
820 DATA 88 :REM DEY ;
830 DATA 30,4E :REM BMI #35C9 ; ROUTINE (RS
V)
840 DATA B9,F7,03 :REM LDA #03F7,Y ;
850 DATA A6,DC :REM LDX #DC ;
860 DATA 4A :REM LSR ;
870 DATA CA :REM DEX ;
880 DATA D0,FC :REM BNE #3580 ;
890 DATA 91,DE :REM STA (#DE),Y ;
900 DATA 4C,7B,35 :REM JMP #357B ;
910 DATA EA :REM NOP ;
920 DATA A9,0B :REM LDA #0B ; UNTEN-VERSC

```

programme

HIEBUNGS-			<50>	1710 DATA 20,84,9D	:REM JSR \$9DB4		<183>
930 DATA 3B	:REM SEC		<233>	1720 DATA 4C,0E,36	:REM JMP \$360E		<137>
940 DATA ES,DD	:REM SBC \$DD	; ROUTINE (UN		1730 DATA EA	:REM NOP	; DEFINITION	
V)			<144>	WENN KX=0			<128>
950 DATA AA	:REM TAX		<198>	1740 DATA A0,0B	:REM LDY #0B		<156>
960 DATA A0,0B	:REM LDY #0B		<141>	1750 DATA B1,E4	:REM LDA (#E4),Y	(KX=0 D)	<41>
970 DATA B1,DE	:REM LDA (#DE),Y		<204>	1760 DATA 91,E2	:REM STA (#E2),Y		<54>
980 DATA CB	:REM INY		<238>	1770 DATA 8B	:REM DEY		<148>
990 DATA 91,DE	:REM STA (#DE),Y		<173>	1780 DATA D0,F9	:REM BNE \$361F		<193>
1000 DATA 8B	:REM DEY		<143>	1790 DATA 60	:REM RTS		<25>
1010 DATA 8B	:REM DEY		<153>	1800 DATA EA	:REM NOP		<34>
1020 DATA 30,03	:REM BMI \$359E		<142>	1810 DATA A5,E2	:REM LDA #E2		<129>
1030 DATA 4C,92,35	:REM JMP \$3592		<175>	1820 DATA A2,0B	:REM LDX #0B		<232>
1040 DATA CB	:REM INY		<43>	1830 DATA 18	:REM CLC		<62>
1050 DATA A9,00	:REM LDA #00		<83>	1840 DATA 69,01	:REM ADC #01		<30>
1060 DATA 91,DE	:REM STA (#DE),Y		<244>	1850 DATA B0,0A	:REM BCS \$363B		<20B>
1070 DATA CA	:REM DEX		<255>	1860 DATA CA	:REM DEX		<24>
1080 DATA F0,23	:REM BEQ \$35C9		<217>	1870 DATA D0,FB	:REM BNE \$362C		<245>
1090 DATA 4C,90,35	:REM JMP \$3590		<185>	1880 DATA 85,E2	:REM STA #E2		<148>
1100 DATA EA	:REM NOP		<99>	1890 DATA 20,1D,36	:REM JSR \$361D		<95>
1110 DATA A5,DD	:REM LDA \$DD	; OBEN-VERSC	<61>	1900 DATA 60	:REM RTS		<136>
HIEBUNGS-			<75>	1910 DATA EA	:REM NOP		<144>
1120 DATA C9,00	:REM CMP #00		<4>	1920 DATA E6,E3	:REM INC #E3		<109>
1130 DATA F0,19	:REM BEQ \$35C9	; ROUTINE (D	<54>	1930 DATA 4C,31,36	:REM JMP \$3631		<173>
BV)			<164>	1940 DATA EA	:REM NOP		<174>
1140 DATA A6,DD	:REM LDX \$DD		<139>	1950 DATA A0,0B	:REM LDY #0B	; DEFINITION	<23>
1150 DATA A0,00	:REM LDY #00		<58>	KX=0 U. KY=0			<228>
1160 DATA B1,DE	:REM LDA (#DE),Y		<108>	1960 DATA A9,00	:REM LDA #00		<192>
1170 DATA 8B	:REM DEY		<193>	1970 DATA 97,09,04	:REM STA \$0409,Y	(KX=0, KY=	<11>
1180 DATA 91,DE	:REM STA (#DE),Y		<203>	0 D)			<113>
1190 DATA CB	:REM INY		<200>	1980 DATA 99,1C,04	:REM STA \$041C,Y		<131>
1200 DATA EB	:REM INY		<72>	1990 DATA 8B	:REM DEY		<171>
1210 DATA C0,09	:REM CPY #09		<118>	2000 DATA D0,F5	:REM BNE \$3643		<165>
1220 DATA D0,F5	:REM BNE \$35B4		<17>	2010 DATA A0,0B	:REM LDY #0B		<75>
1230 DATA 8B	:REM DEY		<178>	2020 DATA B9,F6,03	:REM LDA \$03F6,Y		<158>
1240 DATA A9,00	:REM LDA #00		<189>	2030 DATA 99,00,04	:REM STA \$0400,Y		<174>
1250 DATA 91,DE	:REM STA (#DE),Y		<163>	2040 DATA 99,12,04	:REM STA \$0412,Y		<153>
1260 DATA CA	:REM DEX		<26>	2050 DATA 8B	:REM DEY		<70>
1270 DATA D0,EB	:REM BNE \$35B2		<34>	2060 DATA D0,F4	:REM BNE \$3650		<205>
1280 DATA 60	:REM RTS		<46>	2070 DATA E6,DD	:REM INC \$DD		<209>
1290 DATA EA	:REM NOP		<0>	2080 DATA A9,FF	:REM LDA #FF		<176>
1300 DATA 60	:REM RTS		<120>	2090 DATA 85,DE	:REM STA \$DE		<249>
1310 DATA EA	:REM NOP	; 2-BYTE MUL	<106>	2100 DATA A9,03	:REM LDA #03		<130>
TIPLIKATIONS-			<180>	2110 DATA 85,DF	:REM STA \$DF		<225>
1320 DATA 85,D4	:REM STA \$D4		<170>	2120 DATA 20,8A,35	:REM JSR \$358A		<3>
1330 DATA A9,00	:REM LDA #00	; ROUTINE (2	<102>	2130 DATA A9,13	:REM LDA #13		<247>
-BM)			<190>	2140 DATA 85,DE	:REM STA \$DE		<43>
1340 DATA 85,D6	:REM STA \$D6		<132>	2150 DATA A9,04	:REM LDA #04		<131>
1350 DATA 85,D5	:REM STA \$D5		<160>	2160 DATA 85,DF	:REM STA \$DF		<157>
1360 DATA 1B	:REM CLC		<74>	2170 DATA 20,AA,35	:REM JSR \$35AA		<169>
1370 DATA 85,D5	:REM STA \$D5		<27>	2180 DATA 4C,3D,37	:REM JMP \$373D		<106>
1380 DATA 65,D4	:REM ADC \$D4		<240>	2190 DATA EA	:REM NOP		<100>
1390 DATA B0,06	:REM BCS \$35E0		<176>	2200 DATA A0,0B	:REM LDY #0B		<10>
1400 DATA CA	:REM DEX		<181>	2210 DATA B9,F6,03	:REM LDA \$03F6,Y		<98>
1410 DATA D0,F6	:REM BNE \$35D3		<65>	2220 DATA 99,00,04	:REM STA \$0400,Y		<141>
1420 DATA 85,D5	:REM STA \$D5		<42>	2230 DATA 8B	:REM DEY		<108>
1430 DATA 60	:REM RTS		<103>	2240 DATA D0,F7	:REM BNE \$367A		<17>
1440 DATA E6,D6	:REM INC \$D6		<200>	2250 DATA A0,1A	:REM LDY #1A		<29>
1450 DATA A0,FF	:REM LDY #FF		<240>	2260 DATA A9,00	:REM LDA #00		<148>
1460 DATA C6,D5	:REM DEC \$D5		<244>	2270 DATA 99,09,04	:REM STA \$0409,Y		<20>
1470 DATA 8B	:REM DEY		<0>	2280 DATA 8B	:REM DEY		<21>
1480 DATA D0,FB	:REM BNE \$35E4		<70>	2290 DATA D0,FB	:REM BNE \$3685		
1490 DATA 4C,DA,35	:REM JMP \$35DA		<222>	2300 DATA 4C,3D,37	:REM JMP \$373D		
1500 DATA EA	:REM NOP		<109>	2310 DATA EA,EA,F9,D8,E7,FB,FB,EA	:REM "(S)2 SY		
1510 DATA A9,0D	:REM LDA #0D	; SPRITE ERR	<101>	X(S6 SX2 S2)":REM ADC-MEMORYS (ADM)			<73>
OR-			<49>	2320 DATA 3F,53,50,52,49,54,45,20	:REM "?SPRITE{		
1520 DATA 20,D2,FF	:REM JSR \$FFD2		<15>	SPACE)":REM SPRITE ERROR			<147>
1530 DATA A2,00	:REM LDX #00	; ROUTINE (S	<238>	2330 DATA 45,52,52,4F,52,20,23,00	:REM "ERROR{SP		
ER)			<246>	ACE)":REM MEMORYS (SEM)			<50>
1540 DATA 8D,9B,36	:REM LDA \$369B,X		<87>	2340 DATA EA	:REM NOP		<64>
1550 DATA 20,D2,FF	:REM JSR \$FFD2		<91>	2350 DATA EA	:REM NOP		<74>
1560 DATA EB	:REM INX		<171>	2360 DATA EA	:REM NOP		<84>
1570 DATA C9,00	:REM CMP #00		<23>	2370 DATA EA	:REM NOP		<94>
1580 DATA D0,F5	:REM BNE \$35F4		<178>	2380 DATA 4C,ED,35	:REM JMP \$35ED	; SPRUNGVEKT	
1590 DATA A5,D7	:REM LDA \$D7		<123>	OR SER			<130>
1600 DATA 1B	:REM CLC		<9>	2390 DATA EA	:REM NOP		<114>
1610 DATA 69,30	:REM ADC #30		<37>	2400 REM ---EINLESEN UNTERPROGRAMME---			<198>
1620 DATA 20,D2,FF	:REM JSR \$FFD2		<143>	2410 FOR I=13639 TO 13999			<94>
1630 DATA A9,06	:REM LDA #06		<181>	2420 READ A#			<242>
1640 DATA 3B	:REM SEC		<217>	2430 A=DEC(A#)			<123>
1650 DATA E5,D7	:REM SBC \$D7		<18>	2440 PR=PR+A			<56>
1660 DATA 85,D7	:REM STA \$D7		<143>	2450 POKE I,A			<76>
1670 DATA C6,D7	:REM DEC \$D7		<181>	2460 NEXT I			<185>
1680 DATA D0,01	:REM BNE \$3613		<217>	2470 IF PR<>49080 THEN PRINT"DATA{SPACE}ERROR{SP			<53>
1690 DATA 60	:REM RTS			ACE}IN{SPACE}500-2300":END:ELSE PR=0			<221>
1700 DATA 20,91,94	:REM JSR \$9491			2480 REM ---EINLESEN DER GRUNDSPRITES---			

```

2490 DATA 18,18,FF,3C,3C,3C,42,B1 <206>
2500 DATA 00,00,00,3E,44,44,BB,00 <243>
2510 DATA 3E,49,49,49,3E,1C,3E,00 <87>
2520 DATA 3E,49,49,49,3E,41,7F,00 <181>
2530 DATA 18,24,5A,5A,5A,42,3C,00 <138>
2540 DATA 10,28,28,30,24,64,9B,00 <5>
2550 DATA 00,00,18,3C,3C,18,00,00 <99>
2580 FOR I=14288 TO 14335 <78>
2590 READ A# <157>
2600 A=DEC(A#) <38>
2610 PR=PR+A <227>
2620 POKE I,A <247>
2630 NEXT I <100>
2640 IF PR<>2922 THEN PRINT"DATA(SPACE)ERROR(SP
ACE)IN(SPACE)2590-2570":END:ELSE PR=0 <84>
2650 SCNCLR <76>
2660 PRINT"EINLESEN(SPACE)DES(SPACE)2.(SPACE)PRG
GRAMMTEILS(SPACE)FLASHON?(FLASHOFF SPACE)J/N" <179>
2670 GET T# <45>
2680 IF T#="N" THEN END <13>
2690 IF T#="J" THEN 2710 <194>
2700 GOTO 2670 <241>
2710 REM ---EINLESEN DES 2. PRG-TEILS--- <184>
2720 SCNCLR <146>
2730 PRINT"(HOME)NEW(DOWN3 LEFT3)LOAD"CHR$(34)"B
PRITES(SPACE)#2"CHR$(34);",1":REM BEI DISKETTE .
...8 <133>
2740 PRINT"(DOWN5)RUN(HOME)" <70>
2750 FOR I=1319 TO 1321:POKE I,13:NEXT I <55>
2760 POKE 239,3 <88>
2770 END <92>
10 REM ***** <227>
20 REM * COMMODORE C 16 * <145>
30 REM * * <127>
40 REM * SPRITES #2 * <111>
50 REM * * <147>
60 REM *21.04.86/19.05.86* <145>
70 REM ***** <31>
80 REM WRITTEN 1986 BY NIKI LOCHNER <235>
90 SCNCLR <66>
100 REM ===== <93>
110 REM = DATA'S FUER HAUPTROUTINE = <84>
120 REM ===== <113>
130 DATA A9,00 :REM LDA #00 ; EINLESEN DE
R PARAMETER <254>
140 DATA B5,D7 :REM STA $D7 ; <19>
150 DATA EA :REM NOP ; (ELS) <53>
160 DATA 20,91,94 :REM JSR #9491 ; <207>
170 DATA 20,84,9D :REM JSR #9D84 ; <173>
180 DATA 8A :REM TXA ; <152>
190 DATA A6,D7 :REM LDX #D7 ; <120>
200 DATA 95,D9 :REM STA #D9,X ; <3>
210 DATA E6,D7 :REM INC #D7 ; <246>
220 DATA 1B :REM CLC ; <237>
230 DATA 7D,92,36 :REM ADC #3692,X ; <53>
240 DATA B0,E3 :REM BCS #36AB ; <102>
250 DATA A5,D7 :REM LDA #D7 ; <180>
260 DATA C9,05 :REM CMP #05 ; <85>
270 DATA D0,E7 :REM BNE #36B5 ; <180>
280 DATA EA :REM NOP ; <44>
290 DATA A9,07 :REM LDA #07 ; <235>
300 DATA 3B :REM SEC ; <113>
310 DATA E5,DD :REM SBC #DD ; <62>
320 DATA B5,DD :REM STA #DD ; <204>
330 DATA E6,D9 :REM INC #D9 ; <151>
340 DATA EA :REM NOP ; <104>
350 DATA A5,D9 :REM LDA #D9 ; EINLESEN DE
S GRUND- <78>
360 DATA A2,0B :REM LDX #0B ; <46>
370 DATA 20,CB,35 :REM JSR #35CB ; SPRITES IN
DEN <124>
380 DATA 1B :REM CLC ; <142>
390 DATA 69,CB :REM ADC #CB ; RS-232 PUFF
ER NACH #3F7- <143>
400 DATA B5,E0 :REM STA #E0 ; <158>
410 DATA A9,37 :REM LDA #37 ; <156>
420 DATA B5,E1 :REM STA #E1 ; <198>
430 DATA A0,00 :REM LDY #00 ; <209>
440 DATA B1,E0 :REM LDA (#E0),Y ; <18>
450 DATA 99,F7,03 :REM STA #03F7,Y ; <168>
460 DATA CB :REM INY ; <228>
470 DATA C0,0B :REM CPY #0B ; <204>
480 DATA D0,F6 :REM BNE #36EB ; <130>
490 DATA EA :REM NOP ; <254>
500 DATA A5,DC :REM LDA #DC ; UEBERPRUEFE
N OB <150>

```

Teil 2

```

510 DATA C9,00 :REM CMP #00 ; <230>
520 DATA D0,0C :REM BNE #370B ; KX=0 ODER K
Y=7 <136>
530 DATA A5,DD :REM LDA #DD ; <210>
540 DATA C9,07 :REM CMP #07 ; <152>
550 DATA D0,03 :REM BNE #3705 ; <76>
560 DATA 4C,7B,36 :REM JMP #367B ; <89>
570 DATA 4C,41,36 :REM JMP #3641 ; <111>
580 DATA EA :REM NOP ; <89>
590 DATA A9,FF :REM LDA #FF ; ERMITTELN: <170>
600 DATA B5,DE :REM STA #DE ; <249>
610 DATA A9,03 :REM LDA #03 ; 1. QUADRANT
(LINKS OBEN) = I <159>
620 DATA B5,DF :REM STA #DF ; <33>
630 DATA 20,75,35 :REM JSR #3575 ; <103>
640 DATA 20,8A,35 :REM JSR #358A ; <180>
650 DATA EA :REM NOP ; <159>
660 DATA A9,0B :REM LDA #0B ; 2. QUADRANT
(RECHTS OBEN) = II <47>
670 DATA B5,DE :REM STA #DE ; <63>
680 DATA A9,04 :REM LDA #04 ; <51>
690 DATA B5,DF :REM STA #DF ; <103>
700 DATA 20,5C,35 :REM JSR #355C ; <221>
710 DATA 20,8A,35 :REM JSR #358A ; <250>
720 DATA EA :REM NOP ; <229>
730 DATA A9,11 :REM LDA #11 ; 3. QUADRANT
(LINKS UNTEN) = III <233>
740 DATA B5,DE :REM STA #DE ; <133>
750 DATA 20,75,35 :REM JSR #3575 ; <223>
760 DATA 20,AA,35 :REM JSR #35AA ; <251>
770 DATA EA :REM NOP ; <24>
780 DATA A9,1B :REM LDA #1B ; 4. QUADRANT
(RECHTS UNTEN) = IV <156>
790 DATA B5,DE :REM STA #DE ; <184>
800 DATA 20,5C,35 :REM JSR #355C ; <66>
810 DATA 20,AA,35 :REM JSR #35AA ; <46>
820 DATA EA :REM NOP ; <74>
830 DATA A5,D9 :REM LDA #D9 ; FESTSTELLEN
DER ADRESSE <41>
840 DATA A2,20 :REM LDX #20 ; <142>
850 DATA 20,CB,35 :REM JSR #35CB ; FUER DIE DE
FINITION <219>
860 DATA B5,E2 :REM STA #E2 ; <148>
870 DATA C6,E2 :REM DEC #E2 ; <176>
880 DATA A9,3F :REM LDA #3F ; <175>
890 DATA B5,E3 :REM STA #E3 ; <198>
900 DATA EA :REM NOP ; <154>
910 DATA A9,00 :REM LDA #00 ; DEFINIEREN
IN DIE <26>
920 DATA B5,E4 :REM STA #E4 ; <248>
930 DATA A9,04 :REM LDA #04 ; NEUE ASC-CO
DE TABELLE <249>
940 DATA B5,E5 :REM STA #E5 ; <32>
950 DATA 20,1D,36 :REM JSR #361D ; I <172>
960 DATA A9,09 :REM LDA #09 ; I <142>
970 DATA B5,E4 :REM STA #E4 ; <42>
980 DATA 20,2B,36 :REM JSR #362B ; II <251>
990 DATA A9,12 :REM LDA #12 ; <83>
1000 DATA B5,E4 :REM STA #E4 ; <72>
1010 DATA 20,2B,36 :REM JSR #362B ; III <168>
1020 DATA A9,1C :REM LDA #1C ; III <60>
1030 DATA B5,E4 :REM STA #E4 ; <103>
1040 DATA 20,2B,36 :REM JSR #362B ; IV <86>
1050 DATA EA :REM NOP ; <49>
1060 DATA A5,D9 :REM LDA #D9 ; FESTSTELLE
N DES ZEICHEN- <61>
1070 DATA A2,04 :REM LDX #04 ; <163>
1080 DATA 20,CB,35 :REM JSR #35CB ; CODES ZUM
SETZEN IN DEN <20>
1090 DATA 1B :REM CLC ; <87>
1100 DATA 69,60 :REM ADC #60 ; BILDSCHIRM
SPEICHER (=BSP) <7>
1110 DATA B5,EB :REM STA #EB ; <7>
1120 DATA EA :REM NOP ; <119>
1130 DATA A5,DB :REM LDA #DB ; FESTSTELLE
N DER ADRESSE <10>
1140 DATA A2,2B :REM LDX #2B ; <99>
1150 DATA 20,CB,35 :REM JSR #35CB ; IM BSP <235>
1160 DATA A5,D6 :REM LDA #D6 ; <50>
1170 DATA 1B :REM CLC ; <167>
1180 DATA 69,0C :REM ADC #0C ; <1>
1190 DATA B5,D6 :REM STA #D6 ; <29>
1200 DATA A6,DA :REM LDX #DA ; <54>
1210 DATA EB :REM INX ; <209>
1220 DATA A5,D5 :REM LDA #D5 ; <90>
1230 DATA 1B :REM CLC ; <227>

```


C64

Spannende Action und hervorragende Grafik

ATOM



Soeben zum Commander der neuen Sondereinheit "Sirius" ernannt, haben Sie nicht nur das Vergnügen in eine höhere Gehaltsstufe aufzurücken, sondern müssen auch mit Ihrer Crew zu dem fernen, noch wenig bekannten Planeten "Atom" aufbrechen. Ziel des Unternehmens ist es, das geheime Innenleben dieses künstlichen Sternes zu erkunden und mögliche Neuerungen auch Ihrer Zivilisation zugänglich zu machen. Als Sie schließlich mit Ihrem Raumer durch einen der vielen Schächte in den Untergrund abtauchen, verschließt sich plötzlich mit einem lauten Knall der Eingang. Das automatische Verteidigungssystem hat sich in Gang gesetzt.

Können Sie noch entkommen? Gelingt es Ihnen, die zahlreichen Gefahren eines Bildes zu meistern, so steuern Sie Ihren Raumgleiter durch den Ausgang nach oben, bis der nächste Screen erscheint....

Besonderheiten:
 sehr gut Farbgrafik + Soundeffekte
 Animationseffekte
 absolut ruckfreie Grafikbewegung
 durch Menü mehrere Levels wählbar

Steuerung: Joystick (Port 2): alle 8 Richtungen + Fire

C-64 LISTING + CHECKSUMMEN (C V1.0)

```

0 REM *****
1 REM *
2 REM * ATOM (C) 1986 *
3 REM * C-64 ACTION GAME *
4 REM * WRITTEN BY *
5 REM * CHRISTIAN WURZER *
6 REM *
7 REM *****
8 GOSUB411:GOTO135
9 REM ** START
10 FORT=0T02
11 IFPEEK(53246)=1THEN13
12 NEXT:RETURN
13 POKEV+30,0:POKEV+31,0
14 L=L-1:POKEV,57:POKEV+1,199
15 IFL=-1THEN20
16 POKE214,22:POKE211,35:SYS58640
17 PRINT "(RVSON BROHN)"STR*(L)
18 FORT=30T01STEP-1:POKE51206,T:SYS51200:NEXT
19 POKE53246,0:RETURN
20 REM ** GAME OVER
21 POKEV+21,0
22 GOSUB428:PRINT"(DOWN3 SPACES RED RVSON SPACE2
Z)"
23 PRINT"(SPACES RVSON RED SPACE BROHN SPACE25 R
ED SPACE)"
24 PRINT"(SPACES RVSON RED SPACE BROHN SPACE9)GA
ME(SPACES)OVER(SPACES RED SPACE)"
25 PRINT"(SPACES RVSON RED SPACE BROHN SPACE3)FI
RE(SPACES)=(SPACE)NEUES(SPACES)SPIEL(SPACES3 RED S
PACE)"
26 PRINT"(SPACES RVSON RED SPACE BROHN SPACE25 R
ED SPACE)"
27 PRINT"(SPACES RED RVSON SPACE27)":PRINTTAB(12
)"(RVSON)SCORE(SPACES):"SC
28 IFPEEK(56320)<>111THEN28
29 GOTO305
30 REM ** SCREEN 1
31 POKEV+17,PEEK(V+17)AND239
32 POKE50168,40:POKE50169,41:POKE50170,43:POKE50
171,44:POKE50172,44
33 POKE50173,41:POKE50174,46
34 POKEV+4,72:POKEV+5,110:POKEV+6,230:POKEV+7,12
7:POKEV+41,1
35 POKEV+23,4:POKEV+29,4
36 GOTO324
37 POKEV+17,PEEK(V+17)OR16
38 TI#="000000":HE=50:POKEV+21,79
39 GOSUB98:L=L+1:GOSUB13:GT=0
40 SYS50400

```

programme

```

41 POKEV+30,0:POKEV+31,0 <96>
42 IFPEEK(V+1)<45THEN60 <67>
43 IFVAL(TI#)<(LE-2)THEN59 <9>
44 HE=HE-1:IFHE=-1THENHE=99:GOSUB13 <53>
45 POKE214,4:POKE211,24:SYSS8640 <227>
46 POKE49337,160:POKE49338,160:PRINT"(BROWN RVSON <30>
N)"HE
47 POKE49702,35:POKE49742,36:POKE49782,35:POKE49 <21>
822,36
48 POKE49776,35:POKE49816,36:POKE49856,35:POKE49 <45>
896,36:POKE51206,120:SYSS1200
49 POKE49788,44:POKE49748,43:POKE49747,44:POKE49 <3>
707,43:POKE49706,38
50 POKE49705,37:POKE49704,44:POKE49664,43 <116>
51 FORT=1T03:POKE55920,T:POKE55960,T:POKE56000,T <60>
:POKE56040,T
52 POKE55846,T:POKE55886,T:POKE55926,T:POKE55966 <193>
,T
53 POKE55932,T:POKE55892,T:POKE55891,T:POKE55851 <135>
,T:POKE55850,T
54 POKE55849,T:POKE55848,T:POKE55808,T:NEXT <51>
55 POKE49702,32:POKE49742,32:POKE49782,32:POKE49 <195>
822,32
56 POKE49788,32:POKE49748,32:POKE49747,32:POKE49 <218>
707,32
57 POKE49706,32:POKE49705,32:POKE49704,32:POKE49 <205>
664,32
58 POKE49776,32:POKE49816,32:POKE49856,32:POKE49 <112>
896,32:TI#="000000" <48>
59 GOSUB9:GOTO40 <225>
60 REM ** SCREEN 2 <169>
61 POKEV+29,0:POKE50170,45:POKEV+4,56 <211>
62 POKEV+17,PEEK(V+17)AND239 <211>
63 POKEV+21,0:GOTO352 <61>
64 POKEV,56:POKEV+1,199:SC=SC+HE*10 <236>
65 POKEV+8,200:POKEV+9,167:POKE715,0:POKEV+11,0 <30>
66 POKEV+17,PEEK(V+17)OR16:HE=0 <24>
67 GOSUB98:POKEV+21,247:L=L+1:GOSUB13 <254>
68 HE=HE+1:IFHE=1THENPOKE50175,51 <39>
69 IFHE=2THENPOKE50175,52 <60>
70 IFHE=3THENPOKE50175,53 <81>
71 IFHE=4THENPOKE50175,54 <102>
72 IFHE=5THENPOKE50175,55 <93>
73 IFHE=6THENPOKE50175,54 <84>
74 IFHE=7THENPOKE50175,53 <131>
75 IFHE=8THENPOKE50175,52:HE=0 <254>
76 IFPEEK(V+1)<45THEN99 <135>
77 POKE53240,0 <110>
78 IFVAL(TI#)<5THEN85 <245>
79 POKE51206,255:SYSS1200
80 POKE49526,37:POKE49527,38:POKE49528,37:POKE49 <202>
529,38:POKE49530,37
81 POKE49531,38:POKE49532,37:POKE49533,38:FORT=1 <112>
T04:POKE55670,T:POKE55671,T
82 POKE55672,T:POKE55673,T:POKE55674,T:POKE55675 <83>
,T:POKE55676,T:POKE55677,T
83 NEXT:POKE49526,32:POKE49527,32:POKE49528,32:P <122>
OKE49529,32:POKE49530,32
84 POKE49531,32:POKE49532,32:POKE49533,32:TI#="0 <69>
00000" <84>
85 GOSUB9 <110>
86 IFPEEK(715)<>0THEN89
87 POKEV+10,PEEK(V+8):POKE715,255:POKEV+11,152:P <174>
OKE714,0 <247>
88 POKE51206,30:SYSS1200
89 FORT=0T02:IFPEEK(53240)=1THEN91 <151>
90 NEXT:GOTO68 <219>
91 POKE707,0:POKE53248+3,0:SC=SC+10:GT=GT+1:GOSU <129>
B98
92 POKE214,19:POKE211,16:SYSS8640:PRINT"(RVSON B <123>
ROWN)"BT
93 IFGT<>15THEN68 <154>
94 POKE49743,32:POKE49783,32:POKE49823,32 <36>
95 POKEV+21,199:POKEV+10,0 <94>
96 FORT=0T05:POKEV+33,1:POKE51206,T:SYSS1200:POK <151>
EV+33,0:NEXT <250>
97 GOTO76
98 POKE214,22:POKE211,7:SYSS8640:PRINT"(RVSON BR <169>
OWN)"SC:RETURN
99 REM ** SCREEN 3
100 POKEV+17,PEEK(V+17)AND239 <18>
101 POKEV+21,0:GOTO380 <249>
102 HE=0:TI#="000000":BO=100 <2>
103 GOSUB98 <21>
104 POKEV+17,PEEK(V+17)OR16 <14>
105 POKEV+21,223:L=L+1:GOSUB13 <99>
<71>
106 HE=HE+1:IFHE=1THENPOKE50175,48 <175>
107 IFHE<>1THEN113 <140>
108 POKE51206,1:SYSS1200:POKE49678,45:POKE49679, <13>
46:POKE49680,45:POKE49681,46
109 FORT=1T04:POKE55822,T:POKE55823,T:POKE55824, <132>
T:POKE55825,T:NEXT
110 POKE49678,32:POKE49679,32:POKE49680,32:POKE4 <234>
9681,32
111 BO=BO-1:POKE214,20:POKE211,14:SYSS8640:PRINT <63>
"(RVSON BROWN SPACE6 LEFT5)"BO
112 IFBO=0THENBOGOSUB13:BO=100 <146>
113 IFHE<>ATHEN120 <143>
114 HE=0:POKE51206,250:SYSS1200:POKE49779,35 <239>
115 POKE49819,35:POKE49859,36:POKE49281,36:POKE4 <19>
9321,35:POKE49361,36
116 FORT=1T05:POKE55923,T:POKE55963,T:POKE56003, <234>
T:POKE55425,T:POKE55465,T
117 POKE55505,T:NEXT <9>
118 POKE49779,32:POKE49819,32:POKE49859,32:POKE4 <241>
9281,32:POKE49321,32
119 POKE49361,32 <175>
120 IFPEEK(V+1)<45THEN123 <57>
121 GOSUB9 <120>
122 GOTO106 <205>
123 REM ** SCREEN 3 VERLASSEN <144>
124 SC=SC+BO*10 <78>
125 POKEV+21,0:IFINT(LE)<>3THENLE=LE-.5 <146>
126 GOSUB42B:PRINT"(DOWN3 SPACES RED RVSON SPACE <6>
27)"
127 PRINT"(SPACES RVSON RED SPACE BROWN SPACE25 <169>
RED SPACE)"
128 PRINT"(SPACES RVSON RED SPACE BROWN SPACE)SU <31>
PER... (SPACE)DOCH(SPACE)IHR(SPACE)LEBEN(SPACE RE
D SPACE)"
129 PRINT"(SPACES RVSON RED SPACE BROWN SPACE2)I <204>
ST(SPACE)NUCH(SPACE)IN(SPACE)GEFAHR...(SPACE2 RE
D SPACE)"
130 PRINT"(SPACES RVSON RED SPACE BROWN SPACE25 <172>
RED SPACE)"
131 PRINT"(SPACES RED RVSON SPACE27)" <140>
132 FORT=160T01STEP-1:POKE51206,T:SYSS1200:NEXT <32>
133 FORT=2T0160:POKE51206,T:SYSS1200:NEXT:FORT=0 <91>
T0300:NEXT
134 GOTO30 <5>
135 PRINT"(CLEAR)":POKE53280,0:POKE53281,0 <210>
136 PRINTTAB(7)"(RED DOWN9)BITTE(SPACE)CA(SPACE) <142>
30(SPACE)SEC(SPACE)WARTEN:"
137 PRINTTAB(7)"(DOWN)DATEN(SPACE)WERDEN(SPACE)E <80>
INGELESSEN"
138 DATA120,169,51,133,1,169,0,133,95,133,90,133 <162>
,88,169,208,133,96,169,240,133
139 DATA89,169,224,133,91,32,191,163,169,55,133, <201>
1,88,96
140 DATA120,169,196,141,21,3,169,237,141,20,3,88 <183>
,96,174
141 DATA0,208,142,12,208,174,1,208,142,13,208,17 <46>
3,16,208,41,1,170,224,1,208
142 DAT8,173,16,208,9,64,141,16,208,224,0,208,8 <4>
,173,16,208,41,191,141,16,208
143 DATA174,0,220,224,112,176,73,174,195,2,224,0 <63>
,208,66,173,16,208,41,253,141
144 DATA16,208,173,16,208,41,1,170,224,1,208,6,2 <21>
38,16,208,238,16,208,174,0,208
145 DATA142,2,208,174,1,208,142,3,208,162,1,142, <155>
195,2,162,0,142,4,212
146 DATA162,4,142,1,212,162,10,142,5,212,162,15, <0>
142,24,212,162,129,142,4,212
147 DATA174,31,208,142,255,207,174,30,208,142,24 <35>
9,207
148 DATA173,255,207,41,2,170,224,2,208,8,162,0,1 <111>
42,195,2,142,3,208
149 DATA173,255,207,41,32,170,224,32,208,8,162,0 <122>
,142,203,2,142,11,208
150 DATA173,255,207,41,1,170,224,1,208,23,162,57 <248>
,142,0,208,162,199,142,1,208
151 DATA173,16,208,41,254,141,16,208,162,1,142,2 <237>
54,207,169,0,141,192,2,141,193
152 DATA2,173,249,207,41,1,170,224,1,208,23,162, <29>
57,142,0,208,162,199,142,1,208
153 DATA173,16,208,41,254,141,16,208,162,1,142,2 <239>
54,207,169,0,141,192,2,141,193
154 DATA2,174,233,207,224,1,208,24,174,7,208,224 <165>
,70,176,5,162,1,142,199,2,174
155 DATA7,208,224,135,144,5,162,255,142,199,2,17 <118>
3,249,207,41,18
156 DATA170,224,18,208,5,162,1,142,248,207 <117>

```

157 DATA173,0,220,41,239,141,253,207,174,254,207,224,1,240,112	<113>	223 DATA141,153,165,189,157,157,157,157,255	<118>
158 DATA174,253,207,224,107,208,5,169,255,141,192,2,2,174,253,207,224,103	<145>	224 DATA142,141,181,181,149,149,149,153,255	<149>
159 DATA208,5,169,1,141,192,2	<139>	225 DATA143,129,189,189,157,157,157,129,255	<196>
160 DATA174,253,207,224,110,208,5,169,255,141,193,2,174,253,207,224,109,208,5	<158>	226 DATA144,129,189,189,129,159,159,159,255	<125>
161 DATA169,1,141,193,2,174,253,207,224,106,208,8,169,255,141,193,2,141,192,2	<220>	227 DATA146,129,189,189,129,155,157,157,255	<174>
162 DATA174,253,207,224,102,208,10,169,255,141,193,2,169,1,141,192,2	<84>	228 DATA147,129,189,191,129,249,185,129,255	<157>
163 DATA174,253,207,224,105,208,10,169,1,141,193,2,169,255,141,192,2	<29>	229 DATA148,131,239,239,231,231,231,231,255	<126>
164 DATA174,253,207,224,101,208,8,169,1,141,193,2,141,192,2	<39>	230 DATA149,189,189,189,157,157,157,129,255	<11>
165 DATA173,16,208,41,1,170,224,1,240,12,174,0,208,224,30,176,17,169,1,141	<11>	231 DATA150,189,189,189,219,219,231,231,255	<60>
166 DATA192,2,174,0,208,224,60,144,5,169,255,141,192,2,174,1,208,224,40,176	<226>	232 DATA152,189,219,231,131,157,157,157,255	<115>
167 DATA5,169,1,141,193,2,174,1,208,224,200,144,5,169,255,141,193,2,174,5,208	<217>	233 DATA153,189,189,189,195,231,231,231,255	<238>
168 DATA224,90,176,5,162,1,142,197,2,174,5,208,224,154,144,5,162,255,142,197	<172>	234 DATA176,129,189,185,149,141,157,129,255	<16>
169 DATA2,174,6,208,224,182,176,5,162,1,142,198,2,174,6,208,224,232,144,5,162	<161>	235 DATA177,199,247,247,231,231,231,195,255	<241>
170 DATA255,142,198,2	<131>	236 DATA178,129,189,253,129,159,159,129,255	<196>
171 DATA174,8,208,224,150,176,5,162,1,142,200,2,174,8,208,224,240,144,5,162	<94>	237 DATA179,193,253,253,225,249,249,129,255	<43>
172 DATA255,142,200,2	<204>	238 DATA180,191,191,187,187,129,243,243,255	<156>
173 DATA238,250,207,174,250,207,224,2,208,5,169,46,141,254,195	<220>	239 DATA181,193,223,223,193,249,185,129,255	<239>
174 DATA174,250,207,224,12,208,10,169,47,141,254,195,169,2,141,45,208	<249>	240 DATA182,129,191,191,129,157,129,255	<4>
175 DATA174,250,207,224,30,208,10,169,0,141,250,207,169,1,141,45,208	<243>	241 DATA183,129,189,251,247,231,231,255	<17>
176 DATA162,15,169,128,141,61,3,141	<231>	242 DATA184,195,219,219,129,157,157,129,255	<71>
177 DATA60,3,222,208,2,208,44,189,176,2,157,208,2,189,192,2,240,33,16,12,189,0	<164>	243 DATA185,129,189,189,129,249,249,129,255	<63>
178 DATA208,8,222,0,208,40,240,7,208,19,254,0,208,208,14,173,60,3,208,9,173,61	<200>	244 DATA-1	<100>
179 DATA3,77,16,208,141,16,208,173,60,3,208,3,78,61,3,73,128,141,60,3,202,16	<110>	245 FORT=OT0999:READQ:IFQ=-1THEN247	<59>
180 DATA191,76,49,234,-1,162,,142,11,212	<68>	246 POKES0400+T,Q:NEXT	<249>
181 DATA162,255,142,8,212,162,10,142,12,212,162,15,142,24,212,162,129,142,11	<29>	247 FORT=OT0999:READQ:IFQ=-1THEN249	<103>
182 DATA212,96,-1	<131>	248 POKES1200+T,Q:NEXT	<246>
183 DATA24,255,231,195,141,141,195,231,255	<30>	249 READQ:IFQ=-1THEN252	<64>
184 DATA25,254,253,251,247,239,223,191,127	<155>	250 FORT=OT07:READQ:POKES7344+QD*8+T,Q:NEXT	<14>
185 DATA26,127,191,223,239,247,251,253,254	<175>	251 GOTO249	<104>
186 DATA27,1,1,3,7,15,31,127,255	<147>	252 DATA,,15,255,224,26,68,176,26,68,176,31,255,240,12,,96,6,0	<173>
187 DATA28,255,255,255,255,255,255,255,255,255	<162>	253 DATA192,2,,128,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	<210>
188 DATA29,128,128,192,224,240,248,254,255	<226>	254 DATA,,,,,,,,,,,,,,,,,,,,,,,,,,,,,16,,56,,56,,16,,,,,,,,,,,,,,,,,,,,,	<35>
189 DATA30,255,252,252,240,224,192,128,128	<60>	255 DATA,,,,,	<234>
190 DATA31,255,63,63,7,3,3,1,1	<37>	256 DATA,,42,,127,,42,,42,,127,,42,,42,,127,,42,,42,,127,,42,,42,,127	<184>
191 DATA33,0,251,251,251,0,223,223,223	<115>	257 DATA,42,,42,,127,,42,,42,,127,,42,,127,,42,,4	<9>
192 DATA35,32,32,16,8,8,20,36,18	<179>	258 DATA,,,,,,,,,7,255,255,,,,,,,,,,,,,,,,,,,,,	<89>
193 DATA36,1,2,2,4,4,146,107,22,1	<60>	259 DATA,,,,,7,255,255,,,,,,,,,,,,,,,,,,,,,	<132>
194 DATA37,0,0,6,200,48,12,3,0	<219>	260 DATA,,,,,,,,,48,,48,,48,,48,,48,,48,,120,,120,,120,,120,,72,,204	<214>
195 DATA38,0,0,3,100,152,0,0	<43>	261 DATA,,204,,1,182,,3,123,,14,253,192,29,254,224,,	<91>
196 DATA39,0,255,161,145,137,133,66,98	<119>	262 DATA255,255,255,128,,1,127,255,254,32,,4,63,255,252,,78,,78,,78,,78	<76>
197 DATA40,82,74,36,36,36,52,44,70	<76>	263 DATA,,78,,78,,78,,78,,78,,78,,63,255,252,32,,4,127,255,254,128	<137>
198 DATA41,66,98,82,137,133,195,161,255	<255>	264 DATA,1,255,255,255,	<168>
199 DATA42,31,50,73,164,82,41,31,0	<102>	265 DATA,,,,,,,,,5,1,64,11,131,96,12,130,224,11,131,96	<114>
200 DATA43,,,,,128,128,64,64,32	<177>	266 DATA9,130,160,7,1,64,7,1,192,2,,128,2,,128,,	<29>
201 DATA44,80,136,4,3,196,40,16,32	<176>	267 DATA,,,,,,,,,5,1,64,9,131,32,9,131,160,9,131,32,8,130	<111>
202 DATA45,1,2,41,4,146,107,22,1	<176>	268 DATA160,7,1,64,5,1,64,2,,128,2,,128,2,,128,,	<222>
203 DATA46,128,82,137,58,163,74,36,128	<74>	269 DATA,,,,,,,,,16,,16,,59,222,244,119,189,238,239,123,222,239,123,222	<139>
204 DATA73,243,225,192,140,30,30,30,0	<172>	270 DATA119,189,238,59,222,244,7,224,,12,192,,12,192,,24,96,,24,96,,48,48,,48	<29>
205 DATA74,0,30,30,30,30,158,158,222	<178>	271 DATA48,,96,24,,96,24,,	<122>
206 DATA75,253,248,248,120,56,56,56,32	<166>	272 DATA,,,,,,,,,16,,255,255,254,16,,56,,56,,	<232>
207 DATA76,32,56,56,56,56,56,120,253	<153>	273 DATA,,,,,,,,,	<221>
208 DATA77,255,255,255,255,255,255,255,56	<4>	274 DATA,,,,,,,,,16,,63,255,248,,16,,56,,56,,	<38>
209 DATA78,48,231,231,231,231,240,248	<115>	275 DATA,,,,,,,,,	<223>
210 DATA79,255,255,255,255,255,255,123	<249>	276 DATA,,,,,,,,,16,,15,255,224,,16,,56,,56,,	<129>
211 DATA80,51,146,144,145,147,147,51,123	<146>	277 DATA,,,,,,,,,	<225>
212 DATAB1,25,17,129,137,153,153,153,221	<72>	278 DATA,,,,,,,,,16,,3,255,128,,16,,56,,56,,	<247>
213 DATA129,195,219,129,157,157,157,157,255	<91>	279 DATA,,,,,,,,,	<227>
214 DATA130,131,219,193,205,205,205,129,255	<54>	280 DATA,,,,,,,,,16,,254,,16,,56,,56,,	<103>
215 DATA131,129,189,191,159,159,157,129,255	<150>	281 DATA,,,,,,,,,	<229>
216 DATA132,129,221,221,205,205,205,129,255	<169>	282 DATA,,,,,,,,,16,,84,,84,,56,,254,,56,,84,,16,,	<140>
217 DATA133,129,191,191,135,159,159,129,255	<13>	283 DATA,,,,,,,,,	<231>
218 DATA134,129,191,191,135,159,159,159,255	<111>	284 DATA,255,,,,,255,,,,,255,,,,,255,,,,,255	<97>
219 DATA135,129,189,191,145,157,157,129,255	<23>	285 DATA255,,,,,255,,,,,255,,,,,255,,	<29>
220 DATA136,189,189,189,129,157,157,157,255	<254>		
221 DATA137,239,239,239,231,231,231,255	<181>		
222 DATA140,191,191,191,159,159,159,129,255	<112>		

programme

```
286 DATA34,34,36,255,255,255,34,34,36,255,255,25
5,34,34,36,,,,,,,,,,,,,
287 DATA,,,,,,,,,,,,,
288 FORPZ=OT0512STEP64
289 FORT=OT063:READQ:POKE51712+T+PZ,Q:NEXTT
290 NEXTPZ
291 FORPZ=OT0448STEP64
292 FORT=OT063:READQ:POKE52416+T+PZ,Q:NEXTT
293 NEXTPZ
305 GOTO417
306 GETA#
307 IFA#="1"THENLE=6:GOTO311
308 IFA#="2"THENLE=5:GOTO311
309 IFA#="3"THENLE=3:GOTO311
310 GOTO306
311 V=53248:L=5:SC=0:POKE53247,0:POKE53246,0:POK
E56325,26
312 POKEV+27,255:POKE706,0:POKE707,0
313 POKE708,0:POKE709,1
314 POKE710,1:POKE711,0
315 POKE712,1:POKE713,0
316 POKE688,3:POKE689,3
317 POKE690,3:POKE691,1
318 POKE699,2+LE
319 POKEV+39,13:POKEV+40,1
320 POKEV+42,1:POKEV+43,1:POKEV+44,1
321 POKEV+16,0:POKEV,56:POKEV+1,200
322 POKEV+3,0
323 GOTO30
324 REM ** SCREEN 1
325 POKE693,INT(LE):POKE694,INT(LE):POKE696,INT(L
E):POKE 711,0
326 POKE53248+7,126
327 PRINT"(CLEAR UP HGREY){X}{SPACE}{X}{SPACE}{X
}{SPACE}{X}{SPACE BROWN}{X}{SPACE4}{\CR
VSON)TIME(RVSOFF){SPACE2}"
329 PRINT"(UP HGREY){SPACE2}{SPACE2}{SPACE
}{SPACE}{SPACE BROWN}{SPACE4}{(RVSON)LI
MIT(RVSOFF){SPACE}"
330 PRINT"(UP HGREY){SPACE}{SPACE2}{SPACE2}{
SPACE}{SPACE}{SPACE}{SPACE BROWN}{SPACE3}
{SPACE3}"
331 PRINT"(UP HGREY){SPACE}{SPACE2}{SPACE2}{
SPACE}{SPACE}{SPACE2}{BROWN}{SPACE3}{SPACE3}
{SPACE3}"
332 PRINT"(UP){SPACE3}{SPACE3}{SPACE3}{SPACE3}
{SPACE3}{SPACE3}"
333 PRINT"(UP){SPACE3}{SPACE3}{SPACE3}{SPACE3}
{SPACE3}{SPACE3}{SPACE3}{SPACE3}"
334 PRINT"(UP){SPACE3}{SPACE3}{SPACE3}{SPACE3}
{SPACE3}{SPACE3}{SPACE3}{SPACE3}"
335 PRINT"(UP SPACE){SPACE12}{SPACE5}{SPACE5}
{SPACE4}"
336 PRINT"(UP SPACE2){SPACE11}{SPACE12}{
SPACE4}"
337 PRINT"(UP SPACE2){SPACE10}{SPACE10}{
SPACE4}"
338 PRINT"(UP){SPACE11}{SPACE8}{SPACE}
{SPACE6}"
339 PRINT"(UP){SPACE5}{SPACE5}{SPACE5}{SPACE5}
{SPACE6 BROWN}"
340 PRINT"(UP){SPACE2}{SPACE5}{SPACE HGRE
Y}{BROWN SPACE4}{SPACE10}"
341 PRINT"(UP){SPACE10}{SPACE HGREY}{SPACE5}
{BROWN SPACE11}"
342 PRINT"(UP){SPACE10}{HGREY}{BROWN SPACE
21 YELLOW}{BROWN}"
343 PRINT"(UP){SPACE10}{HGREY}{SPACE}{BRO
WN SPACE20}"
344 PRINT"(UP){SPACE2}{HGREY}{SPACE}{BRO
WN SPACE10}{HGREY}{BROWN}{SPACE3}"
345 PRINT"(UP){HGREY}{SPACE3}{BROWN}{
HGREY}{SPACE}{BROWN SPACE2}{HGREY}{BRO
WN}"
346 PRINT"(UP){HGREY}{SPACE3}{BROWN}{
HGREY}{BROWN}{SPACE}"
347 PRINT"(UP){HGREY}{RED C13 HGREY}{BROWN}
{SPACE}"
348 PRINT"(UP){SPACE}{SPACE}"
349 PRINT"(UP RVSON SPACE)SCORE(SPACE12 RVSOFF)
{SPACE}{SPACE3}{RVSON SPACE3}SHIPS(SPACE6)"
350 PRINT"(UP){SPACE2}{SPACE2}"
```

```
\\\\\\":POKE50111,160:POKE56255,9
351 GOTO37
352 REM ** SCREEN 2
353 POKE50175,51:POKEV+16,PEEK(V+16)OR128:POKEV+
46,5
354 POKEV+14,6:POKEV+15,99:POKEV+43,1
355 PRINT"(CLEAR BROWN UP SPACE3){SPACE3}{
ACES}{SPACE3}{SPACE7}"
356 PRINT"{BROWN UP SPACE2}{SPACE}{SPACE4}
{SI SK SH SO}{SPACE5}"
357 PRINT"(UP SPACE){SPACE8}{SPACE}
{SPACE3}{SPACE4}"
358 PRINT"(UP){SPACE16}{SPACE3}{SPACE3}
{SPACE3}"
359 PRINT"(UP){SPACE7}{SPACE9}{SPACE3}
{SPACE2}{SPACE3}"
360 PRINT"(UP){SPACE7}{SPACE6}{SPACE}
{SPACE3}"
361 PRINT"(UP){SPACE5}{SPACE2}{SPACE4}
{ELLON}*{BROWN}{SPACE8}"
362 PRINT"(UP){SPACE10}{SPACE7}{SPACE}
{SPACE5}"
363 PRINT"(UP){SPACE8}{SPACE8}{SPACE}
{SPACE3}"
364 PRINT"(UP){SPACE4}{SPACE8}{SPACE}
{SPACE10}"
365 PRINT"(UP){SPACE5}{SPACE8}{SPACE}
{SPACE5}*{BROWN}"
366 PRINT"(UP SPACE){SPACE4}{SPACE HGREY}
{BROWN SPACE}{SPACE3 HGREY}{SPACE}{BRO
WN SPACE3}"
367 PRINT"(UP SPACE2){SPACE3}{SPACE HGRE
Y}{BROWN SPACE2}{SPACE HGREY}{SPACE}{BRO
WN SPACE5}"
368 PRINT"(UP SPACE){SPACE4}{HGREY}{BRO
WN SPACE10}{SPACE5}"
369 PRINT"(UP){SPACE8}{HGREY}{BROWN SPACE
16 GREEN}{BROWN SPACE4}"
370 PRINT"(UP){SPACE8}{HGREY}{BROWN SPACE
16 GREEN}{BROWN SPACE4}{SPACE}"
371 PRINT"(UP){SPACE8}{HGREY}{BROWN SPACE
16 GREEN}{BROWN SPACE3}{SPACE}"
372 PRINT"(UP){HGREY}{SPACE4}{BROWN}{
HGREY}{BROWN}{SPACE}"
373 PRINT"(UP){HGREY}{SPACE4}{BROWN}{
HGREY}{BROWN}{SPACE}"
374 PRINT"(UP){HGREY}{SPACE4}{BROWN}{
RVSD N}HITS(RVSOFF){RVSON}{RVSOFF}{HGRE
Y}"
375 PRINT"(UP){HGREY}{RED C04 HGREY}{BRO
WN}{SPACE}"
376 PRINT"(UP){SPACE}{SPACE}"
377 PRINT"(UP RVSON SPACE)SCORE(SPACE13 RVSOFF)
{SPACE2}{RVSON SPACE3}SHIPS(SPACE6)"
378 PRINT"(UP){SPACE3}{SPACE3}
{SPACE3}{SPACE3}POKE50111,160
379 POKE56255,9:GOTO64
380 REM ** SCREEN 3
381 POKEV+16,PEEK(V+16)AND127
382 POKE50175,48:POKEV+14,49:POKEV+15,150:POKEV+
46,2
383 POKEV+4,248:POKEV+23,0:POKE50170,42
384 POKE50171,56:POKE711,1:POKE695,2:POKE53225,1
385 POKEV+9,140:POKE50172,58:POKEV+29,16:POKE696
,1:POKEV+43,2
386 PRINT"(CLEAR BROWN UP SPACE){SPACE2}{
HGREY}"
387 PRINT"{UP BROWN}{SPACE3}{HGREY}"
388 PRINT"(UP BROWN){SPACE3}{SPACE3}
{SPACE}{SPACE}{HGREY}"
389 PRINT"(UP BROWN){SPACE8}{SPACE5}{
SPACE3}"
390 PRINT"(UP){SPACE13 CO SPACE13}{SPACE8}"
391 PRINT"(UP){SPACE11}{SPACE18}{SPACE3}"
392 PRINT"(UP){HGREY}{BROWN}{
SPACE12}{SPACE3}"
393 PRINT"(UP SPACE){SPACE7}{SPACE}
{SPACE4}"
394 PRINT"(UP SPACE){SPACE19}{SPACE5}{
SPACE4}"
395 PRINT"(UP SPACE){SPACE12}{SPACE3}{
SPACE3}{SPACE3}"
396 PRINT"(UP SPACE){SPACE7}{SPACE3}"
```


Zeitschriften und Programme von 1986

Heft 1/86

Commodore
 Bestell-Nr. COM CK 1/2 16,- DM
 Bestell-Nr. COM CD 1/2 20,- DM
 Checksummer 1.0 C, Astronaut Battle (C-64), Hueycobra (C-16/116), Hardcopy-routine (C-16/116), Holmut und die Zeitmaschine (C-16/116), Locos (VC-20), Rock'ne (VC-20), Charly der Raketenmann (VC-20), Interrupt-Programmierung (VC-20)

Schneider
 Bestell-Nr. COM SK 1/2 16,- DM
 Bestell-Nr. COM SD 1/2 39,- DM
 Defender, Boulder Dash, Basic-Befehle auf den Funktionstasten

Heft 5/86

Commodore
 Bestell-Nr. COM CK 5/2 16,- DM
 Bestell-Nr. COM CD 5/2 20,- DM
 Space-Bewil (C-64), Disk-Eddi (C-64), ABC-Legen (V-20), 3-B-Labyrinth (VC-20), Tzap (VC-20), Invaders of Space (C-16/116), Submarine (C-16/116), Balkendiagramm (C-16/116)

Schneider
 Bestell-Nr. COM SK 5/2 16,- DM
 Bestell-Nr. COM SD 5/2 39,- DM
 Startank, Grafik-Designer, Cross-Reference, Itext

Heft 9/86

Commodore
 Bestell-Nr. COM CK 9/2 16,- DM
 Bestell-Nr. COM CD 9/2 20,- DM
 Moonpatrol (VC20), Falcons Lander (VC20), Frogger (VC20), The Games (C64), Cave of Death (C64) Sprites and Sound mit dem UMH-System (C64), Diggy (C16/116), Tapedirectory (C16/116), Reversi (C16/116)

Schneider
 Bestell-Nr. COM SK 9/2 16,- DM
 Bestell-Nr. COM SD 9/2 39,- DM
 Creepy, Briefeschreiben, Fruity Man, Mirror Battle

Heft 2/86

Commodore
 Bestell-Nr. COM CK 2/2 16,- DM
 Bestell-Nr. COM CD 2/2 20,- DM
 Robo (C-64), Basic-Befehlsweiterung (C-64), Raketenstart (C-16/116), Dreikampf (C-16/116), Headline (C-16/116), Defender (VC-20), Formel 1 (VC-20), Memory-Display (VC-20)

Schneider
 Bestell-Nr. COM SK 2/2 16,- DM
 Bestell-Nr. COM SD 2/2 39,- DM
 Alien 2001, Diamond Hunter, Directory

Heft 6/86

Commodore
 Bestell-Nr. COM CK 6/2 16,- DM
 Bestell-Nr. COM CD 6/2 20,- DM
 Music-Master (C-64), The little Puck goes home (C-64), EDDY (C-64), Invers-Utility (C-16/116), Checksummer (C-16/116), Affenwurf (C-16/116), Data-Maker (VC-20), Karate-Kid (VC-20), Q-Bert (VC-20)

Schneider
 Bestell-Nr. COM SK 6/2 16,- DM
 Bestell-Nr. COM SD 6/2 39,- DM
 Disksort, Mini-Forth, Starcommand

Heft 10/86

Commodore
 Bestell-Nr. COM CK 10/2 16,- DM
 Bestell-Nr. COM CD 10/2 20,- DM
 Laufschrift (VC20), Checksummer (VC20), Treasure Robber (VC20), Scramble (VC20), Refusal-Basiclader (C-16), Hardy-Copy-Blitz (C-16), Auto-start (C-16), Fill-Utility (C-16), Railway (C-16), Domino (C-16), Sprite Turn/Character-Switch (C-64), Checksummer (C-64), Horse Games (C-64), Patrouille (C-64)

Schneider
 Bestell-Nr. COM SK 10/2 16,- DM
 Bestell-Nr. COM SD 10/2 39,- DM
 Checksummer, Panic-Car, Scurfy Joe, Drawgenerator, Airfighter

Heft 3/86

Commodore
 Bestell-Nr. COM CK 3/2 16,- DM
 Bestell-Nr. COM CD 3/2 20,- DM
 Minor-Jumper (C-16/116), Hero (C-16/116), Data-Maker (C-16/116), Befehlsweiterung (C-64), Starship (C-64), Mini-Gar-Race (C-64), Text-Scroller (C-64), Text-Editor (C-64), Checksummer (VC-20), Robin at Ravenwreck (VC-20), Mini-Karate (VC-20), Autonumber-Routine (VC-20)

Schneider
 Bestell-Nr. COM SK 3/2 16,- DM
 Bestell-Nr. COM SD 3/2 39,- DM
 Smily, Scrupi, Die verrückte Fabrik, Disc-Help

Heft 7/86

Commodore
 Bestell-Nr. COM CK 7/2 16,- DM
 Bestell-Nr. COM CD 7/2 20,- DM
 VC-20-Sound-Demo, Mad Rush (VC-20), Ghost-Hell (VC-20), Roadrunner (C-64), Basic Extension (C-64), Meteoritensturm (C-64), Checksummer C-16/116, Airwolf (C-16/116), Break-Out (C-16/116)

Schneider
 Bestell-Nr. COM SK 7/2 16,- DM
 Bestell-Nr. COM SD 7/2 39,- DM
 Poker, Cave-Walker, Diskettenarchiv

Heft 11/86

Commodore
 Bestell-Nr. COM CK 11/2 16,- DM
 Bestell-Nr. COM CD 11/2 20,- DM
 Cooky (VC20), Gambol (VC20), Wostok (VC20), Topball (C16), Zauberpuzzle (C16), Schieberoutine (C16), Atlantis (C64), Astronomie (C64), Deutsche Fehlermeldung (C64), Remzeilenkiller (C64)

Schneider
 Bestell-Nr. COM SK 11/2 16,- DM
 Bestell-Nr. COM SD 11/2 39,- DM
 Diskettenhilfe, Horror Driver, Disc of Tron, Space Hawk, Morsedecoder

Heft 4/86

Commodore
 Bestell-Nr. COM CK 4/2 16,- DM
 Bestell-Nr. COM CD 4/2 20,- DM
 Shuttle (VC-20), Josef in den Höhlen (VC-20), Marge-Routine (VC-20), Shoot it (VC-20), Fast Trust (C-64), Data-Text 64 (C-64), Gary, der schnelle Kabelverleger (C-16/116), SDI (C-16/116), Beep-Routine (C-16/116)

Schneider
 Bestell-Nr. COM SK 4/2 16,- DM
 Bestell-Nr. COM SD 4/2 39,- DM
 Space-Rescue, Texteditor, Starlight, Programm-Retter

Heft 8/86

Commodore
 Bestell-Nr. COM CK 8/2 16,- DM
 Bestell-Nr. COM CD 8/2 20,- DM
 Death Desert (C-64), 3-D-Labyrinth (C-64), Klick (C-64), Bahnhof (VC-20), Omega Attack (VC-20), Sound (VC-20), Platinenlayout (C-16/116), Balloon Shoot (16/116), Hardcopy (C-16/116)

Schneider
 Bestell-Nr. COM SK 8/2 16,- DM
 Bestell-Nr. COM SD 8/2 39,- DM

Aus diesem Heft:

Commodore
 Bestell-Nr. COM CK 12/2 16,- DM
 Bestell-Nr. COM CD 12/2 20,- DM

Schneider
 Bestell-Nr. COM SK 12/2 16,- DM
 Bestell-Nr. COM SD 12/2 39,- DM

Alle hier aufgeführten Kassetten und Disketten können jederzeit nachbestellt werden.

Bei den Zeitschriften sind die Ausgaben 5, 9, 12 (85) und 5, 6, 7, 8, 9, 10, 11 (86) noch vorhanden.

Superpakete!
 Beachten Sie bitte unsere neu geschnürten Programmpakete.
 ... und das zu einem Superpreis!

Neu: Superpakete zum Superpreis!

C=64

Superpaket 1 Spiele

Monster Attack, Projekt, High Noon, Brieftaube, Cadilon, Turbo Jet, Sea Attack, The Maze, Graze Game, Gaixi, Space Comets, Sheet, Vampir, Memory

SP-C-K-1 Kassette 20,-
SP-C-D-1 Diskette 20,-

Superpaket 2 Spiele

Tank Duell, Astronaut, Jungle 1, Jungle 2, Sufo, Fire-Legion, Spiders, Star Battle, Super Breakout, Mission Rescue, Roadpainter, Moonsweeper, Scotti

SP-C-K-2 Kassette 20,-
SP-C-D-2 Diskette 20,-

Superpaket 3 Anwender

The Basic, Basicreset, Data-Generator, Print-Generator, Supermon-64, Restore N, Disky 10, Fill-Loader, Funktionstasten-Belegung, Autostart, Multibasic, Bildeditor, Spritehilfe + Demo, Autonumber, Micro-Tools + Mloader

SP-C-K-3 Kassette 20,-
SP-C-D-3 Diskette 20,-

Programm	Preis		Bestell-Nr.
	Kass./Disk.		
Tron/Mercurious	24,50/29,50		C 13
Interceptor			
Base/Schotter	24,50/29,50		C 23
Jump Man/			
The Way/Space Taxi	24,50/29,50		C 33
Fight Night/Monster-			
Jagd/Colossus Mühle	24,50/29,50		C 43
Mr. Postman/Moon-			
tunnel/Moonrallies/			
Steinschlag	24,50/29,50		C 53
Taxi Driver/			
Deadly Mission	24,50/29,50		C 63

C=16

Superpaket 1

Horror Castle, Fantasy County, Senso, Jumping Joe, Schatztaucher, Shaolin, City Runner

SP-O-K-1 Kassette 20,-
SP-O-D-1 Diskette 20,-

Programm	Preis		Bestell-Nr.
	Kass./Disk.		
Goldrausch/			
Moon Fighter	18,-/23,-		O 33
Frogger/Humor	18,-/23,-		O 43
Cosmic Terror/			
Schwimmen/Break In	18,-/23,-		O 53
Panic/Phantom/			
Meteorstorm	18,-/23,-		O 63

VIC-20

Superpaket 1

Flowers, Tanker, Jupiter Rescue, Einbrecher, Hospital, Alien, Flipperball, Sub-Zone, Dr. Cerp, Geo-Zayxon, Froggies Night Mare, Punktefresser, Starwars, Salvage Crew

SP-V-1 nur Kassette 20,-

Programm	Preis		Bestell-Nr.
	Kass.		
Brückenbau/Jango	16,-		V 13
Miner, der Fassaden-			
streicher/Inkaschatz	16,-		V 23
Bergshooting/			
Bobby in Action	16,-		V 33
Josef in den			
Katakomben	8,50		V 43
Hot Food/			
Men-Rescue/			
Soundprogramm	14,50		V 53
Castle Dracula/			
UFO-Destroyer	14,50		V 63

Schneider

CPC 604

Superpaket 1 Spiele

Alien Hunter, Catch, Interceptor 3 D, City Lander, Conan's Castle, Copter Patrol, CPC Bert, Crash, Demon Attack, Duell, Etis, Labyrinth, Miner, Tower, Snider's Maze, Geister Schloß, Fallschirmspringer, Secret Valley, Aladin, Hounded Castle, Cave runner, Cobra, Erwin, Kruezi, Q-Bert

SP-SR-K-1 Kassette 20,-
SP-SR-D-1 Diskette 27,-

Superpaket 2 Anwender

Back up, Data gen, Datenverwaltung, Discopy, Monitor, Painter, Remkiller, SU-File, Tastaturbelegung

SP-SR-K-2 Kassette 20,-
SP-SR-D-2 Diskette 27,-

Programm	Preis		Bestell-Nr.
	Kass./Disk.		
Schneider Pank/			
Killer Ship	18,50/28,50		SR 13
Midnight/Horror Caves	18,50/28,50		SR 23
Spider Maze	18,50/28,50		SR 33
Triton/Traumland	18,50/28,50		SR 43
Starpatrol/			
Helicopter Pilot	18,50/28,50		SR 53
Red Alert/			
Speedy-Froge/Aurion	18,50/28,50		SR 63

software service

Der Tronic-Verlag bietet Spitzensoftware für jedermann.

Überzeugen Sie sich von unseren Angeboten und bestellen Sie noch heute. Alle Bestellungen werden innerhalb nur

1 Woche

nach Eingang bearbeitet. Telefonische Bestellungen unter der Rufnummer

0 56 51/3 00 11

Anleitungen zu unseren Programmen erhalten Sie aus unseren Heften. Bei ausverkauften Heften erhalten Sie zum Heftpreis Fotokopien der Spielbeschreibungen.

programme

```

310 FORI=1TOA:POKE630+I,ASC(MID$(A$,I,1)):NEXT:P
OKE198,A:END
0 REM *****
1 REM *
2 REM *      JOE THE BALL
3 REM *      *****
4 REM *
5 REM *****
6 REM * (W) 1986 BY SEBASTIAN RICHTER *
7 REM *      TELEFON: 040/433212
8 REM *****
9
10 POKE53272,20:POKE53280,0:POKE53281,0
20 POKE53269,0:PRINT"(CLEAR CTRLH WHITE)":CLR
30 V=53248:5=54272:XX=0:YY=0:X=0:Y=0
40 GOTO 210:REM"(SPACE SM SM SPACE)HAUPTPROGRAMM
(SPACE SM SM
50 J=PEEK(V+30):J=PEEK(56320):POKEV+17,23
60 IF(JAND4)=0ANDXX->9THENXX=XX-.4
70 IF(JANDB)=0ANDXX<9THENXX=XX+.4
80 IF(JAND16)=0THENGOBUB190:GOTO50
90 X=X+XX:IFX<24THENX=24
100 YY=YY+.9:Y=Y+YY:IFY>205THENY=205:YY=-8:POKE2
040,226:POKES+4,33:GOTO120
110 IFY>190THENPOKE2040,225
120 IFY<190ANDYY<0THENPOKE2040,224:POKES+4,0
130 POKE53249,Y:POKE53248,XAND255
140 X1=X/256:IFX1=>1THENPOKEV+16,PEEK(V+16)ORI:G
OTO160
150 POKEV+16,PEEK(V+16)AND254
160 IFX>340THENR=1:RETURN
170 IF(PEEK(V+30)AND1)=1THENR=0:RETURN
180 GOTO 50
190 PRINT"(HOME DOWN2)"SPC(17)"(BLACK RVSON)PAUS
E":POKE198,0:WAIT198,1:PRINT"(LEFTS SM SM SM SM
SM)":RETURN
200 REM
210 X$="(RIGHTS RVSON)":GOSUB380:RESTORE:PL=1
220 READA$:IFA$="*"THEN250
230 ILEFT$(A$,1)="*"THENPRINTX$:MID$(A$,2):GOTO
220
240 READY:GOSUB400:GOTO220
250 DATA*(RIGHT2 SPACE RIGHT SPACE3 RIGHT SPACE
3 RIGHTS SPACE3 RIGHT SPACE3 RIGHT SPACE RIGHT3
SPACE)"
260 DATA*(RIGHT2 SPACE RIGHT SPACE RIGHT SPACE
RIGHT SPACE RIGHT7 SPACE RIGHT SPACE RIGHT SPACE
RIGHT SPACE RIGHT SPACE RIGHT3 SPACE)"
270 DATA*(RIGHT2 SPACE RIGHT SPACE RIGHT SPACE
RIGHT SPACE2 RIGHT2 RVSOFF)THE(RVSON RIGHT SPACE
3 RIGHT SPACE3 RIGHT SPACE RIGHT3 SPACE)"
280 DATA*(RIGHT2 SPACE RIGHT SPACE RIGHT SPACE
RIGHT SPACE RIGHT7 SPACE RIGHT SPACE RIGHT SPACE
RIGHT SPACE RIGHT SPACE RIGHT3 SPACE)"
290 DATA*(SPACE3 RIGHT SPACE3 RIGHT SPACE3 RIGH
T5 SPACE3 RIGHT SPACE RIGHT SPACE RIGHT SPACE3 R
IGHT SPACE3)"
300 DATA"COPYRIGHT(SPACE)1986(SPACE)BY(SPACE)SEB
ASTIAN(SPACE)RICHTER",10
310 DATA"1(SPACE)PLAYER(SPACE BROWN)OR(SPACE2)2(S
PACE)PLAYER(WHITE)",15
320 DATA"PRESS(SPACE)FIRE(SPACE)BUTTON(SPACE)TO(S
PACE)START(SPACE)THE(SPACE)GAME",23,"*"
325 FORI=70770766:POKEI,0:NEXT:POKE704,170:POKE7
05,170:POKE706,170
330 J=PEEK(56320):IF(JAND16)=0THEN430
340 IF(JAND4)=0THENPL=1:GOSUB370:PRINT"(WHITE)1(S
PACE)PLAYER(BROWN SPACE2)OR(SPACE2)2(SPACE)PLAY
ER"
350 IF(JANDB)=0THENPL=2:GOSUB370:PRINT"(BROWN)1(S
PACE)PLAYER(SPACE2)OR(WHITE SPACE2)2(SPACE)PLAY
ER"
360 BOTO 330
370 POKE211,8:POKE214,15:SYS58640:RETURN
380 FORI=0TO24:POKES+I,0:NEXT
390 POKES+24,15:POKES+5,5:POKES+1,100:RETURN
400 X=(40-LEN(A$))/2:FORI=1TOLEN(A$):POKE211,X:P
OKE214,Y:SYS58640
410 B$=MID$(A$,I,1):IFB$<>"(SPACE)"ANDB$<>"(WHIT
E)"ANDB$<>"(BROWN)"THENPOKES+4,33
420 PRINTB$:FORI=0TO10:NEXT:POKES+4,32:X=X+1:IF
PEEK(56320)<>11THENNEXT:RETURN
430 PRINT"(CLEAR)":FORI=1TO2:SC(I)=0:LI(I)=3
440 LE(I)=1:NEXT:POKE646,0:PI=1
450 GOSUB1200:POKEV+17,0:POKEV+33,14
455 PRINT"(CLEAR BLACK RVSON SM SM SM SM SM S

```

```

(9)
(3)
(98)
(66)
(97)
(101)
(8)
(38)
(26)
(11)
(67)
(39)
(148)
(89)
(82)
(104)
(53)
(169)
(158)
(117)
(251)
(92)
(31)
(152)
(102)
(229)
(81)
(129)
(55)
(227)
(87)
(83)
(98)
(20)
(122)
(237)
(2)
(225)
(22)
(59)
(224)
(150)
(168)
(83)
(128)
(67)
(145)
(177)
(21)
(171)
(233)
(55)
(178)
(0)
(135)
(36)
(225)
N SM SM SM SM SM SM SM SM SM SM SM SM SM SM SM S
N SM SM SM SM SM SM SM SM SM SM SM SM SM SM SM S
N SM)":
460 PRINT"(SM SM SM)SCORE(SM SM SM)LIVES(SM S
N SM SM)SCREEN(SM SM SM SM)PLAYER(SM SM SM)":
470 PRINT"(SM SM SM SM SM SM SM SM SM SM SM SM SM S
N SM SM SM SM SM SM SM SM SM SM SM SM SM SM SM S
N SM SM SM SM SM SM SM SM SM SM SM SM)":
480 PRINT"(SM SM SM SM SM SM SM SM SM SM SM SM SM S
N SM SM SM SM SM SM SM SM SM SM SM SM SM SM SM S
N SM SM SM SM SM SM SM SM SM SM SM SM)":
490 PRINT"(CO CP CO SPACE CO CP CO SPACE CO CP C
O CP CO CP CO CP CO CP CO SPACE CO CP CO CP CO S
PACE2 CO CP CO2 CP CO CI CO CP CO SPACE CO CP)"
500 POKE214,20:POKE211,0:SYS58640:PRINT"(LIG.GRE
EN SPACE CO CP CO CP CO SPACE CO SPACE CO CP CO
CP CO SPACE CO CP CO CP CO CP CO SPACE CO CP CO
CP CO CP CO SPACE CO CP CO)":
510 PRINT"(CO SPACE CO CP CO)":FORI=1TO3:PRINT"
(RVSON SPACE40)":
520 NEXT:SS=0:P$="0"+CHR$(48+PI)
530 S$=RIGHT$("000000"+MID$(STR$(SC(PI)),2),6):L
$="0"+CHR$(47+LI(PI))
540 B$="0"+CHR$(48+LE(PI)):PRINT"(BLACK RVSON HO
ME DOWN2 RIGHT3)"S$(RIGHT5)"L$(RIGHT7)"B$(RIG
HT8)"P$
550 FORI=V+1TOV+15STEP2:POKEI,200:POKEI-1,0:NEXT
:FORI=1TO7:POKEV+39+I,0
560 POKE2040+I,227+SS:SS=1-SS:NEXT:POKEV+28,255:
POKEV+37,1:POKEV+38,11:POKEV+39,1
570 ONLE(PI)GOSUB1000,1010,1020,1030,1040,1050,1
060,1070,1080,1090,1100
580 GOSUB660:IFR=0THEN600
590 LE(PI)=LE(PI)+1:SC(PI)=SC(PI)+100*LE(PI):GOT
O450
600 POKES+4,0:POKES+5,0:POKES+6,240:POKES+1,200:
POKE2040,224:POKES+4,129
610 FORI=0TO300:NEXT:POKE2040,225:POKES+24,10
620 FORI=0TO300:NEXT:POKE2040,226:POKES+24,5
630 FORI=0TO300:NEXT:POKEV+21,254:POKES+24,0:LI(
PI)=LI(PI)-1:IFLI(PI)>0THEN700
640 PRINT"(HOME DOWN1)"SPC(15)"GAME(SPACE2)OVER
"
650 PRINTSPC(16)"(DOWN)PLAYER"PI:GOTO720
660 GOSUB380:POKES+1,10:X=24:Y=150
670 POKEV+1,Y:POKEV,X:POKE2040,224
680 XX=-.4:YY=-.2:GOSUB50:RETURN
690 PI=PI+1:IFPI>PLTHENPI=1
710 GOTO 450
720 WAIT56320,16,16:IFPL=1THENRUN
730 PL=1:PI=2:GOTO450
997 REM *****
998 REM * THE LEVEL'S *
999 REM *****
1000 POKEV+2,100:POKEV+4,200:POKEV+21,7:RETURN
1010 POKEV+2,100:POKEV+4,121:POKEV+6,255:POKEV+8
,230:POKEV+21,31:RETURN
1020 POKEV+2,20:POKEV+4,255:POKEV+6,170:POKEV+8,
100:POKEV+21,31:RETURN
1030 POKE214,20:POKE211,10:SYS58640:PRINT"(SPACE
3 DOWN LEFT3)^^^:POKEV+2,104:POKE2041,11
1031 POKEV+40,14:POKEV+4,20:POKEV+6,255:POKEV+21
,15:POKEV+3,222:RETURN
1040 POKEV+2,100:POKEV+4,140:POKE2042,11:POKEV+4
1,14:POKEV+21,15:POKEV+6,230
1041 POKEV+5,222:POKE214,20:POKE211,14:SYS58640:
PRINT"(SPACE4 DOWN LEFT4)^^^:RETURN
1050 POKEV+2,150:POKEV+4,171:POKEV+6,192:POKEV+2
1,15:RETURN
1060 PRINT"(HOME DOWN7 RIGHT7)GLEICH(SPACE)HAST(S
PACE)DU(SPACE)ES(SPACE)GESCHAFT":POKEV+21,1:RET
URN
1070 PRINT"(HOME DOWN7 RIGHT11)NOCH(SPACE)EIN(SPA
CE)PAAR(SPACE)METER":POKEV+21,1:RETURN
1080 PRINT"(HOME DOWN7 RIGHT12)DIE(SPACE)LETZTE(S
PACE)HUERDE":POKEV+21,3:POKEV+2,150:RETURN
1090 POKEV+2,20:POKEV+4,100:POKEV+6,150:POKEV+8,
200:POKEV+10,255
1091 POKEV+21,63:RETURN
1100 RUN
1200 PRINT"(CLEAR DOWN10 RIGHT13 WHITE)READY(SPA
CE)PLAYER"PI:POKEV+17,27
1210 R1=PEEK(V+32):R2=PEEK(V+33):R3=PEEK(V+21):P
OKEV+32,0:POKEV+33,0:POKEV+21,0
1220 WAIT56320,16,16:POKEV+21,R3:POKEV+32,R1:POK
EV+33,R2:RETURN

```

Die Fortsetzung der tollen Sprite-, Sound- & Grafikhilfe

Extended UMH

In Heft 9/86 haben wir eine Sprite- und Soundhilfe für den C-64 veröffentlicht, die bereits begeisterte Reaktionen unter unseren Lesern hervorgerufen hat.

Diesmal hat der Autor 18 neue Befehle verfaßt, die wir zusammen mit dem kompletten UMH-System der Ausgabe 9 an dieser Stelle veröffentlichen.

Die neuen Befehle lauten:

1. VIC-Befehle

- SYS 51241 schaltet den Bildschirm aus
- SYS 51232 schaltet den Bildschirm an
- SYS 51672 schaltet auf 38 Spalten
- SYS 51663 schaltet auf 40 Spalten
- sys 51681 schaltet auf 24 Zeilen
- SYS 51690 schaltet auf 25 Zeilen
- SYS 51699,x Auswahl der Sprites, die eine x-Koordinate haben sollen, die größer als 255 ist

2. Extended-Colour-Befehle

- SYS 51250 schaltet den Extended-Colour-Mode ein
- SYS 51259 schaltet den Extended-Colour-Mode aus
- SYS 51268,x,y Ex.-Col. Farbe 1, Ex.-Col. Farbe 2

3. Frame-Befehle

- SYS 51287,... bis zu 8 Parameter, die die Rahmenfarben angeben
- SYS 51387,x Verzögerung für Farbwechsel (0-255) (Waitkey)
- SYS 51487 aktiviert die Waitkey-Rahmenfarbe-Routine (wartet auf eine Taste und erzeugt hübsche farbige Balken auf dem Rahmen)
- SYS 51443,x Verzögerung für den Farbwechsel der Waitfire-Routine (0-255)
- SYS 51405 aktiviert die Waitfire-Routine (wartet auf FIRE an Port 2)

4. Joystick-Befehle

- SYS 51462,x,y X-Wert für Spritebewegung, y-Wert für Spritebewegung (Port 1 - Sprite 1)
- SYS 51481,x,y X-Wert für Spritebewegung, y-Wert für Spritebewegung (Port 2 - Sprite 2)
- SYS 51500 aktiviert die Joystickroutine, welche beide Ports einmal abfragt, die Sprites gegebenenfalls bewegt, und wieder an Basic übergibt

Somit kommt das UMH-System nun auf stattliche 48 Befehle. Das EX-UMH-System ist natürlich voll aufwärtskompatibel zu UMH, was bedeutet, daß die mit UMH geschriebenen Programme auch mit EX-UMH laufen.

Nachtrag aus Heft 10:

- SYS 50334,x1,41,x8,y8
Geschwindigkeit von Sprite 1-8 in x- und y-Vektoren aufgeteilt.

Die anderen Befehle entnehmen Sie bitte dem Heft 9!

C-64 LISTING + CHECKSUMMEN (C V1.0)

0	REM***** PARAVISION COMPANY *****	<57>	142,17,212,32,253,174,32,158	<238>
1	REM* P R E S E N T S: *	<98>	24 DATA183,142,16,212,32,253,174,32,158,183,142,	<30>
2	REM* THE GREAT *	<201>	19,212,32,253,174,32,158,183	
3	REM* ----- *	<50>	25 DATA142,20,212,32,253,174,32,158,183,169,0,14	<241>
4	REM* - EXTENDED UMH-SYSTEM - *	<131>	1,18,212,142,18,212,96,32	
5	REM* ----- *	<52>	26 DATA253,174,32,158,183,134,211,32,253,174,32,	<134>
6	REM*WRITTEN IN 1986 BY: ULRICH MUEHL*	<99>	158,183,142,214,0,32,253,174	
7	REM* COPYRIGHT 1986 BY PARAVISION *	<94>	27 DATA32,158,183,142,134,2,96,32,253,174,32,158	<149>
8	REM*DR. BEUERMANNSTR. 23,3440 ESCHWEGE*	<14>	,183,134,2,160,255,165,2,153	
9	REM*****	<202>	28 DATA255,3,153,254,4,153,253,5,153,252,6,136,1	<206>
10	PRINT" (CLEAR)GENERATING (SPACE)M/CODE..."	<3>	52,208,238,96,32,253,174,32	
11	FOR X=49152 TO 50716:READ DT:POKE X,DT:NEXT X	<209>	29 DATA158,183,134,2,32,253,174,32,158,183,160,2	<80>
12	DATA32,253,174,32,158,183,142,32,208,32,253,1	<183>	55,185,255,3,197,2,208,4,138	
13	DATA32,253,174,32,158,183,142,1,212,32,253,17	<98>	30 DATA153,255,3,185,254,4,197,2,208,4,138,153,2	<92>
14	DATA253,174,32,158,183,142,3,212,32,253,174,3	<139>	54,4,185,253,5,197,2,208,4	
15	DATA174,32,158,183,142,5,212,32,253,174,32,15	<110>	31 DATA138,153,253,5,185,252,6,197,2,208,4,138,1	<32>
16	DATA32,158,183,169,0,141,4,212,142,4,212,96,3	<245>	53,252,6,136,152,208,208,96	
17	DATA160,255,165,2,153,255,215,153,254,216,153	<139>	32 DATA32,253,174,32,158,183,134,2,32,253,174,32	<32>
18	DATA208,238,96,32,253,174,32,158,183,142,24,2	<240>	,158,183,160,255,185,255,3	
19	DATA142,8,212,32,253,174,32,158,183,142,7,212	<120>	33 DATA197,2,208,4,138,153,255,215,185,254,4,197	<233>
20	DATA10,212,32,253,174,32,158,183,142,9,212,32	<101>	,2,208,4,138,153,254,216,185	
21	DATA212,32,253,174,32,158,183,142,13,212,32,2	<240>	34 DATA253,5,197,2,208,4,138,153,253,217,185,252	<180>
22	DATA11,212,142,11,212,96,32,253,174,32,158,18	<131>	,6,197,2,208,4,138,153,252	
23	DATA158,183,142,14,212,32,253,174,32,158,183,		35 DATA218,136,152,208,208,96,32,253,174,32,158,	<46>
			183,142,1,212,32,253,174,32	
			36 DATA158,183,142,0,212,32,253,174,32,158,183,1	<69>
			69,0,141,4,212,142,4,212,96	
			37 DATA32,253,174,32,158,183,142,8,212,32,253,17	<226>
			4,32,158,183,142,7,212,32	
			38 DATA253,174,32,158,183,169,0,141,11,212,142,1	<228>
			1,212,96,32,253,174,32,158	
			39 DATA183,142,15,212,32,253,174,32,158,183,142,	<70>
			14,212,32,253,174,32,158,183	
			40 DATA169,0,141,18,212,142,18,212,96,32,253,174	<79>
			,32,138,173,32,247,183,165	
			41 DATA20,141,0,212,165,21,141,1,212,32,253,174,	<81>
			32,138,173,32,247,183,234	

programme

42 DATA165,20,141,2,212,165,21,141,3,212,96,32,2,53,174,32,138,173,32,247,183	<69>	85 DATA253,174,32,158,183,142,20,199,32,253,174,32,158,183,142,5,199,32,253	<127>
43 DATA165,20,141,7,212,165,21,141,8,212,32,253,174,32,138,173,32,247,183	<36>	86 DATA174,32,158,183,142,21,199,32,253,174,32,158,183,142,6,199,32,253,174	<64>
44 DATA165,20,141,9,212,165,21,141,10,212,96,32,253,174,32,138,173,32,247	<153>	87 DATA32,158,183,142,22,199,32,253,174,32,158,183,142,7,199,32,253,174,32	<23>
45 DATA183,165,20,141,14,212,165,21,141,15,212,3,2,253,174,32,138,173,32,247	<237>	88 DATA158,183,142,23,199,96,120,169,100,141,20,3,169,196,141,21,3,169,5,141	<186>
46 DATA183,165,20,141,16,212,165,21,141,17,212,96,32,253,174,32,138,173,32	<49>	89 DATA255,197,88,96,120,169,49,141,20,3,169,234,141,21,3,88,96,32,253,174	<148>
47 DATA247,183,165,20,141,21,212,165,21,141,22,212,96,169,0,133,2,32,253,174	<172>	90 DATA32,158,183,142,37,208,32,253,174,32,158,183,142,38,208,96,32,253,174	<29>
48 DATA32,158,183,138,10,10,10,10,141,2,0,32,253,174,32,158,183,224,0,240	<159>	91 DATA32,158,183,142,21,208,96,44,50,53,49,32,253,174,32,158,183,142,28,208	<127>
49 DATA7,165,2,24,105,1,133,2,32,253,174,32,158,183,224,0,240,7,165,2,24,105	<172>	92 DATA96,32,253,174,32,158,183,142,29,208,32,253,174,32,158,183,142,23,208	<189>
50 DATA2,133,2,32,253,174,32,158,183,224,0,240,7,165,2,24,105,4,133,2,173	<200>	93 DATA96	<244>
51 DATA2,0,141,23,212,96,169,0,133,2,32,253,174,32,158,183,224,0,240,7,165	<167>	94 FOR X=51232 TO 51708:READ DT:POKE X,DT:NEXT X	<18>
52 DATA2,24,105,64,133,2,32,253,174,32,158,183,224,0,240,7,165,2,24,105,32	<29>	95 PRINT"(CLEAR LIG.BLUE DOWN RIGHT)*** <u>SPACE</u> THE <u>SPACE</u> EXTENDED <u>SPACE</u> UM-H-SYSTEM <u>SPACE</u> ***"	<188>
53 DATA133,2,32,253,174,32,158,183,224,0,240,7,165,2,24,105,16,133,2,32,253	<208>	96 PRINT"(DOWN RIGHT)38911 <u>SPACE</u> BASIC <u>SPACE</u> BYTES <u>SPACE</u> FREE"	<181>
54 DATA174,32,158,183,138,24,101,2,141,24,212,96,32,253,174,32,158,183,142	<253>	97 PRINT"(DOWN RIGHT) (C) 1986 <u>SPACE</u> PARAVISION <u>SPACE</u> COMPANY"	<71>
55 DATA0,208,32,253,174,32,158,183,142,1,208,32,253,174,32,158,183,142,2,208	<124>	98 DATA173,17,208,9,16,141,17,208,96,173,17,208,41,239,141,17,208,96,173,17	<4>
56 DATA32,253,174,32,158,183,142,3,208,32,253,174,32,158,183,142,4,208,32	<151>	99 DATA208,9,64,141,17,208,96,173,17,208,41,191,141,17,208,96,32,253,174,32	<25>
57 DATA253,174,32,158,183,142,5,208,32,253,174,32,158,183,142,6,208,32,253	<144>	100 DATA158,183,142,34,208,32,253,174,32,158,183,142,35,208,96,32,253,174,32	<101>
58 DATA174,32,158,183,142,7,208,32,253,174,32,158,183,142,8,208,32,253,174	<201>	101 DATA158,183,142,197,200,32,253,174,32,158,183,142,198,200,32,253,174,32	<136>
59 DATA32,158,183,142,9,208,32,253,174,32,158,183,142,10,208,32,253,174,32	<92>	102 DATA158,183,142,199,200,32,253,174,32,158,183,142,200,200,32,253,174,32	<144>
60 DATA158,183,142,11,208,32,253,174,32,158,183,142,12,208,32,253,174,32,158	<91>	103 DATA158,183,142,201,200,32,253,174,32,158,183,142,202,200,32,253,174,32	<252>
61 DATA183,142,13,208,32,253,174,32,158,183,142,14,208,32,253,174,32,158,183	<131>	104 DATA158,183,142,203,200,32,253,174,32,158,183,142,204,200,96,162,0,189	<25>
62 DATA142,15,208,96,32,253,174,32,158,183,142,3,9,208,32,253,174,32,158,183	<162>	105 DATA197,200,141,32,208,232,224,8,208,2,162,0,160,47,136,208,253,165,203	<145>
63 DATA142,40,208,32,253,174,32,158,183,142,41,208,32,253,174,32,158,183,142	<217>	106 DATA201,64,240,232,96,32,253,174,32,158,183,142,176,200,96,2,5,7,6,10,13	<3>
64 DATA42,208,32,253,174,32,158,183,142,43,208,3,2,253,174,32,158,183,142,44	<178>	107 DATA1,14,169,224,141,2,220,162,0,189,197,200,141,32,208,232,224,8,208,2	<105>
65 DATA208,32,253,174,32,158,183,142,45,208,32,253,174,32,158,183,142,46,208	<36>	108 DATA162,0,160,65,136,208,253,173,0,220,41,16,208,231,169,255,141,2,220	<206>
66 DATA96,32,253,174,32,158,183,142,248,7,32,253,174,32,158,183,142,249,7	<114>	109 DATA96,32,253,174,32,158,183,142,226,200,96,80,65,0,0,0,0,0,208,105,32	<76>
67 DATA32,253,174,32,158,183,142,250,7,32,253,174,32,158,183,142,251,7,32	<230>	110 DATA253,174,32,158,183,142,0,202,32,253,174,32,158,183,142,1,202,96,32	<118>
68 DATA253,174,32,158,183,142,252,7,32,253,174,3,2,158,183,142,253,7,32,253	<225>	111 DATA253,174,32,158,183,142,2,202,32,253,174,32,158,183,142,3,202,96,169	<199>
69 DATA174,32,158,183,142,254,7,32,253,174,32,158,183,142,255,7,96,162,16	<26>	112 DATA224,141,2,220,173,1,220,41,1,208,10,173,1,208,56,237,1,202,141,1,208	<211>
70 DATA189,255,207,24,125,255,199,157,255,207,20,2,138,208,242,206,255,197	<65>	113 DATA173,1,220,41,2,208,10,173,1,202,24,109,1,208,141,1,208,173,1,220,41	<165>
71 DATA173,255,197,208,31,169,5,141,255,197,162,8,189,247,7,221,15,199,208	<242>	114 DATA4,208,10,173,0,208,56,237,0,202,141,0,20,8,173,1,220,41,8,208,10,173	<175>
72 DATA9,189,255,198,157,247,7,76,151,196,254,24,7,7,202,138,208,232,76,49	<138>	115 DATA0,202,24,109,0,208,141,0,208,169,0,133,1,98,173,0,220,41,1,208,10,173	<131>
73 DATA234,32,253,174,32,158,183,142,0,200,32,253,174,32,158,183,142,1,200	<45>	116 DATA3,208,56,237,3,202,141,3,208,173,0,220,4,1,2,208,10,173,3,202,24,109	<177>
74 DATA32,253,174,32,158,183,142,2,200,32,253,174,32,158,183,142,3,200,32	<81>	117 DATA3,208,141,3,208,173,0,220,41,4,208,10,17,3,2,208,56,237,2,202,141,2	<43>
75 DATA253,174,32,158,183,142,4,200,32,253,174,3,2,158,183,142,5,200,32,253	<128>	118 DATA208,173,0,220,41,8,208,10,173,2,202,24,1,09,2,208,141,2,208,169,255	<54>
76 DATA174,32,158,183,142,6,200,32,253,174,32,158,183,142,7,200,32,253,174	<1>	119 DATA141,2,220,96,162,24,169,0,153,255,211,20,2,138,208,247,96,173,22,208	<167>
77 DATA32,158,183,142,8,200,32,253,174,32,158,183,142,9,200,32,253,174,32	<191>	120 DATA9,8,141,22,208,96,173,22,208,41,247,141,22,208,96,173,17,208,41,247	<6>
78 DATA158,183,142,10,200,32,253,174,32,158,183,142,11,200,32,253,174,32,158	<246>	121 DATA141,17,208,96,173,17,208,9,8,141,17,208,96,32,253,174,32,158,183,142	<121>
79 DATA183,142,12,200,32,253,174,32,158,183,142,13,200,32,253,174,32,158,183	<102>	122 DATA16,208,96	<208>
80 DATA142,14,200,32,253,174,32,158,183,142,15,200,96,32,253,174,32,158,183	<217>	0 REM ***** EX-UHM-DEMO-LOADER *****	<76>
81 DATA142,0,199,32,253,174,32,158,183,142,16,19,9,32,253,174,32,158,183,142	<124>	1 PRINT"(CLEAR YELLOW RIGHT)PARAVISION <u>SPACE</u> COMPANY <u>SPACE</u> PROUDLY <u>SPACE</u> PRESENTS:";A\$=CHR\$(34)	<11>
82 DATA1,199,32,253,174,32,158,183,142,17,199,32,253,174,32,158,183,142,2	<194>	2 PRINTTAB(12)"(GREEN)ULRICH <u>SPACE</u> MUEHL"5"	<193>
83 DATA199,32,253,174,32,158,183,142,18,199,32,253,174,32,158,183,142,3,199	<15>	3 PRINTTAB(7)"(DOWN RED)THE <u>SPACE</u> EXTENDED-UMHBY <u>SPACE</u> DEMO"	<204>
84 DATA32,253,174,32,158,183,142,19,199,32,253,174,32,158,183,142,4,199,32	<214>	4 PRINT"(WHITE)LOADING...(BLACK DOWN)";POKE 532,81,0:POKE 53280,0	<121>
		5 PRINT"LOAD"AS"VEC"AS",8"	<215>
		6 PRINT"(DOWN)RUN"	<13>
		7 PRINT"(DOWN)LOAD"AS"SPRITESET"AS",8"	<217>
		8 PRINT"(DOWN)RUN"	<15>
		9 PRINT"(UP)";POKE631,13:POKE632,13:POKE633,13	

programme

```

:POKE634,13
10 POKE 198,4
0 REM VEKTOREN AENDERN : BASIC-START
1 POKE16385,0:POKE 16386,0:POKE43,2:POKE44,64:NE
W
0 FOR X=2048 TO 3199:READ DT:POKE X,DT:NEXT X
1 A$=CHR$(34):PRINT" (CLEAR)"
2 PRINT"(DOHN2)LOAD"A$"DEMO-MAIN"A$,8"
3 PRINT"(DOHN4)RUN"
4 PRINT"(HOHE)":POKE 631,13:POKE 632,13:POKE 633
,13:POKE 198,3
100 DATA 0, 0, 11, 0, 0, 52, 0, 0, 66
101 DATA 0, 0, 149, 0, 0, 104, 0, 0, 31
102 DATA 0, 0, 0, 252, 1, 255, 255, 192, 64
103 DATA 195, 224, 255, 192, 48, 32, 192, 48, 12
7
104 DATA 192, 48, 7, 192, 48, 7, 195, 224, 7
105 DATA 255, 192, 3, 252, 0, 3, 192, 0, 3
106 DATA 192, 0, 1, 192, 0, 1, 192, 0, 1
107 DATA 0
110 DATA 0, 0, 7, 0, 0, 56, 0, 0, 69
111 DATA 0, 0, 66, 0, 0, 57, 0, 0, 7
112 DATA 0, 0, 0, 252, 1, 255, 255, 192, 64
113 DATA 195, 224, 255, 192, 48, 32, 192, 48, 12
7
114 DATA 192, 48, 7, 192, 48, 7, 195, 224, 7
115 DATA 255, 192, 3, 252, 0, 3, 192, 0, 3
116 DATA 192, 0, 1, 192, 0, 1, 192, 0, 1
117 DATA 0
120 DATA 0, 0, 3, 0, 0, 4, 0, 0, 58
121 DATA 0, 0, 65, 0, 0, 56, 0, 0, 7
122 DATA 0, 0, 0, 252, 1, 255, 255, 192, 64
123 DATA 195, 224, 255, 192, 48, 32, 192, 48, 12
7
124 DATA 192, 48, 7, 192, 48, 7, 195, 224, 7
125 DATA 255, 192, 3, 252, 0, 3, 192, 0, 3
126 DATA 192, 0, 1, 192, 0, 1, 192, 0, 1
127 DATA 0
130 DATA 192, 0, 0, 60, 0, 0, 162, 0, 0
131 DATA 68, 0, 0, 152, 0, 0, 224, 0, 0
132 DATA 0, 0, 0, 255, 128, 63, 2, 3, 255
133 DATA 255, 7, 192, 4, 14, 0, 254, 12, 0
134 DATA 160, 12, 0, 160, 12, 0, 160, 12, 0
135 DATA 64, 12, 0, 64, 12, 0, 64, 14, 0
136 DATA 128, 7, 192, 128, 3, 255, 128, 0, 63
137 DATA 0
140 DATA 128, 0, 0, 112, 0, 0, 136, 0, 0
141 DATA 40, 0, 0, 80, 0, 0, 224, 0, 0
142 DATA 0, 0, 0, 255, 128, 63, 2, 3, 255
143 DATA 255, 7, 192, 4, 14, 0, 254, 12, 0
144 DATA 160, 12, 0, 160, 12, 0, 160, 12, 0
145 DATA 64, 12, 0, 64, 12, 0, 64, 14, 0
146 DATA 128, 7, 192, 128, 3, 255, 128, 0, 63
147 DATA 0
150 DATA 128, 0, 0, 112, 0, 0, 76, 0, 0
151 DATA 162, 0, 0, 60, 0, 0, 192, 0, 0
152 DATA 0, 0, 0, 255, 128, 63, 2, 3, 255
153 DATA 255, 7, 192, 4, 14, 0, 254, 12, 0
154 DATA 160, 12, 0, 160, 12, 0, 160, 12, 0
155 DATA 64, 12, 0, 64, 12, 0, 64, 14, 0
156 DATA 128, 7, 192, 128, 3, 255, 128, 0, 63
157 DATA 0
160 DATA 255, 255, 255, 128, 0, 0, 189, 198, 115
161 DATA 153, 173, 107, 153, 173, 107, 153, 205,
107
162 DATA 153, 166, 107, 128, 0, 0, 255, 255, 255
163 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
164 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
165 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
166 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
167 DATA 0
170 DATA 255, 255, 255, 0, 0, 0, 48, 107, 185
171 DATA 96, 107, 53, 103, 107, 181, 96, 107, 57
172 DATA 48, 51, 180, 0, 0, 0, 255, 255, 255
173 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
174 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
175 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
176 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
177 DATA 0
180 DATA 255, 252, 0, 0, 4, 0, 140, 100, 0
181 DATA 154, 196, 0, 158, 228, 0, 154, 212, 0
182 DATA 218, 100, 0, 0, 4, 0, 255, 252, 0
183 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
184 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
185 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
<7>
<162>
<236>
<211>
<125>
<130>
<1>
<10>
<211>
<3>
<1>
<25>
<103>
<59>
<44>
<4>
<78>
<175>
<59>
<35>
<113>
<69>
<54>
<14>
<88>
<95>
<48>
<45>
<123>
<79>
<64>
<24>
<98>
<245>
<5>
<83>
<49>
<87>
<77>
<218>
<108>
<127>
<166>
<93>
<59>
<97>
<87>
<228>
<118>
<62>
<55>
<103>
<69>
<107>
<97>
<238>
<128>
<236>
<71>
<41>
<198>
<199>
<200>
<201>
<138>
<228>
<232>
<47>
<208>
<209>
<210>
<211>
<148>
<25>
<181>
<36>
<218>
<219>
<220>
186 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0
187 DATA 0
190 DATA 0, 0, 0, 1, 128, 0, 1, 205, 0
191 DATA 1, 173, 0, 1, 207, 0, 0, 1, 0
192 DATA 0, 14, 0, 0, 0, 0, 0, 0
193 DATA 0, 0, 0, 96, 192, 24, 96, 224, 56
194 DATA 96, 240, 120, 96, 216, 216, 96, 205, 15
2
195 DATA 96, 199, 24, 96, 194, 24, 49, 128, 24
196 DATA 49, 128, 27, 31, 0, 14, 14, 0, 4
197 DATA 0
200 DATA 0, 0, 0, 126, 0, 1, 255, 128
201 DATA 7, 255, 224, 15, 255, 240, 15, 255, 240
202 DATA 31, 255, 248, 31, 255, 248, 63, 255, 25
2
203 DATA 63, 255, 252, 63, 255, 252, 63, 255, 25
2
204 DATA 63, 255, 252, 31, 255, 248, 31, 255, 24
8
205 DATA 15, 255, 240, 15, 255, 240, 7, 255, 224
206 DATA 1, 255, 128, 0, 126, 0, 0, 0, 0
207 DATA 0
210 DATA 0, 0, 0, 126, 0, 1, 231, 128
211 DATA 7, 231, 224, 15, 231, 240, 15, 231, 240
212 DATA 31, 231, 248, 31, 231, 248, 63, 231, 25
2
213 DATA 63, 231, 252, 32, 0, 4, 63, 231, 252
214 DATA 63, 231, 252, 31, 231, 248, 31, 231, 24
8
215 DATA 15, 231, 240, 15, 231, 240, 7, 231, 224
216 DATA 1, 231, 128, 0, 126, 0, 0, 0, 0
217 DATA 0
220 DATA 0, 0, 0, 126, 0, 1, 195, 128
221 DATA 7, 195, 224, 15, 195, 240, 15, 195, 240
222 DATA 31, 195, 248, 31, 195, 248, 63, 195, 25
2
223 DATA 32, 0, 4, 32, 0, 4, 32, 0, 4
224 DATA 63, 195, 252, 31, 195, 248, 31, 195, 24
8
225 DATA 15, 195, 240, 15, 195, 240, 7, 195, 224
226 DATA 1, 195, 128, 0, 126, 0, 0, 0, 0
227 DATA 0
230 DATA 0, 0, 0, 126, 0, 1, 129, 128
231 DATA 7, 129, 224, 15, 129, 240, 15, 129, 240
232 DATA 31, 129, 248, 31, 129, 248, 32, 0, 4
233 DATA 32, 0, 4, 32, 0, 4, 32, 0, 4
234 DATA 32, 0, 4, 31, 129, 248, 31, 129, 248
235 DATA 15, 129, 240, 15, 129, 240, 7, 129, 224
236 DATA 1, 129, 128, 0, 126, 0, 0, 0, 0
237 DATA 0
240 DATA 0, 0, 0, 126, 0, 1, 129, 128
241 DATA 7, 0, 224, 15, 0, 240, 15, 0, 240
242 DATA 31, 0, 248, 16, 0, 8, 32, 0, 4
243 DATA 32, 0, 4, 32, 0, 4, 32, 0, 4
244 DATA 32, 0, 4, 16, 0, 8, 31, 0, 248
245 DATA 15, 0, 240, 15, 0, 240, 7, 0, 224
246 DATA 1, 129, 128, 0, 126, 0, 0, 0, 0
247 DATA 0
250 DATA 0, 0, 0, 126, 0, 1, 129, 128
251 DATA 6, 0, 96, 14, 0, 112, 14, 0, 112
252 DATA 16, 0, 8, 16, 0, 8, 32, 0, 4
253 DATA 32, 0, 4, 32, 0, 4, 32, 0, 4
254 DATA 32, 0, 4, 16, 0, 8, 16, 0, 8
255 DATA 14, 0, 112, 14, 0, 112, 6, 0, 96
256 DATA 1, 129, 128, 0, 126, 0, 0, 0, 0
257 DATA 0
260 DATA 0, 0, 0, 126, 0, 1, 129, 128
261 DATA 6, 0, 96, 12, 0, 48, 8, 0, 16
262 DATA 16, 0, 8, 16, 0, 8, 32, 0, 4
263 DATA 32, 0, 4, 32, 0, 4, 32, 0, 4
264 DATA 32, 0, 4, 16, 0, 8, 16, 0, 8
265 DATA 8, 0, 16, 12, 0, 48, 6, 0, 96
266 DATA 1, 129, 128, 0, 126, 0, 0, 0, 0
267 DATA 0
270 DATA 0, 0, 0, 126, 0, 1, 129, 128
271 DATA 6, 0, 96, 8, 0, 16, 8, 0, 16
272 DATA 16, 0, 8, 16, 0, 8, 32, 0, 4
273 DATA 32, 0, 4, 32, 0, 4, 32, 0, 4
274 DATA 32, 0, 4, 16, 0, 8, 16, 0, 8
275 DATA 8, 0, 16, 8, 0, 16, 6, 0, 96
276 DATA 1, 129, 128, 0, 126, 0, 0, 0, 0
277 DATA 0
<221>
<158>
<57>
<189>
<111>
<160>
<225>
<47>
<73>
<168>
<86>
<195>
<141>
<54>
<143>
<33>
<208>
<178>
<234>
<161>
<89>
<43>
<91>
<243>
<184>
<188>
<164>
<106>
<61>
<59>
<63>
<206>
<244>
<198>
<121>
<240>
<189>
<67>
<253>
<78>
<243>
<208>
<131>
<98>
<99>
<79>
<227>
<156>
<253>
<218>
<141>
<75>
<202>
<87>
<32>
<27>
<8>
<229>
<152>
<171>
<213>
<100>
<43>
<9>
<18>
<239>
<162>
<118>
<223>
<110>
<53>
<206>
<28>
<249>

```

ENDE DES LISTINGS

DEMO

```

0 REM ***** <3>
1 REM * PARAVISION COMPANY PRESENTS * <100>
2 REM ***** <5>
3 REM *** THE EXTENDED-LMH-SYS DEMO *** <207>
4 REM ***** <7>
5 REM * LICENSED AT TRONIC-VERLAG * <26>
6 REM * WRITTEN BY ULRICH MUEHL * <26>
7 REM * (C)1986 BY PARAVISION COMPANY * <150>
8 REM *DR. BEUERMANSTR. 23, 344 ESCHWEGE* <233>
9 REM ***** <12>
10 SYS 49152,14,14:PRINT"(CLEAR)":SYS 49269,0 <124>
20 SYS 50479,32,34,35,37,38,38,39,39,40,40,41,41 <44>
,42,49,42,49
30 SYS 50130,7,7,0,0,0,13,6,6 <114>
40 SYS 50203,32,35,38,39,40,41,42,42 <114>
50 SYS 49985,158,55,182,55,118,110,166,110,214,1 <202>
10,168,205,123,205,212,205
60 SYS 50674,255 <65>
70 SYS 50334,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 <170>
80 SYS 50698,28,28:SYS 50624 <97>
90 PRINT"(YELLOW DOWN3 RIGHT11)PARAVISION(SPACE) <14>
COMPANY" <172>
100 PRINT"(BLACK DOWN RIGHT10)IN(SPACE)CONJUNCTI <218>
ON(SPACE)WITH:"
110 PRINT"(GREY DOWN3 RIGHT15)PRESENTS:" <103>
120 PRINT"(WHITE DOWN RIGHT4)*(SPACE)THE(SPACE)E <42>
XTENDED(SPACE)LMH-SYSTEM(SPACE)DEMO(SPACE)*"
125 PRINT"(DOWN2 RIGHT7 BLACK)(C)1986(SPACE)PARA <244>
VISION(SPACE)COMPANY"
130 PRINT"(BLUE RIGHT4 DOWN2)ALL(SPACE)CODEING,8 <217>
RAPHSICS,SDUNDS,ETC."
140 PRINT"(DOWN4 RIGHT7)---(SPACE)PRESB(SPACE)A(S <34>
PACE)KEY(SPACE)TD(SPACE)BD(SPACE)ON(SPACE)---"
150 SYS 51287,2,10,7,13,5,14,6,4:SYS 51387,212:8 <11>
YS 51360
160 SYS 49152,2,2:SYS 50674,3:PRINT"(CLEAR)" <26>
170 SYS 51287,1,15,12,11,0,11,12,15:SYS 51387,13 <206>
8
180 PRINT"(WHITE DOWN4 RIGHT17)MENUE(DOWN LEFTS <31>
SC5)"
190 PRINT"(DOWN2 RIGHTS)WELCHES(SPACE)DEMO(SPACE) <55>
)WOLLEN(SPACE)SIE(SPACE)SEHEN(SPACE)?"
200 PRINT"(YELLOW DOWN2 RIGHT10)(1)(SPACE)JOYSTI <245>
CK(SPACE)DEMO"
210 PRINT"(YELLOW DOWN RIGHT10)(2)(SPACE)EXTENDE <116>
D(SPACE)COLOUR(SPACE)DEMO"
220 PRINT"(YELLOW DOWN RIGHT10)(3)(SPACE)VIC-DEM <67>
O"
230 PRINT"(YELLOW DOWN RIGHT10)(4)(SPACE)FRAME(S <56>
PACE)DEMO"
240 SYS 51360:GET A$:IF VAL(A$)<1 OR VAL(A$)>4 T <229>
HEN 240
250 ON VAL(A$) GOTO 1000,2000,3000,4000 <114>
260 RUN <143>
1000 REM ***** <176>
1010 REM * <86>
1020 REM * JOYSTICK DEMO * <97>
1030 REM * <107>
1040 REM ***** <217>
1050 PRINT"(CLEAR)":SYS 50674,0:SYS 49152,0,0 <189>
1060 PRINT"(YELLOW)IHRE(SPACE)WAHL:(SPACE)JOYSTI <173>
CH(SPACE)DEMO"
1070 PRINT"(DOWN)BEIDE(SPACE)JOYSTICKS(SPACE)SIN <59>
D(SPACE)AKTIV"
1080 PRINT"(DOWN)'SPACE'(SPACE)ZUM(SPACE)ABBRECH <76>
EN"
1090 SYS 51462,5,5:SYS 51481,4,4 <31>
1100 SYS 49269,15:SYS 49171,0,0,0,0,0,100,17 <103>
1110 SYS 49279,0,0,0,0,0,100,17 <97>
1115 SYS 50642 <255>
1120 SYS 50203,42,42,0,0,0,0,0,0,0 <184>
1130 SYS 50479,42,49,42,49,0,0,0,0,0,0,0,0,0,0,0 <77>
,0
1140 SYS 49985,100,100,200,100,0,0,0,0,0,0,0,0,0,0,0 <59>
,0,0,0
1150 SYS 50674,3:SYS 50624 <76>
1160 SYS 51500:SYS49702,PEEK(53248)*100,0:SYS497 <220>
42,PEEK(53250)*100,0:GETA$
1170 IFA*>"(SPACE)"THEN 1160 <232>
1180 GOTO 260 <239>
2000 REM ***** <156>
2010 REM * <66>
2020 REM * EXTENDED COLOUR DEMO * <39>

```

```

2030 REM * * <86>
2040 REM ***** <196>
2050 PRINT"(CLEAR)":SYS 50674,0:SYS 49152,5,5 <143>
2060 PRINT"(YELLOW)IHRE(SPACE)WAHL:(SPACE)EXTEND <220>
ED(SPACE)COLOUR(SPACE)DEMO"
2065 PRINT"(YELLOW)'SPACE'(SPACE)ZUM(SPACE)FORTF <64>
AHREN"
2070 SYS 51250:SYS 51268,7,6 <116>
2080 PRINT"(DOWN3 RIGHT9 SHIFTSPACE22)":REM 22 8 <107>
HIFT-SPACES !!!!
2090 PRINT"(RIGHT9 RED SHIFTSPACE SE SX SI SE SN <45>
SD SE SD SHIFTSPACE SC SD SL SD SU SR SHIFTSPAC
E SM SD SE SHIFTSPACE)":REM ALLES GESHIFTET !
!!
2100 PRINT"(RIGHT9 SHIFTSPACE22)":REM 22 SHIFT-S <173>
PACES !!!!
2110 PRINT"(DOWN3 RVSON WHITE RIGHT3 SPACE3)IM(S <10>
PACE)EXTENDED(SPACE)COLOUR(SPACE)MODE(SPACE)KANN
(SPACE3)"
2120 PRINT"(RVSON WHITE RIGHT3 SPACE2)MAN(SPACE) <102>
GANZ(SPACE)VERBLUEFFENDE(SPACE)EFFEKTE(SPACE2)"
2125 PRINT"(RVSON WHITE RIGHT3 SPACE13)ERZIELEN( <161>
SPACE13)"
2130 PRINT"(DOWN3 RIGHT11 WHITE SP SA SR SA SY S <102>
I SS SI SO SH SHIFTSPACE SC SO SM SP SA SN SY)":
REM GESHIFTET
2140 SYS 51268,1,6:FOR WT=0 TO 50:NEXT WT <234>
2150 SYS 51268,15,6:FOR WT=0 TO 50:NEXT WT <235>
2160 SYS 51268,12,6:FOR WT=0 TO 50:NEXT WT <218>
2170 SYS 51268,11,6:FOR WT=0 TO 50:NEXT WT <219>
2180 SYS 51268,0,6:FOR WT=0 TO 50:NEXT WT <10>
2190 GET A$:IF A$<>"(SPACE)" THEN 2140 <233>
2195 FOR X=0 TO 10:GET A$:NEXT X <253>
2200 SYS 51268,7,6:FOR WT=0 TO 50:NEXT WT <86>
2210 SYS 51268,7,14:FOR WT=0 TO 50:NEXT WT <34>
2220 SYS 51268,7,13:FOR WT=0 TO 50:NEXT WT <33>
2230 SYS 51268,7,5:FOR WT=0 TO 50:NEXT WT <106>
2240 SYS 51268,7,12:FOR WT=0 TO 50:NEXT WT <42>
2250 GET A$:IF A$<>"(SPACE)" THEN 2200 <244>
2255 FOR X=0 TO 10:GET A$:NEXT X <57>
2260 SYS 51268,1,6:FOR WT=0 TO 50:NEXT WT <98>
2270 SYS 51268,15,14:FOR WT=0 TO 50:NEXT WT <84>
2280 SYS 51268,12,13:FOR WT=0 TO 50:NEXT WT <59>
2290 SYS 51268,11,5:FOR WT=0 TO 50:NEXT WT <72>
2300 SYS 51268,0,12:FOR WT=0 TO 50:NEXT WT <46>
2310 GET A$:IF A$<>"(SPACE)" THEN 2260 <145>
2315 FOR X=0 TO 10:GET A$:NEXT X <118>
2320 SYS 51268,1,6:SYS 49152,13,13:FOR WT=0 TO 4 <31>
0:NEXT WT
2330 SYS 51268,15,14:SYS 49152,7,7:FOR WT=0 TO 4 <233>
0:NEXT WT
2340 SYS 51268,12,13:SYS 49152,8,8:FOR WT=0 TO 4 <248>
0:NEXT WT
2350 SYS 51268,11,5:SYS 49152,2,2:FOR WT=0 TO 40 <105>
:NEXT WT
2360 SYS 51268,0,12:SYS 49152,4,4:FOR WT=0 TO 40 <163>
:NEXT WT
2370 GET A$:IF A$<>"(SPACE)" THEN 2320 <156>
2375 FOR X=0 TO 10:GET A$:NEXT X <178>
2380 SYS 51259 <23>
2390 GOTO 260 <174>
3000 REM ***** <136>
3010 REM * * <46>
3020 REM * VIC-DEMO * <218>
3030 REM * * <66>
3040 REM ***** <176>
3050 PRINT"(CLEAR)":SYS 50674,0:SYS 49152,6,14 <68>
3060 PRINT"(WHITE)IHRE(SPACE)WAHL(SPACE):(SPACE) <98>
VIC(SPACE)DEMO"
3070 PRINT"(DOWN)DER(SPACE)VIC(SPACE)BIETET(SPAC <47>
E)3(SPACE)MOEGLICHKEITEN,"
3080 PRINT"(DAS(SPACE)BILDFORMAT(SPACE)ZU(SPACE)B <183>
EEINFLUSSEN..."
3090 PRINT"(DOWN)JEWELIS(SPACE)'SPACE'(SPACE)DRU <109>
ECKEN..."
3100 FOR X=0 TO 10:GET A$:NEXT X <138>
3105 GET A$:IF A$<>"(SPACE)" THEN 3105 <163>
3110 SYS 51241 <191>
3120 GET A$:IF A$<>"(SPACE)" THEN 3120 <125>
3130 FOR X=0 TO 10:GET A$:NEXT X <168>
3140 SYS 51232 <222>
3150 GET A$:IF A$<>"(SPACE)" THEN 3150 <203>
3160 FOR X=0 TO 10:GET A$:NEXT X <198>
3170 SYS 51672 <32>
3180 GET A$:IF A$<>"(SPACE)" THEN 3180 <25>
3190 FOR X=0 TO 10:GET A$:NEXT X <228>

```



```

3200 SYS 51663
3210 GET A$: IF A$<>"{SPACE}" THEN 3210
3220 FOR X=0 TO 10:GET A$:NEXT X
3230 SYS 51681
3240 GET A$: IF A$<>"{SPACE}" THEN 3240
3250 FOR X=0 TO 10:GET A$:NEXT X
3260 SYS 51690
3270 GET A$: IF A$<>"{SPACE}" THEN 3270
3280 FOR X=0 TO 10:GET A$:NEXT X
3290 GOTO 260
4000 REM *****
4010 REM *
4020 REM * FRAME DEMO *
4030 REM *
4040 REM *****
4050 PRINT "{CLEAR}":SYS 50674,0:SYS 49152,3,3
4060 PRINT "{WHITE}IHRE{SPACE}WAHL: {SPACE}FRAME{S
PACE}DEMO"
4070 PRINT "{DOHN}MIT{SPACE}DEN{SPACE}FRAME-ROUTI
NEN{SPACE}KANN{SPACE}MAN{SPACE}SEHR"
4080 PRINT "SCHOENE{SPACE}EFFEKTE{SPACE}MIT{SPACE
}DEN{SPACE}SONST{SPACE}SO{SPACE}TOTEN"
<63>
<214>
<2>
<91>
<36>
<32>
<120>
<114>
<62>
<53>
<116>
<26>
<162>
<46>
<156>
<10>
<224>
<205>
<34>
4090 PRINT"RAHMEN{SPACE}ERZEUGEN..."
4100 PRINT"{DOHN}>PRESS{SPACE}A{SPACE}KEY"
4110 SYS 51287,5,13,4,14,6,11,12,15:SYS 51387,51
:SYS 51360
4120 FOR X=0 TO 10:GET A$:NEXT X
4130 SYS 51287,1,15,12,11,0,11,12,15:SYS 51387,5
0:SYS 51360
4140 FOR X=0 TO 10:GET A$:NEXT X
4150 SYS 51387,52:SYS 51360
4160 FOR X=0 TO 10:GET A$:NEXT X
4170 SYS 51287,1,7,8,2,4,6,14,13:SYS 51387,148:5
YS 51360
4180 FOR X=0 TO 10:GET A$:NEXT X
4190 SYS 51387,144:SYS 51360
4200 FOR X=0 TO 10:GET A$:NEXT X
4210 SYS 51287,5,5,13,13,1,1,3,3:SYS 51387,226:5
YS 51360
4220 FOR X=0 TO 10:GET A$:NEXT X
4230 SYS 51287,2,9,7,1,13,5,6,4:SYS 51387,212:SY
S 51360
4240 FOR X=0 TO 10:GET A$:NEXT X
4250 GOTO 260
<207>
<71>
<135>
<138>
<70>
<158>
<168>
<178>
<195>
<198>
<161>
<218>
<161>
<238>
<243>
<2>
<249>

```

IRQ-DIRECTORY

Inhaltsverzeichnis der Diskette ohne Programmverlust!



Das Maschinencode-Programm "IRQ-Disk" erleichtert die Arbeit mit der Floppy des C-64, indem es das Lesen des Directory und des Floppy-Fehlerkanals auf Tastendruck und ohne Programmverlust ermöglicht. Das Directory

wird beim Laden unmittelbar auf den Bildschirm gebracht, kann dabei zum Lesen kurz angehalten und zum Laden eines Programmes abgebrochen werden, mit dem Vorteil, nicht jedesmal das ganze Directory laden zu müssen.

Ansteuerung:
SYS 52914

Einschalten von IRQ-Disk - die Ansteuerung erfolgt dann durch die Funktionstasten.

SYS 52914

Einschalten von IRQ-Disk - die Ansteuerung erfolgt nun durch Commodoretaste plus Funktionstasten. Hierdurch ergibt sich eine höhere Verträglichkeit mit den Programmen, in die IRQ-Disk eingebunden werden soll.

Die Tasten sind wie folgt belegt:
(Commodore +) F 1

Directory anzeigen (nochmal F 1 = kurz anhalten, RUN/STOP = Abbruch)

(Commodore +) F 3
Fehlerkanal anzeigen

(Commodore +) F 7
Erweiterung ausschalten (vor Programmen, die den Bereich von 52914 - 53247 beschreiben)

Das MC-Programm wird als Basiclader unter dem Namen "IRQ-DISK" auf Diskette oder Kassette abgelegt und wie ein normales Basicprogramm geladen und gestartet. Nach dem Programmstart wird das MC-Programm ab 52914 in den Speicher geschrieben. Dies garantiert eine hohe Verträglichkeit mit anderen Programmen. Die oben beschriebenen Funktionen stehen nun zur Verfügung.

```

1 REM -----
2 REM          I R Q - D I S K
3 REM          (C)1986 THOMAS BEIGEL
4 REM          TELEFON 05136/84228
5 REM -----
6 REM
20 PRINT "{CLEAR DOHN SPACE RVSON SPACE4}I{SPACE}
R{SPACE}Q{SPACE2}-{SPACE2}D{SPACE}I{SPACE}S{SPAC
E}K{SPACE4}RVSOFF}"
25 PRINT "{SPACE RVSON SPACE} (C) {SPACE}1986{SPACE}
2}THOMAS{SPACE}BEIGEL{SPACE RVSOFF}"
30 PRINT "{DOHN2 RIGHT}F1{SPACE8}DIRECTORY";
32 PRINT "{SPACE4}F1=KURZ{SPACE}ANHALTEN{SPACE24}
";
34 PRINT"RUN/STOP=ABBRUCH"
40 PRINT"{RIGHT}F3{SPACE8}DISKERROR
50 PRINT"{DOHN RIGHT}F7{SPACE8}INTERRUPT{SPACE4}A
USSCHALTEN
60 PRINT"{DOHN RIGHT}SYS52914{SPACE2}EINSCHALTEN
{SPACE}DES{SPACE}FX-MODUS
65 PRINT"{DOHN RIGHT}SYS52914,{SPACE}EINSCHALTEN
{SPACE}DES{SPACE}C={SPACE}FX-MODUS
70 PRINT"{DOHN SPACE11}(ALLE{SPACE}F-TASTEN{SPAC
E}NUR{SPACE}ZUSAMMEN{SPACE14}MIT{SPACE}C=-TASTE)
75 IFPEEK(52914)=120ANDPEEK(52915)=173THENPO
80 FORF=0TO333:READD:POKE52914+F,D:S=S+D:NEXT
85 IFS<>44463THENPRINT"{DOHN RVSON SPACE}DATEN-F
EHLER!{SPACE RVSOFF}":END
90 SYS52914:PRINT"{DOHN2 RIGHT}IRQ-DISK{SPACE}AK
TIVIERT."
32000 DATA120,173,20,3,201,238,240,22,173,20,3,1
41,42,207,173,21,3,141,43,207
32001 DATA169,238,141,20,3,169,206,141,21,3,169,
0,133,2,169,44,160,0,209,122
32002 DATA208,11,32,115,0,169,1,141,245,206,76,2
36,206,169,0,141,245,206,88,96
32003 DATA165,2,201,1,240,35,169,0,240,7,173,141
,2,201,2,208,24,165,203,201,4
32004 DATA208,7,169,0,133,203,32,44,207,201,5,20
8,7,169,0,133,203,32,173,207
32005 DATA201,3,208,14,120,173,42,207,141,20,3,1
73,43,207,141,21,3,88,76,49,234
32006 DATA32,210,207,169,36,133,251,169,251,133,
187,169,0,133,188,169,1,133,183
32007 DATA169,8,133,188,169,96,133,185,32,213,24
3,165,186,32,180,255,165,185
32008 DATA32,150,255,169,0,133,144,160,3,132,251
,32,165,255,133,252,164,144,208
32009 DATA63,32,165,255,164,144,208,58,164,251,1
36,208,233,166,252,32,205,189
32010 DATA169,32,32,210,255,32,165,255,166,144,2
08,36,170,240,6,32,210,255,76
32011 DATA123,207,169,13,32,210,255,32,225,255,2
40,17,32,228,255,201,133,208
32012 DATA6,165,203,201,4,240,250,160,2,208,180,
32,66,246,32,229,207,96,32,210
32013 DATA207,169,8,133,186,32,180,255,169,111,1
33,185,32,150,255,32,165,255
32014 DATA32,210,255,201,13,208,246,32,171,255,3
2,245,207,32,229,207,96,169,1
32015 DATA141,14,220,169,1,133,204,169,1,133,2,1
69,13,32,210,255,96,169,1,141
32016 DATA14,220,169,0,133,198,133,203,133,197,1
33,2,96,162,100,160,0,136,208
32017 DATA253,202,208,248,96
ENDE DES LISTINGS
<250>
<16>
<130>
<231>
<254>
<149>
<57>
<228>
<106>
<251>
<125>
<252>
<52>
<152>
<15>
<115>
<8>
<6>
<146>
<168>
<166>
<77>
<248>
<140>
<37>
<200>
<80>
<190>
<101>
<106>
<76>
<114>
<72>
<160>
<48>
<20>
<204>
<108>

```

ASSEMBLERKURS

Teil 19

Um professionell auf dem Commodore C64 programmieren zu können, bedarf es der genauen Kenntnis der Systemadressen. Durch geschicktes Ausnutzen dieser Adressen, bleibt es dem Programmierer meist erspart, das Rad zum zweiten mal zu erfinden. Wie wir im letzten Teil schon erfahren haben, lassen sich viele Zeropage-Adressen auf einfachste Weise manipulieren oder ausnutzen. Neben der Zeropage benutzt das Betriebssystem bzw. der Basicinterpreter jedoch eine ganze Reihe weiterer wichtiger Speicherstellen, welche durch den Programmierer ausgenutzt werden können, diese wollen wir Ihnen in diesem Teil durch eine Aufstellung etwas näher vorstellen.

Adresse Hex Dezimal Belegung

\$100-\$13E	256-318	Speicherbereich für Bandfehler
\$100-\$1FF	256-511	Prozessor Stack
\$200-\$258	512-600	Basic Eingabepuffer
\$259-\$262	601-610	Tabelle der logischen Filenummern
\$263-\$26C	611-620	Tabelle der Gerätenummern
\$26D-\$276	621-630	Tabelle der Sekundäradressen
\$277-\$280	631-640	Tastaturpuffer
\$281-\$282	641-642	Startadresse des Basic-Ram
\$283-\$284	643-644	Ende des Basic-Ram
\$285	645	Timeout-Flag für seriellen IEC-Bus
\$286	646	augenblickliche Farbe
\$287	647	Farbe unter dem Cursor
\$288	648	High-Byte Video-RAM
\$289	649	Maximalgröße des Tastaturpuffers
\$28A	650	Tastaturwiederholung
\$28B	651	Zähler für Wiederholungs-Geschwindigkeit
\$28C	652	Zähler für Wiederholungs-Verzögerung
\$28D	653	Flag für SHIFT und CTRL
\$28E	654	Shift-Flag
\$28F-\$290	655-656	Zeiger für Tastatur-Decodierung
\$291	657	Flag für Shift/Commodore gesperrt
\$292	658	0 = SCROLL möglich
\$293	659	RS232 Chipkontrolle
\$294	660	RS232 Chipbefehl
\$295-\$296	661-662	Bit-Timing
\$297	663	RS232 Status
\$298	664	Anzahl der zu sendenden Bits
\$299-\$29A	665-666	RS232 Geschwindigkeit / Code
\$29B	667	RS232 Empfangszeiger
\$29C	668	RS232 Eingabezeiger
\$29D	669	RS232 Sendezeiger
\$29E	670	RS232 Ausgabezeiger
\$29F-\$2A0	671-672	Speicher für IRQ während Bandbetrieb
\$2A1	673	CIA 2 NMI-Flag
\$2A2	674	CIA 1 Timer A
\$2A3	675	CIA 1 Interruptflag
\$2A4	676	CIA 1 Flag für Timer A
\$2A5	677	Bildschirmzeile
\$2C0-\$2FE	704-766	Sprite 11
\$300-\$301	768-769	Verbindung Fehlernachricht
\$302-\$303	770-771	Verbindung Basic-Warmstart
\$304-\$305	772-773	Verbindung Basic-Befehlswerte decodiert
\$306-\$307	774-775	Verbindung Ausdruck Befehlswerte
\$308-\$309	776-777	Verbindung Anfang des neuen Basic-Code
\$30A-\$30B	778-779	Verbindung Holen Zahlenelement
\$30C	780	Zwischenspeicher für Akkumulator

\$30D	781	Zwischenspeicher für X-Register
\$30E	782	Zwischenspeicher für Y-Register
\$30F	783	Speicher SYS Statusregister
\$310-\$312	784-785	USR Sprung (\$B248)
\$314-\$315	788-789	IRQ Vektor
\$316-\$317	790-791	Break Interrupt Vektor (\$FE66)
\$318-\$319	792-793	NMI Interrupt Vektor (\$FE47)
\$31A-\$31B	794-795	Open Vektor (\$F34A)
\$31C-\$31D	796-797	Close Vektor (\$F291)
\$31E-\$31F	798-799	Set-Input Vektor (\$F20E)
\$320-\$321	800-801	Set-Output Vektor (\$F250)
\$322-\$323	802-803	Speichere I/O Vektor (\$F333)
\$324-\$325	804-805	INPUT-Vektor (\$F157)
\$326-\$327	806-807	Ausgabe Vektor (\$F1CA)
\$328-\$329	808-809	Test-STOP Vektor (\$F6ED)
\$32A-\$32B	810-811	GET-Vektor (\$F13E)
\$32C-\$32D	812-813	Vergiß I/O Vektor (\$F32F)
\$32E-\$32F	814-815	USR Vektor (\$FE66)
\$330-\$331	816-817	Verbindung LOAD (\$F4A5)
\$332-\$333	818-819	Verbindung SAVE (\$F5ED)
+33C-\$3FB	828-1019	Kassettenpuffer
\$340-\$37E	832-894	Sprite 13
\$380-\$3BE	896-958	Sprite 14
\$3C0-\$3FE	960-1022	Sprite 15

Wichtige ROM-Routinen !

Neben den Systemvariablen gibt es eine sehr große Anzahl nützlicher ROM-Routinen, welche sich vom Anwender auf einfachste Weise ausnutzen lassen. Wir wollen an dieser Stelle keine Tabelle aller Rom-Routinen bringen, denn dieses würde sich über zahlreiche Seiten erstrecken und würde immer noch keinen Anspruch auf Vollständigkeit beanspruchen können. Aus diesem Grund beschränken wir uns an dieser Stelle auf einige sehr wichtige Routinen des Systems und erklären andere Rom-Routinen nur bei deren Verwendung in einem Assemblerprogramm.

Die folgenden ROM-Routinen sind so aufgebaut, daß der letzte Befehl "RTS" einen Rücksprung in das Hauptprogramm bewirkt. Die Routinen können natürlich in jedem beliebigen Maschinenprogramm durch einen "JSR" Befehl aufgerufen bzw. aktiviert werden. In unserer Tabelle geben wir jedem Unterprogramm ein LABEL welches wir später in unserem Assemblerlistings wiederfinden werden. Dadurch prägen sich die Routinen wesentlich besser ein als die wirren Zahlenfolgen, die Übersichtlichkeit von Assemblerlistings wird dadurch zusätzlich gesteigert.

Wichtige Routinen zur Parameterübergabe

VAREAL \$AD8A Argument auswerten und auf numerisch prüfen
 NUMTYP \$AD8C auf numerisch prüfen
 STRTYP \$AD8F auf String prüfen
 VALPAR \$AD9E Argumentauswertung, beliebiger Ausdruck
 VALKLA \$AEF1 Argument in Klammern auswerten
 KLAMZU \$AEF7 Prüfen ob offene Klammer vorhanden

KLMAUF \$AEFA Prüfen ob geschlossene Klammer vorhanden
 CKOMMA \$AEFD Prüfen auf Komma
 GETBYT \$B79E Hole Byte in das X-Register
 CHRGET \$0073 Nächstes Zeichen aus Basic Text holen
 CHRGOT \$0079 Augenblickliches Zeichen aus Basic Text holen

Wichtige Routinen für die Bildschirmausgabe

CHROUT \$FFD2 Akkumulator wird als ASCII-Zeichen ausgegeben
 INTOU \$BDCD Eine Integer Zahl (X/Y-Register) wird ausgegeben
 CURPOS \$E56C Berechnet die Position des Cursors
 STROUT \$AB25 String ausgeben (Adresse in 34/35; X gleich Länge)
 STRZER \$ABLE String ausgeben (Adresse in A/X; Ende des Strings=0)

Maschinenunterprogramme unter Basic genutzt

Da Maschinsprache nach wie vor zu den schwierigsten Programmiersprachen gehört, gehen viele Programmierer dazu über, ihre Programme in Basic zu schreiben und nur zeitkritische Aufgaben durch kleine Maschincode-Routinen zu ersetzen. Diese Methode wäre relativ einfach und schnell zu programmieren, wenn es da nicht einen kleinen Haken gäbe:

Will man von Basic aus ein Maschinenprogramm aufrufen, dann geschieht das in der Regel mit dem Befehl `SYS xxxx`. Werden zur Durchführung dieses "SYS-Programmes" keine festen Werte verarbeitet, sondern ergeben sich diese erst aus dem Verlauf des Basic-Programmes, dann müssen diese sog. Parameter auf irgendeine Weise vom Maschinenprogramm übernommen werden können. Das Problem besteht jedoch darin, daß der Basic Befehl "SYS" keine Übergabe von Parametern gestattet, jedenfalls laut Handbuch!

Das man mittels "SYS" Befehl auch Parameter übergeben kann, haben wir schon am Beispiel unseres letzten Teiles kennen gelernt. Allerdings wird

hier dem Einem oder Anderem die Methode noch etwas unklar erscheinen und wird aus diesem Grund noch etwas näher erläutert. Wer sich im Aufbau und Ablauf von Basic-Programmen noch nicht auskennt, sollte sich dies erst einmal genauer zu Gemüte führen. Zum besseren Verständnis erläutere ich hier grob die Arbeitsweise.

- In Basic Programmen sind Befehle und Zeichen in codierter Form im Speicher abgelegt. Die Zeilennummer sind ab Zeilenanfang in Form von Low und High Byte gespeichert. Basic Befehle werden durch einen bestimmten Code, man spricht von Schlüsselwörtern oder Tokens, ersetzt.

- Nach dem Start durch den Befehl "RUN" holt der Basic-Interpreter Zeichen für Zeichen aus dem Basic-Text und untersucht ob es sich um einen Befehlscode oder einen String handelt. Liegt ein Befehl vor, so wird über eine Tabelle im ROM die entsprechende Befehlsadresse herausgesucht und in diese ROM-Routine verzweigt.

- Von nun an übernimmt die ROM-Routine die weitere Bearbeitung der Basic-Zeile. Das bedeutet, falls erforderlich, werden aus dem Basic-Text weitere Zeichen geholt (Parameter) und verarbeitet. Ein Beispiel wäre die PRINT-Anweisung, welche einen String oder eine Variable als Parameter besitzt.

- Damit der Basic-Interpreter jederzeit weiss an welcher Stelle er im Basic-Text stehen geblieben ist, existiert in der Zeropage eine kleine Routine (CHRGET) mit einem Zeiger in den Basic-Speicher. Diese Routine, oft als CHRGET bezeichnet, stellt eine der wichtigsten Routinen des C64 da. Sie wird immer dann angesprungen wenn ein Befehl ein Parameter aus dem Basic-Text benötigt. Nach deren Aufruf wird das nächste Zeichen aus dem Basic Text gelesen, dem Akkumulator übergeben und der Zeiger in den Basic-Text um eins erhöht.

Wir haben nun den groben Ablauf des Basic-Interpreters kennengelernt, und wissen welche Routinen wir anzapfen müssen, um neue Befehle zu implementieren oder Parameter an Maschinenprogramme zu übergeben.

Die Lösung unseres Problemes ist nämlich folgende:

Wir schreiben ein kleines Maschinenunterprogramm, welches mit einem "SYS" Befehl aufgerufen wird. Die Parameter geben wir einfach durch ein Komma getrennt, hinter dem "SYS" Befehl an. Unser kleines Maschinenprogramm holt nach dessen Aufruf einfach die nachfolgenden Zeichen aus dem Basic-Text und wertet diese als Parameter aus. Da über die CHRGET-Routine Zeichen geholt werden, wird gleichzeitig der Basic-Zeiger erhöht so das nach der Ausführung des Maschinenprogrammes auf den nächsten Basic-Befehl gezeigt wird. Das Basic-Programm wird also nicht durch das Maschinenprogramm beendet, sondern weiter ausgeführt. Auf diese geniale Weise lassen sich Parameter beliebigen Typs in beliebiger Anzahl an ein Maschinenprogramm übergeben.

Das nachfolgende Profi-Assemblerlisting zeigt noch einmal unsere neu gewonnenen Kenntnisse an einem praktischem Beispiel. Unsere kleine Maschinenroutine macht nichts weiter, als eine Zahl als ASCII-Code auf dem Bildschirm auszugeben. Das im Listing enthaltene Basic-Programm druckt mittels unserem neuen Befehl die Buchstaben "A" bis "Z" aus.

```

10 SYS 36864
15 .OPT 00
20 *= $C000
50 ;
60 KOMMA = $AEFD ;PRUEFEN AUF KOMMA
70 GETBYT = $B79E ;BYTE IN X HOLEN
80 CHROUT = $FFD2 ;ZEICHEN AUSGEBEN
90 ;
100 START JSR KOMMA ;PRUEFE AUF KOMMA
110 JSR GETBYT ;HOLE ZAHL AUS BASIC
120 TXA ;ZEICHENCODE NACH AKKUMULATOR
130 JSR CHROUT ;ZEICHEN AUSGEBEN
140 RTS
150 .END
200 REM
210 REM BASIC DEMOPROGRAMM
220 REM
230 PRINT"WELCH GEBE DAS ABC AUS: 00"
240 FOR I=65 TO 90
250 SYS 49152,I:REM ZEICHEN AUSGEBEN
260 NEXT I
READY.

READY.
```

Im nächsten Teil unseres Kurses werden wir unsere neu gewonnenen Kenntnisse anhand weiterer Beispiele verdeutlichen. Außerdem werden wir einige Kenntnisse über Sprites und deren Programmierung gewinnen.

Frank Brall

Verwendete Literatur:

6502 Microcomputer-Programmierung, Peter Heuer, Hofacker-Verlag

64 INTERN, Angershausen, Becker, Englisch, Gerits, Data Becker Buch

C64 Programmieren in Maschinsprache, Winfried Kassera, Frank Kassera, Markt & Technik

Profihandbuch, H.L.Schneider, Werner Eberl, Markt und Technik

Interruptprogrammierung auf dem Commodore 64

Artikelserie von Daniel Durstewitz und Waldemar Raaz

Interruptprogrammierung - eine von der Hardware unterstützte, viel zu selten genutzte Programmierertechnik, die Ihnen hilft, das Letzte aus Ihrem Computer herauszuholen ...

Wenn Sie sich immer schon gefragt haben, wie dieses oder jenes Profiprogramm Text und Grafik auf einmal darstellen kann, den Rahmen in mehreren Farben gleichzeitig blinken läßt, mehr als acht Sprites auf den Bildschirm »zaubern« kann oder einfach nur eine berauschende Geschwindigkeit bei der Spritesteuerung erreicht, so ist dieser Artikel genau das Richtige für Sie; denn alle diese Effekte lassen sich mit einer Programmierertechnik erzeugen, nämlich der Interruptprogrammierung. Dieses wollen wir Ihnen hier näher erläutern und möglichst ausführlich und verständlich erklären.

Zu Beginn unserer Ausführung muß vorausgeschickt werden, daß eine ernsthafte Interruptprogrammierung in vielen Punkten Kenntnisse in Maschinensprache voraussetzt, und wir uns daher primär an die Maschinensprache-programmierer unter den Lesern wenden. Aber auch diejenigen, die sich bisher »nur« mit Basic oder einer anderen höheren Programmiersprache auseinandergesetzt haben, sollten sich nicht scheuen, an dieser Stelle weiterzulesen. Zum einen sind die Beispielprogramme, die das Gesagte demonstrieren sollen, sicherlich auch für Sie interessant, und zum anderen können die Möglichkeiten der Interruptprogrammierung in Maschinensprachen vielleicht den einen oder anderen dazu bewegen, seine - sicherlich unbegründete - Hemmschwelle zu überwinden und sich mit der Programmierung in Assembler zu beschäftigen.

Zunächst einmal zur Klärung des Begriffs »Interrupt« (zu deutsch: Unterbrechung): ein Interrupt ist eine durch ein bestimmtes Ereignis ausgelöste Unterbrechung

des laufenden Programmes. Der gerade bearbeitete Befehl wird zu Ende ausgeführt, Programmzähler und Prozessorstatus werden auf den Stack gerettet, und dann verzweigt das Betriebssystem in eine sogenannte Interruptroutine. Hier prüft der Computer die Ursache des Interrupts, beseitigt Sie gegebenenfalls und fährt dann mit der Bearbeitung des aktuellen Programms fort. Dazu holt er Programmzähler und Prozessorstatus wieder vom Stack, so daß die gleiche Konstellation wiederhergestellt ist wie vor der Programmunterbrechung.

Beim C64 unterscheidet man vier verschiedene Arten von Interrupts (Reset, Break, NMI, IRQ), die im folgenden - mit den sich daraus ergebenden Möglichkeiten - ausführlich erläutert werden sollen:

1. Reset

Sobald der Pin 40 des 6510, des Prozessores im C64, kurzzeitig auf Masse gelegt wird, stoppt der Computer die Bearbeitung des laufenden Programms und führt einen Kaltstart aus. Dazu holt er sich aus den Speicherzellen 65532/65533 (hex. FFFC/FFFD) die Startadresse der Resetroutine und verzweigt dorthin. In dieser Routine wird der Computer beinahe in den Einschaltzustand zurückgesetzt - mit dem kleinen Unterschied, daß zwar alle Register in eine vordefinierte Ausgangskonstellation gebracht werden, der RAM-Speicherinhalt (bis auf die ersten Pages bis zum Bildschirmspeicher) sonst aber erhalten bleibt. So bleiben Maschinenprogramme völlig unangetastet und selbst Basicprogramme können mit Hilfe einer Re-New-Routine (siehe zum Beispiel: CPU August '85) gerettet werden. Für den Anwender gibt es drei Möglichkeiten, einen Reset auszulösen - streng genommen sogar vier, doch die Möglichkeit, softwaremäßig einen Reset »auszulösen«, indem man mit SYS64738 (bzw. JMPFCE2) ein-

fach in die Resetroutine im ROM einspringt, sei hier nur am Rande erwähnt:

- am Expansionport liegt Reset auf PIN C.
- am Seriellen Port ist dies PIN 6.
- am Userport ist es PIN 3.

Um nun einen Reset auszulösen, verbinden Sie einem der genannten Pins kurzzeitig leitend mit Masse (die Massepins sind in den Belegungsplänen im Handbuch ab Seite 141 mit GND (Ground) gekennzeichnet). Daraufhin wird der Computer in den Einschaltzustand versetzt, was bei abgestürzten Maschinenprogrammen oft die einzige Möglichkeit ist, den Computer ohne Programmverlust wieder »zum Leben zu erwecken«. Falls Sie sich einen Reset-taster selbst bauen wollen, so ist das Resetsignal am Userport unbedingt den anderen Möglichkeiten vorzuziehen: am Expansionport stellt sich das Problem, eine passende Platine zu besorgen, und das Resetsignal am Seriellen Port kann bei neueren C64 nicht mehr dazu benutzt werden, dem Rechner einen Reset mitzuteilen, sondern liefert nur noch Resetimpulse aus dem Computer.

Obwohl man den Reset softwaremäßig nicht abschalten kann, gibt es dennoch einen Trick, Programme »resetfest« zu machen und so vor unerlaubtem Kopieren zu schützen (professionelle Programme machen davon fast immer Gebrauch): nach einem Reset testet der Rechner nämlich auf die Zeichenfolge »CBM80« in den Speicherzellen 32772 bis 32776 (hex. \$8004 bis \$8008). Existiert diese Zeichenfolge in den besagten Speicherzellen, so wird auf ein Eprom ab der Adresse 32768 (hex. \$8000) erkannt und zu der in den Speicherzellen 32768/32769 (hex. \$8000/\$8001) abgelegenen Adresse verzweigt. Dies kann man sich für seine eigene Programme zunutze machen, auch wenn sie nicht auf Eprom gebrannt sind, sondern normal eingeladen werden. Legt man nämlich ab Adresse 32768 (hex. \$8000)

bestimmte Daten ab, so folgt nach einem Reset nicht, wie es normalerweise der Fall ist, ein Kaltstart, sondern es wird zu einer eigenen Routine verzweigt, deren Adresse sich in den Speicherzellen 32768/32769 (hex. \$8000/\$8001) in Low-/High-Byte Darstellung befindet. Falls dann auch noch der NMI-Vektor (siehe Abschnitt NMI) entsprechend programmiert wird, ist ein Verlassen des Programms fast unmöglich - in Verbindung mit einem Autostart ein ganz gängiger Kopierschutz. Beim Einladen bzw. nach dem Starten des Programms muß also dafür gesorgt werden, daß sich folgende Informationen in folgenden Speicherzellen befinden:

32768 - \$8000 :	Lo-Byte der Startadresse des eigenen Programms
32769 - \$8001 :	Hi-Byte der Startadresse des eigenen Programms
32770 - \$8002 :	Lo-Byte der Adresse der NMI-Routine
32771 - \$8003 :	Hi-Byte der Adresse der NMI-Routine
32772 - \$8004	dez.
bis	196/194/205/5
32776 - \$8008:	6/48 - hex. \$C3/C2/CD/3
	8/30 - entspricht der Zeichenfolge »CBM80« in ASCII-Schreibweise

Es empfiehlt sich, zu Beginn des eigenen Programms alle Aufgaben zu erledigen, die die Kaltstart-Routine nach einem Reset normalerweise übernimmt, d. h. das Initialisieren der RAMs, der Vektoren u.s.w.. Man sollte also dem eigenen Programm folgende kleine Startroutine in Maschinensprache voranstellen:

2. Break

Aus unserer Aufstellung der verschiedenen Arten des Interrupts ragt der Break als eine Art Sonderling heraus. Anders als der Reset und auch anders als NMI und IRQ, wie Sie noch sehen werden, kann der Break nicht hardwaremäßig erzeugt werden, sondern wird ausschließlich durch den Maschinenbefehl BRK ausgelöst. Trifft der Rechner bei der Abarbeitung eines Maschinenprogramms auf diesen Befehl, so werden Programmzähler und Prozessorstatus auf den Stack gerettet, wobei im Prozessorstatus sowohl die Interrupt- als auch die Break-Flagge gesetzt werden. Danach

holt sich der Rechner aus den Speicherzellen 65534/65535 (hex.\$FFFE/\$FFFF die Startadresse der IRQ-Routine und verzweigt dorthin. In der Routine wird anhand der Break-Flagge im Prozessorstatus entschieden, ob es sich bei dem Interrupt um einen Break handelt. Wie beim Auftreten eines IRQ verfahren wird, soll noch im folgenden bei der Besprechung dieser Interruptart ausführlich erklärt werden. Wird jedoch auf Break erkannt, so verzweigt der Rechner über den Break-Vektor in den RAM-Speicherzellen 790/791 (hex. 0316/\$0317) in die Break-Routine des Betriebssystems. Dies ist zum

Teil mit der NMI-Routine identisch und sorgt dafür, daß I/O- und Interruptvektoren initialisiert werden und ein Basic-Warmstart erfolgt. Hier ist der Punkt, an dem es möglich ist, sich den Break zunutze zu machen. Da der Break-Vektor, der normalerweise auf die Break-Routine weist, im RAM angesiedelt ist, ist es ohne weiteres möglich, ihn zu verändern und mit der Startadresse einer eigenen Routine zu belegen. Nach einem Break erfolgt dann kein Sprung in die Break-Routine des Betriebssystems, sondern vielmehr an den Anfang unseres eigenen Programms. Diese Technik wird von fast allen gängi-

gen Maschinensprachemonitoren genutzt. Diese legen im Break-Vektor die Startadresse einer Monitorroutine ab, die den Inhalt der X- und Y-Register, des Akkumulators, Programmzählers und des Prozessorstatus auf dem Bildschirm anzeigt. Somit ist der BRK-Befehl vorzüglich zum Aus-testen von Maschinenprogrammen geeignet, da er dazu dient, die Ausgabe aller wichtigen Register an einer beliebigen Stelle des Programms zu erzwingen. Diese Methode dürfte allen Maschinenspracheprogrammierern unter den Lesern hinlänglich bekannt sein.

Fortsetzung folgt!

Für die Schneider-User:

ALIEN ATTACK



Wir schreiben das Jahr 2911. Die Menschheit, die seit vielen Jahrhunderten in Frieden lebt, wird plötzlich angegriffen. Da sich die terristrische Flotte, die aus tausend Raumschiffen besteht, sich in einer anderen Galaxis befindet, ist die Menschheit in großer Gefahr. Die wenigen Zerstörer, die noch zur Verfügung stehen, werden in den Kampf geschickt, in der Hoffnung, die Angreifer bis zum Eintreffen der eigenen Flotte aufhalten zu können.

Auf einem dieser Zerstörer sind Sie Kommandant. Steuern Sie Ihr Raumschiff mit dem Joystick. Das Spiel hat 25 Runden. Die Geschwindigkeit erhöht sich mit je-

der Runde. Auch die Bombenabwürfe der Feinde verstärken sich. Insgesamt 20 Angreifer müssen Sie in jeder Runde vernichten. Jeder zählt 75 Punkte. Vernichten Sie die Mutterbasen der Feinde (12mal treffen), erhalten Sie 1000 Punkte.

Hinweise zum Spiel:

Ersten Teil eingeben und abspeichern. Zweiten Teil unter "A2" und den dritten Teil unter "A3" abspeichern.

Durch Starten des ersten Teils werden die anderen beiden Teile nachgeladen und das Spiel beginnt.

100 REM *****		145 REM *****		175 BORDER 26:PAPER 0:PEN 1	(NN)
*****	(MJ)	*****		180 DATA 1,156,336,2,-20,0,2,-3	(LO)
105 REM *		150 REM		0,-96	(L0)
*	(FM)	155 REM		185 DATA 2,20,0,2,10,32,2,20,0	(GE)
110 REM * A L I E N A T T A		161 REM *****		190 DATA 2,10,-32,2,20,0,2,-30,	(KL)
C K *	(CH)	*****		96	(KL)
115 REM *		162 REM *		195 DATA 3,-6,-44,2,-10,0,2,7,2	(LE)
*	(AL)	*		0	(LE)
120 REM * TEIL 1		163 REM * Titelbild		200 DATA 2,5,-20	(CF)
*	(DK)	*		205 DATA 3,-16,44,2,14,14,2,20,	(MJ)
125 REM *		164 REM *		0	(MJ)
*	(FK)	*		210 DATA 2,-14,-14,3,14,14,2,30	(HH)
130 REM * Winklbauer Markus		165 REM *****		-96	(HH)
*	(FB)	*****		215 DATA 2,-14,-14,3,-60,0,2,14	(CI)
135 REM * Desterreich		166 REM		.14	(CI)
*	(JE)	167 REM		220 DATA 2,6,18,3,-4,22,2,6,6	(BO)
140 REM *		168 MODE 2		230 DATA 1,210,336,2,20,0,2,0,-	(AK)
*	(FJ)	170 INK 0,26:INK 1,26		76	(AK)

programme

```

235 DATA 2,40,0,2,0,-20,2,-60,0 (NM) AR5 RIIN "A7" (FD) B (AE)
240 DATA 2,0,96,2,14,14,2,20,0 (JN) 100 REM ***** (DL) 216 DATA 78,B1,20,F2,C9 (HD)
245 DATA 2,-14,-14,3,14,14,2,0, (LG) 102 REM * (HL) 218 REM (CK)
-76 (LJ) 104 REM * ALIEN ATTACK * (BK) 220 REM ***** (LB)
250 DATA 2,-14,-14,3,14,14,2,40 (LJ) 106 REM * (IP) 222 REM * Joystick abfrage * (MC)
,0 (LJ) 108 REM * Teil II * (IB) 224 REM * TEST abfrage * (FG)
255 DATA 2,-14,-14,3,0,-20,2,14 (LJ) 110 REM * (IE) 226 REM ***** (OE)
,14 (KC) 112 REM ***** (NA) 228 REM (HI)
260 DATA 2,0,20 (EA) 114 REM (GE) 230 REM (KE)
270 DATA 1,295,336,2,20,0,2,0,- (JD) 116 REM (LJ) 232 RESTORE 236 (JA)
96,2,-20,0,2,0,96 (JH) 118 REM (KP) 234 FOR adr=&A000 TO &A0D9:READ (NG)
275 DATA 2,14,14,2,20,0,2,-14,- (JH) 120 REM (KE) a#:POKE adr,VAL("&"a#):NEXT (JG)
14,3,14,14,2,0,-96,2,-14,-14 (JH) 122 SYMBOL AFTER 199 (CJ) 236 DATA 21,16,0A,22,D6,A0,CD,0 (JD)
285 DATA 1,344,336,2,70,0,2,0,- (GG) 124 SYMBOL 200,1,1,2,6,7,3,1,3 (FF) A (JG)
20,2,-50,0 (GG) 126 SYMBOL 201,0,0,128,192,192. (FI) 238 DATA A0,C9,00,CD,24,BB,FE,0 (JN)
290 DATA 2,0,-18,2,40,0,2,0,-20 (IF) 128,0,128 (IF) 0 (JN)
,2,-40,0 (IF) 128 SYMBOL 202,6,14,20,20,20,14 (JG) 240 DATA CB,2A,D6,A0,FE,08,2B,4 (JC)
295 DATA 2,0,-18,2,50,0,2,0,-20 (EJ) 130 SYMBOL 203,192,224,80,80,80 (LK) 5 (JC)
,2,-70,0 (EJ) ,224,128 (AI) 242 DATA FE,04,28,01,C9,7C,FE,0 (OD)
300 DATA 2,0,96,2,14,14,2,70,0, (NI) 132 SYMBOL 204,3,68,99,93,91,10 (AI) 244 DATA CB,25,2E,16,22,D6,A0,C (FF)
2,-14,-14 (NI) 1,3,1 (BN) D 246 DATA 9F,A0,CD,75,BB,3E,05,C (NE)
305 DATA 3,14,14,2,0,-20,2,-14, (EE) 134 SYMBOL 205,128,68,140,124,1 (BN) D (NE)
-14 (EE) 80,76,128,0 (BN) D 248 DATA 90,BB,3E,C8,CD,5D,BB,3 (BN)
310 DATA 3,-36,0,2,0,-4,2,-14,- (IM) 136 SYMBOL 206,0,0,0,0,0,0,48,4 (PI) 5 (BN)
14,3,14,14,2,40,0 (IM) B (PI) 250 DATA C9,CD,5D,BB,3E,20,CD,5 (FK)
315 DATA 2,-14,-14,3,14,14,2,0, (BO) 138 SYMBOL 210,7,15,20,20,63,63 (CB) 5 (IA)
-20,2,-14,-14 (BO) ,0,0 (CB) 252 DATA BB,2A,D6,A0,2C,CD,75,B (IA)
320 DATA 3,-26,0,2,0,-4,2,-14,- (DL) 140 SYMBOL 211,224,240,40,40,25 (NB) D 254 DATA 3E,06,CD,90,BB,3E,CA,C (IL)
14,3,14,14,2,50,0 (DL) 2,252,0,0 (NB) D 256 DATA 5D,BB,3E,CB,CD,5D,BB,3 (JG)
325 DATA 2,-14,-14,3,14,14,2,0, (NH) 142 SYMBOL 212,51,52,28,31,54,5 (IE) 5 (GA)
-20,2,-14,-14 (NH) 1,0,0 (IE) 258 DATA 20,CD,5D,BB,C9,7C,FE,1 (GA)
335 DATA 1,442,336,2,20,0,2,30, (JF) 144 SYMBOL 213,204,44,56,248,10 (II) 2 (GA)
-60,2,0,60,2,20,0,2,0,-96 (JF) 8,204,0,0 (II) D 260 DATA CB,24,2E,16,22,D6,A0,C (NE)
340 DATA 2,-20,0,2,-30,60,2,0,- (AI) 146 SYMBOL 214,7,24,63,63,24,7, (EP) 5 (NE)
60,2,-20,0,2,0,96 (AI) 0,0 (EP) D 262 DATA B9,A0,25,CD,75,BB,3E,0 (EO)
345 DATA 2,14,14,2,20,0,2,-14,- (FN) 148 SYMBOL 215,224,24,252,252,2 (DM) 5 (EO)
14,3,14,14,2,16,-32,3,0,18 (FN) 4,224,0,0 (DM) 5 (EO)
350 DATA 2,14,14,2,20,0,2,-14,- (JH) 150 SYMBOL 216,51,14,3,3,14,51 (AN) 264 DATA CD,90,BB,3E,20,CD,5D,B (GJ)
14,3,14,14,2,0,-96,2,-14,-14 (PD) 152 SYMBOL 217,204,112,192,192, (JH) 266 DATA 3E,C8,CD,5D,BB,3E,C9,C (BF)
355 DATA 3,-50,0,2,14,14,2,0,14 (PJ) 112,204 (EG) D 268 DATA 5D,BB,2A,D6,A0,2C,25,C (PJ)
360 DATA 4,4,4 (ED) 154 SYMBOL 218,1,2,31,51,31,4 (LH) 270 DATA 75,BB,3E,06,CD,90,BB,3 (IO)
365 RESTORE 180 (ED) 48,32 (LH) E 272 DATA 20,CD,5D,BB,3E,CA,CD,5 (HE)
370 READ a,b,c:ON a GOSUB 375,3 (ED) 156 SYMBOL 219,128,64,248,204,2 (JM) D (HE)
80,385,390:GOTO 370 (LD) 158 SYMBOL 220,3,68,99,93,91,10 (LI) D 273 DATA BB,3E,CB,CD,5D,BB,C9,C (AP)
375 PLOT b,c,15:RETURN (FJ) 1,3,1 (HA) 0 (AP)
380 DRAWR b,c,15:RETURN (FN) 160 SYMBOL 221,128,68,140,124,1 (AN) 274 DATA 75,BB,CD,60,BB,32,DB,A (PD)
385 PLOTR b,c,15:RETURN (HN) 80,76,128 (AN) 0 (PD)
390 LOCATE 16,14:PRINT "G E S C (NI) 162 SYMBOL 230,63,31,7 (LE) D 276 DATA 2A,D6,A0,2C,CD,75,BB,C (IN)
H R I E B E N" (NB) 164 SYMBOL 231,252,248,224 (CK) 0 (IN)
410 LOCATE 23,16:PRINT "V O N" (PG) 166 SYMBOL 232,255,255,255 (CK) 280 DATA C9,24,CD,75,BB,CD,60,B (JN)
400 LOCATE 10,18:PRINT "W I N K (PG) 168 SYMBOL 233,15,28,56,112,224 (CK) 282 DATA 32,DB,A0,2A,D6,A0,24,2 (FC)
L B A U E R M A R K U S" (CHA) D,255,197,69 (CK) 9 (FF)
405 LOCATE 12,21:PRINT"L o a (BK) 170 SYMBOL 234,240,56,28,14,7,2 (AN) 284 DATA CD,75,BB,CD,60,BB,32,D (FF)
d i n g ! ! ! A L (BK) 55,75,74 (AN) 0 (FF)
I E N A T T A C K" (BK) 172 SYMBOL 235,255,12,24,24,48, (GA) 286 DATA A0,2A,D6,A0,C9,00,00,0 (GO)
410 PLOT 46,96,1:DRAWR 544,0:DR (FK) 255,69,69 (GA) 0 (GO)
AWR 0,110:DRAWR -544,0:DRAWR 0, (FK) 174 SYMBOL 236,255,12,12,12,12, (CLP) 288 DATA 00,00 (OD)
-110 (FK) 255,69,69 (CLP) 290 REM (GN)
415 PLOT 46,56:DRAWR 544,0:DR (NI) 176 SYMBOL 237,255,48,24,24,12, (LP) 292 REM ***** (NF)
R 0,32:DRAWR -544,0:DRAWR 0,-32 (NP) 255,69,69 (NF) 294 REM * Schuss testen * (CN)
420 R=30 (PP) 178 SYMBOL 239,0,0,0,0,255,255, (BM) 296 REM * Treffer Ja-nein * (IN)
425 ORIGIN 410,168 (PP) 195,195 (HA) 298 REM ***** (DL)
430 FOR J=1 TO R STEP 2:X=J:Y=S (GD) 200 REM (HA) 300 REM (LA)
DR(R^2-X^2)*1.1:MOVE -Y,X:DRAW (GD) 202 MEMORY &9FFF (GK) 302 REM (GJ)
50,X,1:MOVE -Y,-X:DRAW 50,-X:NE (FL) 204 RESTORE 208 (PH) 304 RESTORE 308 (JE)
XT (FL) 206 FOR adr=&A100 TO &A124:READ (LN) 306 FOR adr=&A130 TO &A141:READ (HE)
435 R=B (FL) a#:POKE adr,VAL("&"a#):NEXT (IJ) 308 DATA 2A,3E,A1,CD,75,BB,CD,6 (AN)
440 FOR J=1 TO R STEP 2:X=J:Y=S (CI) 210 DATA 01,CD,16,A1,21,Dd,F6,0 (HL)
DR(R^2-X^2)*1.1:MOVE -Y,X:DRAW (BL) 1 (MI)
50,X,0:MOVE -Y,-X:DRAW 50,-X:NE (HG) 212 DATA 68,01,CD,16,A1,C9,7E,E (LJ)
XT (GG) 5 (GD) 316 REM ***** (DO)
445 ORIGIN 520,132 (GG) 214 DATA 19,77,E1,2B,2B,2B,0 (JH)
450 R=30
455 FOR J=1 TO R STEP 2:X=J:Y=S
DR(R^2-X^2)*1.1:MOVE Y,X:DRAW -
50,X,1:MOVE Y,-X:DRAW -50,-X:NE
XT
460 R=B
465 FOR J=1 TO R STEP 2:X=J:Y=S
DR(R^2-X^2)*1.1:MOVE Y,X:DRAW -
50,X,0:MOVE Y,-X:DRAW -50,-X:NE
XT
470 ORIGIN 1,1
475 FOR X=470 TO 560 STEP 2:PLD
T X,102,0:DRAW X,162:NEXT
480 INK 1,3

```

```

320 REM *****
322 REM
324 REM
326 RESTORE 330
328 FOR adr=&A200 TO &A296:READ
a$:POKE adr,VAL("&"a$):NEXT
330 DATA 21,F0,FB,11,EF,FB,CD,4
E
332 DATA A2,21,F0,F0,11,EF,F0,C
D
334 DATA 4E,A2,21,F0,EB,11,EF,E
B
336 DATA CD,4E,A2,21,F0,E0,11,E
F
338 DATA E0,CD,4E,A2,21,F0,DB,1
1
340 DATA EF,DB,CD,4E,A2,21,F0,D
0
342 DATA 11,EF,D0,CD,4E,A2,21,F
0
344 DATA CB,11,EF,CB,CD,4E,A2,2
1
346 DATA F0,C0,11,EF,C0,CD,4E,A
2
348 DATA CD,54,A2,C9,C9,C9,01,3
1
350 DATA 02,ED,B0,C9,21,CF,FA,1
1
352 DATA 50,00,CD,8B,A2,21,CF,F
2
354 DATA CD,8B,A2,21,CF,EA,CD,8
B
356 DATA A2,21,CF,E2,CD,8B,A2,2
1
358 DATA CF,DA,CD,8B,A2,21,CF,D
2
360 DATA CD,8B,A2,21,CF,EA,CD,8
B
362 DATA A2,21,CF,C2,CD,8B,A2,C
9
364 DATA 01,00,00,06,07,7E,19,7
7
366 DATA ED,52,ED,52,10,F7,C9
368 REM
370 REM *****
372 REM * Grosses Raumschiff *
374 REM *****
376 REM
378 RESTORE 382
380 FOR adr=&A150 TO &A1DA:READ
a$:POKE adr,VAL("&"a$):NEXT
382 DATA 3E,0B,CD,90,BB,21,0A,0
A
384 DATA ES,CD,75,BB,3E,20,CD,5
A
386 DATA BB,3E,F1,CD,5A,BB,3E,E
f
388 DATA CD,5A,BB,3E,EF,CD,5A,B
B
390 DATA 3E,ef,CD,5A,BB,3E,F2,C
D
392 DATA 5A,BB,3E,20,CD,5A,BB,3
E
394 DATA 0C,CD,90,BB,E1,2C,E5,C
D
396 DATA 75,BB,3E,20,CD,5A,BB,3
E
398 DATA E9,CD,5A,BB,3E,EB,CD,5
A
400 DATA BB,3E,EC,CD,5A,BB,3E,E
D
402 DATA CD,5A,BB,3E,EA,CD,5A,B
B
404 DATA 3E,20,CD,5A,BB,3E,0D,C
D
406 DATA 90,BB,E1,2C,CD,75,BB
408 DATA 3E,20,CD,5A,BB,3E,E6,C
D
410 DATA 5A,BB,3E,EB,CD,5A,BB,3
E
412 DATA EB,CD,5A,BB,3E,EB,CD,5
A
414 DATA BB,3E,E7,CD,5A,BB,3E,2
0
416 DATA CD,5A,BB,C9
418 REM
(COL) 420 REM *****
(GH) 422 REM * Titelbild speichern *
(GJ) 424 REM * +
(HB) 426 REM * Titelbild hollen *
(DL) 428 REM *****
(FD) 430 REM
432 REM
(OP) 434 RESTORE 438
436 FOR adr=&A300 TO &A397:READ
a$:POKE adr,VAL("&"a$):NEXT
438 DATA 21,F0,FB,11,AB,61,CD,4
9
440 DATA A3,21,F0,F0,11,00,64,C
D
442 DATA 49,A3,21,F0,EB,11,58,6
6
444 DATA CD,49,A3,21,F0,E0,11,B
0
446 DATA 6B,CD,49,A3,21,F0,DB,1
1
448 DATA 0B,6B,CD,49,A3,21,F0,D
0
450 DATA 11,60,6D,CD,49,A3,21,F
0
452 DATA CB,11,BB,6F,CD,49,A3,2
1
454 DATA F0,C0,11,10,72,CD,49,A
3
456 DATA C9,01,31,02,ED,B0,C9,1
1
458 DATA F0,FB,21,AB,61,CD,49,A
3
460 DATA 11,F0,F0,21,00,64,CD,4
9
462 DATA A3,11,F0,EB,21,58,66,C
D
464 DATA 49,A3,11,F0,E0,21,80,6
B
466 DATA CD,49,A3,11,F0,DB,21,0
B
468 DATA 6B,CD,49,A3,11,F0,D0,2
1
470 DATA 60,6D,CD,49,A3,11,F0,C
B
472 DATA 21,BB,6F,CD,49,A3,11,F
0
474 DATA C0,21,10,72,CD,49,A3,C
9
476 CALL &A300
478 RUN "IA3"
(CA) 100 REM *****
(HP) *
(EB) *
(CN) *
(CAL) *
(CHE) *
(CRM) *
(CMH) *
(CPG) *
(COM) *
(CCL) *
(NHD) *
(PH) *
(PK) *
(KI) *
(MI) *
(CG) *
(MB) *
(PP) *
(GP) *
(NA) 126 MODE 2:POKE &A0DB,32:POKE &
A0D9,32
(BE) 127 INK 1,3:INK 0,26
(HN) 128 BORDER 26
(DL) 129 gh=INT(hisc/256):gl=hisc-(2
56*gh)
(FI) 130 POKE &A500,gh:POKE &A501,gl
(GH) 131 PEN 1
(HA) 132 PAPER 0:CLS:CALL &A34F
(CD) 133 WINDOW #4,7,74,13,19
(CK) 134 PAPER #4,1:PEN #4,0:CLS #4
135 s$=" A L I E N A
T T A C K"
(CN) 136 LOCATE#4,10,2:PRINT#4,"G E
S C H R I E B E N"
(CP) 137 LOCATE#4,17,4:PRINT#4, "V O
N"
(BJ) 138 LOCATE#4,5,6:PRINT#4,"W I N
K L B A U E R M A R K U S"
(BO) 139 LOCATE 50,14:PRINT"Hi. Scor
e : ";USING "#####":hisc
(CA) 140 LOCATE 50,15:PRINT"La. Scor
e : ";USING "#####":score
(CJ) 141 LOCATE 50,17:PRINT"S = St
art Spiel
(JM) 142 LOCATE 50,18:PRINT"E = Er
klaerung"
(EK) 143 PLOT 46,96,1:DRAWR 544,0
144 DRAWR 0,110:DRAWR -544,0:DR
AWR 0,-110
(FE) 145 PLOT 46,56:DRAWR 544,0
(AD) 146 DRAWR 0,32:DRAWR -544,0:DR
AWR 0,-32
(FK) 147 PLOT 47,96:DRAW 47,206
148 g=1
(CM) 149 FOR x=7 TO 72:CALL &A200:a$
=INKEY$
(OA) 150 IF a$="e" OR a$="E" OR a$="
s" OR a$="S" THEN 155 ELSE 151
(GC) 151 LOCATE x,21:PRINT LEFT$(s$,
73-x)
(IE) 152 IF x>37 THEN g=g+1:LOCATE 7
,21:PRINT RIGHT$(s$,g)
(AD) 153 NEXT
154 g=1:GOTO 149
(CP) 155 IF a$="s"OR a$="S" THEN RUN
184
(CD) 156 IF a$="e"OR a$="E" THEN 163
(OB) 157 GOTO 148
(CB) 158 REM
(CM) 159 REM *****
(AK) *
(PL) 160 REM * Erklaerung
(ID) *
(CN) 161 REM *****
(IE) 162 REM
(CB) 163 MODE 2
(CN) 164 PLOT 1,10,1:DRAW 639,10
165 DRAW 639,398:DRAW 1,398:DR
AW 1,10
(CA) 166 PLOT 1,356:DRAW 640,356
167 PLOT 1,130:DRAW 640,130
(CB) 168 PLOT 1,40:DRAW 640,40
169 LOCATE 30,2:PRINT"E r k l a
e r u n g"
(CM) 170 LOCATE 2,5:PRINT"Wir schrei
ben das Jahr 2911 . Die Menschh
eit , die seit vielen Jahrhunde
rten
(GD) 171 LOCATE 2,6:PRINT"mit den Le
bewesen der Galaxies in Frieden
lebt , wird ploetzlich angegri
ffen .
(FD) 172 LOCATE 2,8:PRINT"Da sich di
e menschliche Flotte , die aus
tausenden von Raumschiffen best
eht ,
(CB) 173 LOCATE 2,9:PRINT"in einer a
nderen Galaxies befindet , ist
die Menschheit in grosser Gefahr
(EO) 174 LOCATE 2,11:PRINT"Die wenig
en Zerstoeerer , die der Menschh
eit n o c h zur Verfuegung st
ehen ,
(CM)

```

programme

```

175 LOCATE 2,12:PRINT"werden in
den K a m p f geschickt , in
der Hoffnung die Angreifer bis
zum
176 LOCATE 2,13:PRINT"Eintreffe
n der menschliche Flotte a u f
h a l t e n z u k o e n n
e n .
177 LOCATE 2,15:PRINT"S i e
s i n d a u f e i n e m d i e s e r
Zerstoerer K o m m a n d a
n t .
178 LOCATE 2,16:PRINT"i n i h r e n
Haenden liegt die Z u k u n f
t der menschlichen Zivilisat
ion .
179 LOCATE 2,19:PRINT"Sie haben
drei Raumschiffe zur Verfuegan
g u m d i e A n g r e i f e r z u z e r s t o e r e
n .
180 LOCATE 2,21:PRINT"S T E U E
R N k o e n n e n S I E i h r R a u m
schiff mit dem J o y s t i c k
.
181 LOCATE 16,24:PRINT"S P A
C E D R U E C K
E N "
182 IF INKEY(47)=0 THEN 126
183 GOTO 182
184 fa=2
185 INK 0,0:INK 1,26
186 hisc=(FEEK(&A500))*256+(PEE
K(&A501))
187 fe=0
188 an=216
189 REM
190 REM *****
*
191 REM * Spiel START
*
192 REM *****
*
193 REM
194 REM
195 sch=0.85
196 treffer=0
197 raum=0
198 fy=10
199 leb=3
200 level=0
201 GOTO 349
202 scy=3
203 GOSUB 333
204 GOTO 431
205 CALL &A000
206 gew0=8:gew1=12:gew2=7
207 EVERY 12,1 GOSUB 216
208 EVERY 7,2 GOSUB 292:REM Sc
huss Scrollen
209 EVERY 8,0 GOSUB 278:REM sc
huss abfrage
210 GOTO 210
211 REM
212 REM *****
*
213 REM * Angreifer
*
214 REM *****
*
215 REM
216 DI:ON 1e GOTO 220,225,230,2
35,240,245,250,257,264
217 REM
218 REM *** von links nach rech
ts ***
219 REM
220 si=RND*1:PEN fa:sx=sx+1:sy=
sy+1:LOCATE sx-1,sy-s1:PRINT"
":IF sx< 20 THEN LOCATE sx,sy
:PRINT CHR$(an):CHR$(an+1):ELSE
266
221 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx+1,sy:PRINT C
HR$(22):CHR$(1):CHR$(206):CHR$(
22):CHR$(0):GOTO 272
222 REM
223 REM *** von rechts nach lin
ks ***
224 REM
225 si=RND*1:sx=sx-1:PEN fa:sy=
sy+1:LOCATE sx+1,sy-s1:PRINT"
":IF sx> 1 THEN LOCATE sx,sy
:PRINT CHR$(an):CHR$(an+1):ELSE
266
226 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx,sy:PRINT CHR
$(22):CHR$(1):CHR$(206):CHR$(22
):CHR$(0):GOTO 272
227 REM
228 REM ** von links nach recht
s 1 **
229 REM
230 PEN fa:sx=sx+1:LOCATE sx-1,
sy:PRINT " ":CHR$(an):CHR$(an+1
):IF sx=19 THEN LOCATE sx,sy:PR
INT " ":GOTO 266
231 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx+1,sy:PRINT C
HR$(22):CHR$(1):CHR$(206):CHR$(
22):CHR$(0):GOTO 272
232 REM
233 REM ** von rechts nach link
s 1 **
234 REM
235 PEN fa:sx=sx-1:LOCATE sx,sy
:PRINT CHR$(an):CHR$(an+1):" ":
:IF sx=1 THEN LOCATE sx,sy:PRIN
T " ":GOTO 266
236 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx,sy:PRINT CHR
$(22):CHR$(1):CHR$(206):CHR$(22
):CHR$(0):GOTO 272
237 REM
238 REM ** von unten nach oben
1-r **
239 REM
240 si=RND*1:PEN fa:sx=sx+1:sy=
sy-s1:LOCATE sx-1,sy+1:PRINT"
":IF sx< 20 THEN LOCATE sx,sy
:PRINT CHR$(an):CHR$(an+1):ELSE
266
241 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx+1,sy:PRINT C
HR$(22):CHR$(1):CHR$(206):CHR$(
22):CHR$(0):GOTO 272
242 REM
243 REM ** von oben nach unten
r-1 **
244 REM
245 si=RND*1:sx=sx-1:PEN fa:sy=
sy-s1:LOCATE sx+1,sy+1:PRINT"
":IF sx> 1 THEN LOCATE sx,sy
:PRINT CHR$(an):CHR$(an+1):ELSE
266
246 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx,sy:PRINT CHR
$(22):CHR$(1):CHR$(206):CHR$(22
):CHR$(0):GOTO 272
247 REM
248 REM *** Spezial 1 ***
249 REM
250 IF sy<10 THEN dx=-1:dy=1:EL
SE dx=1
251 IF sy=17 THEN dy=-1:dx=1
252 PEN fa:LOCATE sx,sy:PRINT"
":sy=sy+dy:sx=sx+dx:IF sx=20
THEN 266 ELSE:LOCATE sx,sy:PRIN
T CHR$(an):CHR$(an+1)
253 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx,sy:PRINT CHR
$(22):CHR$(1):CHR$(206):CHR$(22
):CHR$(0):GOTO 272
254 REM
255 REM *** Spezial 2 ***
256 REM
257 IF sy<10 THEN dx=1:dy=1:ELS
E dx=-1
258 IF sy=17 THEN dy=-1:dx=-1
259 PEN fa:LOCATE sx,sy:PRINT"
":sy=sy+dy:sx=sx+dx:IF sx=1 T
HEN 266 ELSE:LOCATE sx,sy:PRIN
T CHR$(an):CHR$(an+1)
260 IF RND<sch THEN GOTO 272 EL
SE:PEN 1:LOCATE sx,sy:PRINT CHR
$(22):CHR$(1):CHR$(206):CHR$(22
):CHR$(0):GOTO 272
261 REM
262 REM *** B O N U S
***
263 REM
264 SOUND 2,100,6,13:PEN 3:sx=s
x+1:LOCATE sx-1,sy:PRINT " ":CH
R$(220):CHR$(221):IF sx=19 THEN
LOCATE sx,sy:PRINT " ":GOTO 2
66
265 GOTO 272
266 GOSUB 333
267 REM
268 REM *****
*
269 REM * Joy - Abfrage
*
270 REM *****
*
271 REM
272 CALL &A00B:IF PEEK(&A00B)<>
32 OR PEEK(&A009) <> 32 THEN 29
0:RETURN
273 REM
274 REM *****
*
275 REM * Schuss ja-nein
*
276 REM *****
*
277 REM
278 DI:IF SC=1 THEN 281
279 IF JOY(0)=16 THEN SC=1:SCX=
PEEK(&A0D7):SOUND 1,140,20,14,0
,1,15
280 IF sc=0 THEN RETURN
281 SCY=SCY+1:PLOT SCX*32-4,SCY
*16+6,3:DRAW 0,0:PLOT 0,-32,0
:DRAW 0,-8:
282 IF INT(sx)=scx AND INT(sy)=
25-scy OR INT(sx)=scx AND INT(s
y)=24-scy THEN 323
283 scy=scy+1:IF INT(sx)=scx AN
D INT(sy)=25-scy OR INT(sx)=scx
AND INT(sy)=24-scy THEN 323
284 IF SCY=21 THEN PLOT 0,32,0
:DRAW 0,0:SCY=3:SC=0
285 EI:RETURN
286 INK 13,20
287 REM
288 REM *****
*
289 REM * Scrollen Schuss
*
290 REM *****
*
291 REM
292 DI:CALL &A100:IF TEST (PEEK
(&A0D7)*32-20,50)>0 OR TEST ((P
EEK(&A0D7)+1)*32-20,50)>0 THEN
298 ELSE RETURN
293 REM
294 REM *****
*
295 REM * Getroffen
*
296 REM *****
*
297 REM
298 fe=0:pe=FEEK(&A0D7)
299 POKE &A00B,32:POKE &A0D9,32
300 TAG:MOVE 700,1:PRINT REMAIN
(2):REMAIN(1):REMAIN(0):TAGOFF
301 SOUND 1,10,200,15,2,2,5
302 FOR x=1 TO 40:PEN RND*14+1:
LOCATE pe,22:PRINT CHR$(RND*15+
129):CHR$(RND*15+129):LOCATE pe
,23:PRINT CHR$(RND*15+129):CHR$(
RND*15+129):NEXT:PEN 1
303 LOCATE pe,22:PRINT " ":LOC
AIE pe,23:PRINT " "
304 leb=leb-1

```

```

305 FOR x=1 TO 1000:NEXT
306 TAG:MOVE 192,20:PRINT " ";
MOVE 188,14:PRINT " ";
307 PLOT 650,1,11:MOVE 192,20:P
RINT leb;
308 TAGOFF:PRINT CHR$(23);CHR$(
1);:TAG
309 PLOT 650,1,10:MOVE 188,16:P
RINT leb;
310 TAGOFF:PRINT CHR$(23);CHR$(
0);
311 IF leb=0 THEN 485
312 WINDOW #2,1,25,5,23:CLS #2
313 IF raum=1 THEN GOSUB 371 EL
SE GOSUB 464
314 POKE &A0D7,10
315 PEN 5:LOCATE 10,22:PRINT CH
R$(200);CHR$(201)
316 PEN 6:LOCATE 10,23:PRINT CH
R$(202);CHR$(203)
317 RETURN
318 REM
319 REM *****
*
320 REM *      Treffer
*
321 REM *****
*
322 REM
323 PEN 8:LOCATE sx,sy:PRINT CH
R$(22);CHR$(1);CHR$(159);CHR$(1
59);CHR$(22);CHR$(0)
324 SOUND 1,1000,100,7,1,1,15:P
LOTR 0,32,0:DRAWR 0,8
325 TAG:MOVE 700,1:PRINT REMAIN
(2):TAGOFF
326 IF le=9 THEN score=score+50
0 ELSE score=score+75
327 LOCATE sx,sy:PRINT " ";
328 GOSUB 475
329 scy=3:sc=0
330 EVERY gew2,2 GOSUB 292:REM
Schuss Scrollen
331 treffer=treffer+1:IF treffe
r =20 THEN 365
332 IF treffer=7 OR treffer=14
THEN le=9:GOTO 334
333 le=INT(RND*8+1)
334 IF le=1 THEN sy=5:sx=1
335 IF le=2 THEN sy=5:sx=20
336 IF le=3 THEN sy=INT(RND*12+
5):sx=1
337 IF le=4 THEN sy=INT(RND*12+
5):sx=19
338 IF le=5 THEN sy=19:sx=1
339 IF le=6 THEN sy=19:sx=20
340 IF le=7 THEN sy=5:sx=INT(RN
D*14+6)
341 IF le=8 THEN sy=5:sx=INT(RN
D*13+1)
342 IF le=9 THEN sy=5:sx=1
343 RETURN
344 REM
345 REM *****
*
346 REM *      Bildschirmaufbau
*
347 REM *****
*
348 REM
349 MODE 0:BORDER 0
350 PRINT CHR$(23);CHR$(1)
351 TAG:PLOT 640,400,11:MOVE 0,
398:PRINT " ALIEN Score 0"
;:MOVE 0,366:PRINT" ATACK Hi s
c ";:PRINT USING "#####":hisc;

352 MOVE 0,20:PRINT" Lives 3
Level 1";
353 PLOT 640,400,10:MOVE -4,394
:PRINT " ALIEN Score 0";:M
OVE -4,362:PRINT" ATACK Hi sc
";:PRINT USING "#####":hisc;
354 MOVE -4,16:PRINT" Lives 3
Level 1";
355 TAGOFF

```

```

6 THEN an=216:INK 2,20
458 IF level=2 OR level=7 OR level=12 OR level=17 OR level=22
THEN an=214:INK 2,23
459 IF level=3 OR level=8 OR level=13 OR level=18 OR level=23
THEN an=212:INK 2,17
460 IF level=4 OR level=9 OR level=14 OR level=19 OR level=24
THEN an=210:INK 2,14
461 IF level=5 OR level=10 OR level=15 OR level=20 OR level=25
THEN an=218:INK 2,22
462 treffer=0
463 POKE &A0D7,10
464 EVERY gew1,1 GOSUB 216
465 EVERY gew2,2 GOSUB 292
466 EVERY gew0,0 GOSUB 278
467 x=10:scy=3:sc=0
468 GOSUB 333
469 RETURN
    
```

```

(AH) 470 REM
      471 REM *****
      *
(FI) 472 REM *   Score anzeigen
      *
(IP) 473 REM *****
      *
      474 REM
      475 TAG: PLOT 700,1,11: MOVE 448,
      398: PRINT "      ": MOVE 444,394
      : PRINT "      "
      : TAG
(LI) 477 MOVE 448,398: PRINT USING "##
      ###": score: PLOT 700,1,10: MOVE
      444,394: PRINT USING "#####": score
(LN)
      478 TAGOFF: PRINT CHR$(23); CHR$(
      0);
(FM) 479 RETURN
      480 REM
    
```

```

(LB) 481 REM *****
      *
      482 REM *           Game Over
      *
      483 REM *****
      *
      484 REM
      485 WINDOW #4,8,13,11,15: PAPER
      #4,14:CLS #4
      486 PEN #4,0
      487 PRINT#4,"           Game
      Over":
      488 FOR x=1 TO 1500:NEXT
      489 FOR x=1 TO 320 STEP 4: PLOT
      x,1,0: DRAW 0,400: PLOT 640-x,1:
      DRAW 0,400:NEXT
      490 IF score>hisc THEN hisc =score
      491 MODE 2:FEN 1:INK 1,26
      492 GOTO 126
      (NM)
      (JJ)
      (AK)
      (KH)
      (BH)
      (MN)
      (HK)
      (MK)
      (JE)
      (HE)
      (KP)
      (CE)
    
```



Lucky Luck!



In Listing 3 von Lucky Luck muß folgende Zeile eingefügt werden:
5 MEMORY 35000

Sie sind der Revolverheld Lucky Luck und müssen ein Fort von bösen, bösen Banditen befreien. Steuern Sie Ihr Zielkreuz auf die auftauchenden Banditen. Mit Feuer schießen Sie. Wenn ein Bandit vollkommen erschienen ist, wird er auf Sie schießen. Um nicht getroffen zu werden, können Sie mit SPACE in Deckung gehen. An der Bildschirmrandfarbe erkennen Sie wie lange Sie sich in der Deckung befinden.

Eingeben:

1. Listing 1 eingeben und abspeichern.
2. Listing 2 eingeben und mit RUN starten. Dieses Programm erzeugt den Maschinencode und speichert ihn automatisch ab.
3. Computer mit SHIFT+CTRL+ESC zurücksetzen.
4. Listing 3 eingeben und wie mit Listing 2 verfahren.
5. Zum Spielen Listing 1 mit RUN starten, es lädt die anderen beiden Teile nach.

```

10 MEMORY 30000:LOAD"sprite":CALL
40000:LOAD"town"
20 DATA 0,39837,1,39675,2,39513,
3,39351,4,39189,5,39027,6,38863,
7,38863,8,38701,9,38459,10,38
297,11,38215,12,38063,13,37911,
14,37879,15,37827,16,37793,17,3
7759,18,37725,19,37623,20,37521
,21,37489,22,37439,23,37389,24,
37339,25,37289,26,37239,27,3715
3
30 DATA 28,37067,29,36981,30,36
895,31,36809
40 RESTORE 20:FOR n=0 TO 31:REA
D a,b: I PHASE,a,b:NEXT
50 ENV 1,1,15,3,3,-5,10:ENV 2,1
,15,14,5,-3,30:ENV 3,5,3,1,9,0,
0,5,-3,0:ENT 1,50,2,2
60 MODE 0:RESTORE 230:FOR n=1 T
O 15:READ a:INK n,a:NEXT
70 LOCATE 6,6:PRINT"LUCKY LUCK"
:LOCATE 6,7:PRINT"=====":L
OCATE 16,25:PRINT CHR$(164)+"19
86":LOCATE 1,21:PRINT"WRITTEN B
Y":LOCATE 10,23:PRINT"R. TUELLMA
    
```

```

(NN):RESTORE 90
80 FOR n=1 TO 15:READ a,b,c: I SP
RITE,1,a: I SWITCH,1,b,c:NEXT
90 DATA 12,7,30,13,22,31,12,38,
29,13,49,30,12,63,31,13,75,30,1
9,0,34,19,15,35,19,30,34,19,43,
35,19,55,35,19,70,34,14,11,35,1
4,40,37,14,67,36
100 DATA .16,130,.16,146,.16,17
4,.16,110,130,138,369,103,.16,1
30,.16,110,.16,130,.16,138,.16,
174,.16,110,130,138,369,82,.16,
97,.16,82,.16,97,.16,110,.16,13
0,.16,82,97,110,277,110,.16,130
,.16,110,.16,130,.16,138,.16,17
4,.16,110,130,138,739,73,.16,87
,.16,73
(CFA) 110 DATA .16,87,.16,97,.16,116,
      .16,82,97,103,277,103,.16,130,
      .16,110,.16,130,.16,138,.16,174,
      .16,110,130,138
      (HJ) 120 Y=11:RESTORE 100:FOR Q=1 TO
      56:READ X
      130 IF X<>INT(X) THEN READ Z:SOU
      ND 9,7,X*150,5,1:GOTO 160
    
```

```

(EN) 140 SOUND 9,X,Y,5,1
      150 IF JOY(0)=16 THEN 170
      160 NEXT:GOTO 120
      170 CLS:LOCATE 1,10:PRINT"Gelen
      kt wird mit dem":LOCATE 1,12:PR
      INT"Joystick.Mit Space":LOCATE 1
      ,14:PRINT"gehst Du in Deckung."
      (CF)
      (OL)
      (AB)
      (CO)
      (CJ)
      (GN)
      (DA)
      (BJ)
      (LI)
      (OL)
      (DJ)
      210 FOR n=1 TO 1000:NEXT:lev=30
      :lev1=12:sch=50:men=3
      220 MODE 0:FOR n=1 TO 15:INK n,
      1:NEXT:GOSUB 250:RESTORE 230:FO
      R n=1 TO 15:READ a:INK n,a:NEXT
      230 DATA 26,24,16,20,18,18,10,1
      2,10,9,8,7,6,4,5
      240 GOTO 340
      250 REM"-----Bild aufbau
      -----
      (LI)
      260 LOCATE 1,3:PRINT"LEBEN: "me
    
```



```

n:LOCATE 1,1:PRINT"PUNKTE:"pun:
LOCATE 1,2:PRINT"SCHUSS:"sch
270 IF lev=30 THEN RESTORE 1180
280 IF lev=25 THEN RESTORE 1210
290 IF lev=20 THEN RESTORE 1240
300 IF lev=15 THEN RESTORE 1270
310 IF lev=10 THEN RESTORE 1300
320 FOR n=1 TO 65:READ a,b,c:IS
PRITE,1,a:ISWITCH,1,b,c:NEXT:RE
TURN
330 RETURN
340 REM -----Zielkreuz
Seuerung-----
350 x=39:y=20:ISPRITE,2,20:ISWI
TCH,2,x,y
360 i$=BIN$(JOY(0),5)
370 IF sch>0 AND MID$(i$,1,1)="
1"THEN GOSUB 540
380 IF MID$(i$,2,1)="1"AND x<74
THEN x=x+2
390 IF MID$(i$,3,1)="1"AND x>1
THEN x=x-2
400 IF MID$(i$,4,1)="1"AND y<45
THEN y=y+1
410 IF MID$(i$,5,1)="1"AND y>0
THEN y=y-1
420 IF sut=40 THEN sut=0
430 IF sut>15 THEN 470
440 IF INKEY(47)=0 THEN BORDER
10:sut=sut+1
450 IF sut=15 THEN BORDER 1
460 IF sut=0 THEN 480
470 sut=sut+1
480 ISPRITE,2,20:IMOVE,2,x,y:IS
YNC
490 IF kom=1 THEN GOSUB 840
500 IF kom1=1 THEN GOSUB 1020
510 IF kom1=0 AND INT(RND*lev)=
1 THEN kom1=1:pha1=27:GOTO 930
520 IF kom=0 AND INT(RND*lev)=1
THEN kom=1:pha=22:GOTO 730
530 GOTO 360
540 SOUND 129,0,0,0,1,1,15:ISPR
ITE,3,21:ISWITCH,3,x+1,y+1:sch=
sch-1:LOCATE 8,2:PRINT sch
550 IF x+1<x1+2 AND x+1>x1-1 AN
D y+1<y1+3 AND y+1>y1 AND kom=1
THEN SOUND 2,200,15,5,0,0,1:IS
PRITE,S,pha:ISWITCH,5,x1,y1:kom
=0:pun=pha+pun:tre=tre+1:sch=sc
h+1
560 IF x+1<x2+2 AND x+1>x2-1 AN
D y+1<y2+5 AND y+1>y2-1 AND kom
1 THEN SOUND 2,200,15,5,0,0,1
:ISPRITE,6,pha1:ISWITCH,6,x2,y2
:kml=0:pun=pha1+pun:tre=tre+1:
sch=sch+1
570 IF tre=10 THEN 580 ELSE 650
580 lev=lev-5:lev1=lev1-2:tre=0
:sch=sch+5
590 IF lev=5 THEN 600 ELSE 640
600 CLEAR:FOR n=1 TO 100:NEXT
610 FOR n=1 TO 15:m=INT(RND*27)
:INK n,m:SOUND 130,m*20,m,5
620 IF JOY(0)=16 THEN 60
630 NEXT:GOTO 610
640 pun=pun+100:SOUND 2,70,5,7:
SOUND 2,150,10,7:SOUND 2,70,5,7
:kml=0:kml=0:GOTO 220
650 LOCATE 8,1:PRINT pun:LOCATE
8,2:PRINT sch:IF pun>4999 THEN
men=men+1
660 ISPRITE,3,21:ISWITCH,3,x+1,
y+1:RETURN
670 REM -----Cowbo
ys setzen-----
680 DATA 14,23,5,22,38,25,62,22
,66,22
690 DATA 4,23,8,23,24,23,48,27,
37,23
700 DATA 26,23,54,23,58,23,6,28
,75,26
710 DATA 9,23,25,23,38,19,43,19
,55,23
720 DATA 9,23,25,23,37,21,53,21

```

```

,75,21
(IL) 730 IF lev=30 THEN RESTORE 680
(MI) 740 IF lev=25 THEN RESTORE 690
(BH) 750 IF lev=20 THEN RESTORE 700
(FK) 760 IF lev=15 THEN RESTORE 710
(FN) 770 IF lev=10 THEN RESTORE 720
(ID) 780 READ a1,a2,a3,a4,a5,a6,a7,a
8,a9,a10:ata=INT(RND*5):IF ata=
0 THEN x1=a1:y1=a2
(NG) 790 IF ata=1 THEN x1=a3:y1=a4
(NB) 800 IF ata=2 THEN x1=a5:y1=a6
810 IF ata=3 THEN x1=a7:y1=a8
820 IF ata=4 THEN x1=a9:y1=a10
830 ISPRITE,5,pha:ISWITCH,5,x1,
y1:GOTO 360
(FE) 840 dau=dau+1:IF dau<lev1 THEN
RETURN ELSE dau=0
(GK) 850 IF pha=26 THEN 1060
860 ISPRITE,5,pha+1:ISWITCH,5,x
1,y1:ISPRITE,5,pha:ISWITCH,5,x1
,y1:pha=pha+1:RETURN
(NF) 870 REM -----
(PN) 880 DATA 22,23,47,22,54,22
890 DATA 16,23,57,25,64,25
(AN) 900 DATA 2,23,17,23,68,25
(GN) 910 DATA 2,23,17,23,45,21
(BC) 920 DATA 19,23,34,23,46,23
930 IF lev=20 THEN RESTORE 920
(FN) 940 IF lev=30 THEN RESTORE 880
(ND) 950 IF lev=25 THEN RESTORE 890
(PC) 960 IF lev=15 THEN RESTORE 900
(KD) 970 IF lev=10 THEN RESTORE 910
980 READ a1,a2,a3,a4,a5,a6:ata=
INT(RND*3):IF ata=0 THEN x2=a1:
y2=a2
(EK) 990 IF ata=1 THEN x2=a3:y2=a4
(HD) 1000 IF ata=2 THEN x2=a5:y2=a6
1010 ISPRITE,6,pha1:ISWITCH,6,x
2,y2:GOTO 360
(BP) 1020 dau1=dau1+1:IF dau1<lev1 T
HEN RETURN ELSE dau1=0
(PD) 1030 IF pha1=31 THEN 1060
1040 ISPRITE,6,pha1+1:ISWITCH,6
,x2,y2:ISPRITE,6,pha1:ISWITCH,6
,x2,y2:pha1=pha1+1:RETURN
(KM) 1050 REM -----Ende-----
1060 IF sut>0 AND sut<16 THEN 1
090
(BP) 1070 SOUND 130,0,0,0,2,0,31:FOR
m=10 TO 1 STEP-1:FOR n=1 TO 15
:INK n,m:NEXT:FOR j=1 TO 100:NE
XT:NEXT:men=men-1:IF men>0 THEN
1080 ELSE 1120
1080 ISPRITE,5,pha:ISWITCH,5,x1
,y1:kom=0:pha=0:ISPRITE,6,pha1:
ISWITCH,6,x2,y2:kom1=0:pha1=0:G
OTO 220
(NA) 1090 IF pha=26 THEN ISPRITE,5,ph
a:ISWITCH,5,x1,y1:kom=0:pha=0
(JM) 1100 IF pha1=31 THEN ISPRITE,6,p
ha1:ISWITCH,6,x2,y2:kom1=0:pha1
=0
(ON) 1110 GOTO 370
(GC) 1120 MODE 0:INK 1,20:LOCATE 9,1
3:PRINT"END":IF rek=0 THEN rek=
pun:GOTO 1140
(HD) 1130 IF rek<pun THEN rek=pun
1140 LOCATE 5,1:PRINT"REKORD:"r
ek:LOCATE 5,25:PRINT"PUNKTE:"pu
n
(OD) 1150 IF JOY(0)=16 THEN 1170
1160 GOTO 1150
(PO) 1170 pun=0:dau=0:dau1=0:tre=0:s
ut=0:kom=0:kml=0:pha=0:pha1=0:
GOTO 60
(JB) 1180 DATA 0,10,20,4,14,20,3,18,
20,5,22,20,1,26,20,2,30,20,6,10
,15,7,14,15,7,18,15,7,22,15,8,2
6,15,9,34,26,10,71,29,11,5,25,1
2,10,40,13,50,42,14,25,45,19,75
,40,13,28,42,12,60,40,19,33,45,
19,2,43,19,18,45,19,56,44,0,50,
19,5,54,19,3,58,19,4,62,19,4,66
(LK) 1190 DATA 19,1,70,19,2,74,19,6,

```

```

(CHH) 50,14,7,54,14,7,58,14,7,62,14,7
(BB) ,66,14,8,70,14,15,0,35,16,2,36,
(GJ) 17,2,38,15,10,35,18,12,36,16,12
(JH) ,38,15,20,35,17,22,36,16,22,38,
(HH) 15,30,35,16,32,36,18,32,38,15,4
(CB) 0,35,18,42,36,17,42,38,15,50,35
,17,52,36,18,52,38,15,60,35,16
(EI) 1200 DATA 62,36,18,62,38,15,70,
35,17,72,36,17,72,38,14,7,32,14
(CJ) ,43,40,14,55,33,14,68,46
(FN) 1210 DATA 0,0,20,4,4,20,4,8,20,
(GK) 3,12,20,5,16,20,3,20,4,24,20
(HC) ,1,28,20,2,32,20,6,0,15,7,4,15,
7,8,15,7,12,15,7,16,15,7,20,15,
(JM) 7,24,15,8,28,15,0,60,22,5,64,22
,3,68,22,1,72,22,2,76,22,6,60,1
7,7,64,17,7,68,17,8,72,17,9,43,
(BE) 28,11,37,26,15,0,35,15,10,35,15
,20
(NG) 1220 DATA 35,15,30,35,15,48,35,
15,58,35,15,68,35,15,78,35,16,2
(COE) ,36,17,12,36,18,22,36,18,50,36,
16,60,36,17,70,36,18,2,38,16,12
(GF) ,38,17,22,38,16,50,38,17,60,38,
(PN) 18,70,38,12,5,40,13,25,42,12,51
(LJ) ,41,13,72,39,19,0,45,19,11,42,1
(JD) ,9,19,45,19,46,44,19,60,43,19,75
(EP) 1230 DATA 45,14,20,30,14,60,32,
(AF) 14,9,45,14,55,46,14,69,46,14,2,
(FH) 32,14,30,47
(CI) 1240 DATA 0,22,20,4,26,20,3,30,
(IA) 20,5,34,20,3,38,20,3,42,20,5,46
(MK) ,20,3,50,20,4,54,20,4,58,20,1,6
(LJ) 2,20,2,66,20,6,22,15,7,26,15,7,
30,15,7,34,15,7,38,15,7,42,15,7
(NN) ,46,15,7,50,15,7,54,15,7,58,15,
(FL) 8,62,15,9,2,29,14,27,34,11,70,2
(NI) 9,11,75,29,19,10,45,19,50,44,14
,25,45
(KH) 1250 DATA 19,75,43,19,28,43,19,
60,45,13,33,40,12,2,41,13,18,42
(BJ) ,12,56,40,15,0,35,16,2,36,17,2,
(FB) 38,15,10,35,18,12,36,16,12,38,1
(IP) 5,20,35,17,22,36,16,22,38,15,30
,35,16,32,36,18,32,38,15,40,35,
(KM) 18,42,36,17,42,38,15,50,35,17,5
2,36,18,52,38,15,60,35,16,62,36
,18,62
(AG) 1260 DATA 38,15,70,35,17,72,36,
17,72,38,14,20,32,14,43,40,14,5
(FJ) 5,33,14,68,46
1270 DATA 0,5,20,4,9,20,3,13,20
,5,17,20,3,21,20,4,25,20,1,29,2
,0,2,33,20,6,5,15,7,9,15,7,13,15
(FL) ,7,17,15,7,21,15,7,25,15,8,29,1
5,0,64,22,5,68,22,1,72,22,2,76,
22,6,64,17,7,68,17,8,72,17,11,3
8,22,11,43,22,9,50,24,15,0,35,1
5,10,35,15,20,35,15,30,35,15,48
,35,15
(CBB) 1280 DATA 58,35,15,68,35,15,78,
35,16,2,36,17,12,36,18,22,36,18
,50,36,16,60,36,17,70,36,18,2,3
8,16,12,38,17,22,38,16,50,38,17
,60,38,18,70,38,19,5,45,19,25,4
4,19,51,45,19,72,43,12,0,40,13,
12,42,12,19,42,13,46,42,12,60,4
(PP) 1,13,75,42,14,20,30,14,60,32,14
(PL) 1290 DATA 9,45,14,55,46,14,69,4
6,14,2,32,14,30,47,14,77,32,14,
25,40,14,35,31
(CCE) 1300 DATA 0,5,20,4,9,20,3,13,20
,5,17,20,3,21,20,4,25,20,1,29,2
(GF) 0,2,33,20,6,5,15,7,9,15,7,13,15
(FE) ,7,17,15,7,21,15,7,25,15,8,29,1
5,4,37,18,3,41,18,5,45,18,3,49,
18,4,53,18,1,57,18,2,61,18,6,37
(JL) ,13,7,41,13,7,45,13,7,49,13,7,5
3,13,8,57,13,11,75,24,10,66,24,
15,0,35
(GM) 1310 DATA 16,2,36,17,2,38,15,10
,35,18,12,36,16,12,38,15,20,35,
17,22,36,16,22,38,15,30,35,16,3
2,36,18,32,38,15,40,35,18,42,36
,17,42,38,15,50,35,17,52,36,18,
52,38,15,60,35,16,62,36,18,62,3

```



```
00000000000000000000000000000000
000C3C3C3C300000000000000000000000
3C3C2B0000142B0000142B0000142B0
000142B0000142B0000142B0000142B
0000142B0000142B0000142B0000142
B0000142B0000", 3180
320 DATA "142B0000142B0000142B0
000142B0000142B0000142B0000142B
0000142B0000142B0000142B0000142
B0000142B0000142B0000142B000014
2B0000142B00001430303030042B3C
3C3C3000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
330 DATA "3C2B0000142B0000142B0
000142B0000142B0000142B0000142B
0000142B0000142B0000142B0000142
B0000142B0000142B0000142B000014
2B0000142B0000142B0000142B00001
43C3C3C3C30000000000000000000000
00000000000000000000000000000000
000C3C3C3C300", 2940
340 DATA "0000000000000000000003030303
0042B3C3C3C3000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
```

```
00000000C3C3C3C3000000000000000000
00000000000000000000000000000000C3C3C3
C3000000000000000000000000000000000000
0000000000000000000000000000000000000
350 DATA "C3C3C300000000000000000000
00000000000000000000000000000000000000
00000000000000000000000000000000000000
00000000000000000000000000000000000000
00000000000000000000000000000000000000
000030303030042B0000000000000000000000
20041B2B241B241C3B20041C3000041
C3000000000000000000000000000000000000
041B200004100", 5176
360 DATA "00004100000041000041B
20041B2B241B241C3B20041C3000041
C3000000000000000000000000000000000000
041B200004100000041000000410000
41B20041B2B241B241C3B20041C30000
041C3000000000000000041B20041B24141
B20010B200102000102000102000002
00000000000000", 5652
370 DATA "0000000000042B3C341B20
041B20000C3C30000C3C30000C3C3000
4141B241B2C341B20000C3000000C3B
20000C3B20000C3B24100C341B2C341
B20041B20000C3C30000C3C30000C3C
3004141B241B2C341B20000C3000000
```

```
C3B20000C3B20000C3B24100C341B2C
341B20041B200", 9079 (CI)
380 DATA "00C3C30000C3C30000C3C
3004141B241B2C341B20000C3000000
C3B20000C3B20000C3B24100C341B2C
341B20041B20010C3C3102030302000
042B00C3C3C341B20000C3C30000C3C
30000C3C3000041B2000000C3C3C341
0000004100000041000000410000004
100000000C3C3", 8296 (FM)
400 start= 36700:wieoft= 29 (BL)
10000 MODE 2 (BH)
10020 RESTORE:FOR n=1 TO wieoft (MP)
10030 READ a$,b:check=0:PRINT"Z
eile";90+n*10; (EO)
10040 FOR n1=1 TO LEN(a$) STEP
2:POKE start,VAL("&" +MID$(a$,n1
,2)):check=check+PEEK(start):st
art=start+1:NEXT (GM)
10050 IF check<>b THEN PRINT"Fe
hier in Zeile";90+n*10 (HH)
10060 PRINT"OK." (LB)
10070 NEXT (OL)
10080 PRINT"BITTE DISCK/CASS EI
NBLEGEN UND TASTE DRUECKEN":CAL
L &BB18:SAVE"DOWN",b,36000,3979 (ED)
```

MOUSE

nur CPC 464



Es gibt ein altes Kinderspiel, bei dem man durch Verschieben von kleinen Plättchen, die sich in einem Rahmen befinden, ein Bild oder eine Zahlenfolge zusammensetzen muß. Der Reiz dieses Spiels liegt darin, daß nur eine Lücke existiert, in die die benachbarte Plättchen geschoben werden können. Das Prinzip von Mouse ist ein ähnliches. In einer 4 x 4 Matrix müssen die Blöcke so verschoben werden, daß die in dem Labyrinth herumirrende Maus sich frei bewegen kann. Die Maus muß nun die 8 Gegenstände in dem Labyrinth einsammeln. Allerdings existiert da noch ein Zeitlimit.

```
10 REM *****
*****
20 REM *
(NK) 100
110 DEF FNx1(z)=INT((z-11)/5+1)
(NN) 120 DEF FNy1(z)=INT((z-4)/5+1)
(MH) 130 DEF FNx(z)=10+z*10+1
140 DEF FNy(z)=z*5-1
(AH) 150 MODE 1:CALL &BC02
160 f%:0:lev=0:anf=0:liv=4:sc=0
:DIM I(5,5)
(NB) 170 LOCATE 15,12:PRINT "Bitte w
arten"
(OH) 180 GOSUB 4020:GOSUB 4320
190 GOSUB 800
(OC) 200 GOTO 4110
(DC) 210 INK 0,0:INK 1,26:INK 2,6:IN
K 3,9
(NL) 220 GOSUB 1690:GOSUB 1190
(OC) 230 CLS:INK 2,13
(DJ) 240 GOSUB 1410:GOSUB 2420
(LA) 250
(FO) 260 lev=lev+1:IF lev<=5 THEN GO
(LD) SUB 3110 ELSE GOTO 3500
(LH) 270 GOSUB 3890
(DC) 280
290 FOR i=1 TO 4
(KH) 300 FOR j=1 TO 4
310 I(j,i)=0
(BE) 320 NEXT j
(PF) 330 NEXT i
(EN) 340
(CI) 350 FOR j=1 TO 15
360 x=INT(RND*4)+1:y=INT(RND*4
(JB) )+1
(NB) 370 :IF I(x,y)<>0 THEN 360
(IK) (BJ) (DF) (BF) (FN) (DA) (KN) (AC) (DN) (GM) (DA) (DH) (JM) (MK) (EJ)
```

programme

```

380 I(x,y)=INT(RND*9)+1
390 x1=FNx(x):y1=FNy(y)
400 CALL &7000,x1,y1,x1+9,y1+4
,24600+I(x,y)*400,I
410 NEXT i
420
430 GOSUB 2560
440 FOR i=1 TO 4
450 FOR j=1 TO 4
460 IF I(j,i)=0 THEN lx=j:ly=
i
470 NEXT j
480 NEXT i
490
500 SOUND 7,213,6,7
510 PEN 1
520
530 t=TIME
540 a=JOY(0):IF a=0 THEN 690
550 IF a=1 AND I(lx,ly+1)<>0 T
HEN ax=lx:ay=ly+1:rx=0:ry=-5:GO
TO 610
560 IF a=2 AND I(lx,ly-1)<>0 T
HEN ax=lx:ay=ly-1:rx=0:ry=5:GO
TO 610
570 IF a=4 AND I(lx+1,ly)<>0 T
HEN ax=lx+1:ay=ly:rx=-5:ry=0:GO
TO 610
580 IF a=8 AND I(lx-1,ly)<>0 T
HEN ax=lx-1:ay=ly:rx=5:ry=0:GO
TO 610
590 GOTO 690
600
610 m=I(ax,ay):I(ax,ay)=0
620 a1=FNx(ax):a2=FNy(ay):l1=FN
x(lx):l2=FNy(ly)
630 CALL &7000,a1,a2,a1+9,a2+4,
30000,1
640 CALL &7000,l1,l2,l1+9,l2+4,
24600+(m*400),1
650 I(lx,ly)=m:Ix=ax:ly=ay
660 zx=FNx1(PEEK(31000)):zy=FNy
1(PEEK(31001))
670 IF zx=ax AND zy=ay THEN PO
KE 31000,PEEK(31000)+rx:POKE 31
001,PEEK(31001)+ry:POKE 31003,P
EEK(31000):POKE 31004,PEEK(310
01):LOCATE PEEK(31000),PEEK(3100
1):PRINT CHR$(PEEK(31005));
680
690 CALL &7100,@f%
700 IF f%=255 THEN f%=0:sc=sc+
pu:anf=anf+1:LOCATE 2,3:PRINT U
SING "#####":sc
710 IF anf=8 THEN 260
720 ze=lim-INT(TIME-t)/300:LOCA
TE 35,25:PRINT USING "###":ze;
730 IF ze<=0 THEN GOSUB 3770:G
OTO 270
740
750 zx=FNx1(PEEK(31000)):zy=FNy
1(PEEK(31001))
760 IF zx<1 OR zx>4 OR zy<1 OR
zy>4 THEN 540
770 IF I(INT(zx),INT(zy))=0 TH
EN GOSUB 3770:GOTO 270
780 GOTO 540
790
800 REM Zeichendefinition
810 SYMBOL AFTER 0:MEMORY &61A7
820 SYMBOL 33,127,127,127,127,2
55,255,255,255
830 SYMBOL 39,255,255,255,255,1
27,127,127,127
840 SYMBOL 35,255,255,255,255,2
54,254,254,254
850 SYMBOL 36,254,254,254,254,2
55,255,255,255
860 SYMBOL 37,15,255,255,255,25
5,255,255,255
870 SYMBOL 38,240,255,255,255,2
55,255,255,255
880 SYMBOL 40,255,255,255,255,2
55,255,255,15
890 SYMBOL 41,255,255,255,255,2
55,255,255,240
(IK) 900 SYMBOL 43,1,2,4,8,16,32,64,
128
(FC)
(LA) 910 SYMBOL 61,128,64,32,16,8,4,
2,1
(CD) 920 SYMBOL 131,0,0,32,64,132,18
2,126,127
(CD) 930 SYMBOL 97,255,255,255,255
(CD) 940 SYMBOL 98,0,0,0,0,255,255,2
55,255
(CD) 950 SYMBOL 99,240,240,240,240,2
40,240,240,240
(CD) 960 SYMBOL 100,15,15,15,15,15,1
5,15,15
(CD) 970 SYMBOL 101,0,0,0,0,240,240,
240,240
(CD) 980 SYMBOL 102,0,0,0,0,15,15,15
,15
(CD) 990 SYMBOL 103,240,240,240,240
(CD) 1000 SYMBOL 104,15,15,15,15
(CD) 1010 SYMBOL 105,240,240,240,240
,255,255,255,255
(CD) 1020 SYMBOL 106,15,15,15,15,255
,255,255,255
(CD) 1030 SYMBOL 107,255,255,255,255
,15,15,15,15
(CD) 1040 SYMBOL 108,255,255,255,255
,240,240,240,240
(CD) 1050 SYMBOL 117,127,127,127,127
,127,127,127,127
(CD) 1060 SYMBOL 118,254,254,254,254
,254,254,254,254
(CD) 1070 SYMBOL 119,0,255,255,255,2
55,255,255,255
(CD) 1080 SYMBOL 120,255,255,255,255
,255,255,255
(CD) 1090 SYMBOL 121,0,31,63,127,127
,127,127,127
(CD) 1100 SYMBOL 122,0,248,252,254,2
54,254,254,254
(CD) 1110 SYMBOL 123,127,127,127,127
,127,63,31
(CD) 1120 SYMBOL 125,254,254,254,254
,254,252,248
(CD) 1130 SYMBOL 126,255,255,255,255
,255,255,255,255
(CD) 1140 SYMBOL 133,0,0,4,2,33,109,
126,254
(CD) 1150 SYMBOL 130,1,7,15,3,7,39,1
9,12
(CD) 1160 SYMBOL 132,48,200,228,224,
192,240,224,128
(CD) 1170 SYMBOL 151,24,24,255,255,2
4,24,24,24
(CD) 1180 RETURN
(CD) 1190 REM Blöcke definieren
(CD) 1200 CLS:INK 2,0:PEN 2
(CD) 1210 INK 0,0:PAPER 0:BORDER 0
(CD) 1220 RESTORE 1320
(CD) 1230
(CD) 1240 FOR i=1 TO 9
(CD) 1250 FOR j=1 TO 5
(CD) 1260 READ bl$:LOCATE 1,j:PRIN
T bl$;
(CD) 1270 NEXT j:ad=24600+i*400
(CD) 1280 CALL &7000,i,1,10,5,ad,0
(CD) 1290 NEXT i
(CD) 1300
(CD) 1310 RETURN
(CD) 1320 DATA "vwxyz","!@#$%","
","^_`~","{ }"
(CD) 1330 DATA "%&","u~v","u~
v","u~v","{ }"
(CD) 1340 DATA "%&","!~v","
v","u~v","{ }"
(CD) 1350 DATA "%&","u~v","u~
v","u~v","{ }"
(CD) 1360 DATA "vwxyz","u~v","u~
v","u~v","{ }"
(CD) 1370 DATA "vwxyz","!~v","
v","u~v","{ }"
(CD) 1380 DATA "%&","!~v","
v","u~v","{ }"
(CD) 1390 DATA "%&","!~v","
v","u~v","{ }"
(CD) 1400 DATA "%&","!~v","
v","u~v","{ }"
(CD) 1410 REM Bildschirmaufbau (PG)
(CD) 1420 PEN 3 (EJ)
(CD) 1430 sp$=SPACE$(20) (PB)
(CD) 1440 LOCATE 1,1:PRINT " PUNKTE (EO)
d c fbbbbbbe d c MAEUSE (EOD)
(CD) 1450 LOCATE 12,2:PRINT "d c d (EJB)
c d c"
(CD) 1460 LOCATE 6,3:PRINT " d (EJF)
c d laak c d c ";PEN 1:PR
INT STRING$(4,131):PEN 3 (EJG)
(CD) 1470 LOCATE 4,5:PRINT "fbbbbb" (EJH)
sp$="bbbbbbe" (EJI)
(CD) 1480 LOCATE 4,6:PRINT "d"sp$SPA (EJJ)
CE$(12)"c"
(CD) 1490 LOCATE 4,7:PRINT "d laaaa" (EJK)
sp$="aaaak c"
(CD) 1500 LOCATE 4,8:PRINT "d c " (EJL)
sp$ " d c"
(CD) 1510 LOCATE 4,9:PRINT "d c " (EJM)
sp$ " d c"
(CD) 1520 LOCATE 4,10:PRINT"d cfbbb" (EJN)
sp$="bbbed c"
(CD) 1530 LOCATE 4,11:PRINT"d cd " (EJO)
sp$ " cd c"
(CD) 1540 LOCATE 4,12:PRINT"d cd la" (EJP)
sp$="ak cd c"
(CD) 1550 LOCATE 4,13:PRINT"d cd c " (EJQ)
sp$ " d cd c"
(CD) 1560 LOCATE 4,14:PRINT"d cd c " (EJR)
sp$ " d cd c"
(CD) 1570 LOCATE 4,15:PRINT"d ij ib" (EJS)
sp$="bj ij c"
(CD) 1580 LOCATE 4,16:PRINT"d"sp$SPA (EJT)
CE$(12)"c"
(CD) 1590 LOCATE 4,17:PRINT"haak la" (EJU)
sp$="ak laag"
(CD) 1600 LOCATE 7,18:PRINT "d c "sp (EJV)
$" d c"
(CD) 1610 LOCATE 7,19:PRINT "d c "sp (EJW)
$" d c"
(CD) 1620 LOCATE 7,20:PRINT "d ib"sp (EJX)
$"bj c"
(CD) 1630 LOCATE 7,21:PRINT "d "sp (EJY)
$" c"
(CD) 1640 LOCATE 7,22:PRINT "haaa"sp (EJZ)
$"aaag"
(CD) 1650 LOCATE 12,24:PRINT "d laaa (EKA)
aaaaaaaaak c"
(CD) 1660 LOCATE 12,25:PRINT "d c (EKB)
d c 0";
(CD) 1670 LOCATE 34,23:PRINT "ZEIT:" (EKC)
(EKL)
(CD) 1680 RETURN (EKM)
(CD) 1690 REM MC-Code (EKN)
(CD) 1700 z=15000:RESTORE 1790 (EKO)
(CD) 1710 FOR i=&7000 TO &70D2 STEP (EKP)
16 (EKL)
(CD) 1720 sum=0 (EKS)
(CD) 1730 (EKT)
(CD) 1740 FOR j=i TO i+15 (EKV)
(CD) 1750 READ wert$:wert=VAL("&"+w (EKE)
ert$)
(CD) 1760 sum=sum+wert (EKF)
(CD) 1770 POKE j,wert (EKG)
(CD) 1780 NEXT j (EKH)
(CD) 1790 (EKI)
(CD) 1800 READ chsum (EKL)
(CD) 1810 IF chsum<>sum THEN PRINT (EKM)
"DATA-Fehler in Zeile ":z:END (EKN)
(CD) 1820 z=z+10 (EKO)
(CD) 1830 NEXT i (EKP)
(CD) 1840 z=16000:RESTORE 2130 (EKL)
(CD) 1850 (EKN)
(CD) 1860 FOR i=&7100 TO &72D6 STEP (EKO)
16 (EKL)
(CD) 1870 sum=0 (EKS)
(CD) 1880 FOR j=i TO i+15 (EKV)
(CD) 1890 READ wert$:wert=VAL("&"+ (EKE)
wert$)
(CD) 1900 sum=sum+wert (EKF)
(CD) 1910 POKE j,wert (EKG)
(CD) 1920 NEXT j (EKH)
(CD) 1930 READ chsum (EKL)
(CD) 1940 IF chsum<>sum THEN PRINT (EKM)
"DATA-Fehler in Zeile ":z:END (EKN)
(CD) 1950 z=z+10 (EKO)

```

programme

```

1960 NEXT i
1970
1980 RETURN
1990 DATA fe,06,c0,dd,7e,00,dd,
4e,02,dd,46,03,dd,6e,04,dd, 19
50
2000 DATA 66,06,dd,5e,08,dd,56,
0a,22,d5,70,ed,53,d3,70,32, 18
00
2010 DATA d7,70,ed,43,d8,70,21,
00,c0,3a,d4,70,4f,06,00,09, 16
00
2020 DATA 2b,22,da,70,3a,d3,70,
6f,26,00,2b,11,50,00,cd,be, 14
72
2030 DATA bd,ed,4b,da,70,09,22,
dc,70,3a,d3,70,47,3a,d5,70, 20
4f
2040 DATA 90,3c,6f,26,00,11,08,
00,cd,be,bd,22,de,70,3a,d4, 16
00
2050 DATA 70,47,3a,d6,70,90,3c,
32,e0,70,3a,d7,70,fe,00,ca, 19
98
2060 DATA 7d,70,dd,2a,d8,70,fd,
2a,dc,70,c3,85,70,dd,2a,dc, 23
78
2070 DATA 70,fd,2a,d8,70,3a,de,
70,4f,3a,e0,70,47,dd,22,e2, 21
52
2080 DATA 70,fd,22,e4,70,dd,7e,
00,fd,77,00,dd,23,fd,23,05, 20
07
2090 DATA 78,fe,00,c2,95,70,3a,
d7,70,fe,00,ca,be,70,2a,e4, 22
42
2100 DATA 70,cd,26,bc,22,e4,70,
fd,2a,e4,70,c3,cb,70,2a,e2, 23
30
2110 DATA 70,cd,26,bc,22,e2,70,
dd,2a,e2,70,0d,79,fe,00,c8, 21
04
2120 DATA c3,89,70,00,00,00,00,
00,00,00,00,00,00,00,00,00, 04
44
2130 DATA 3a,1a,79,cd,9e,72,cd,
77,72,cd,75,bb,cd,60,bb,fe, 23
71
2140 DATA 20,ca,b2,71,fe,2b,ca,
b5,71,fe,3d,ca,34,72,fe,96, 24
05
2150 DATA ca,bb,72,3a,1d,79,3d,
fe,81,c2,2e,71,3e,85,32,21, 17
86
2160 DATA 79,3a,1a,79,3d,fe,00,
c2,3c,71,3e,04,32,1e,79,cd, 14
80
2170 DATA 9e,72,cd,77,72,cd,75,
bb,cd,60,bb,fe,20,ca,7e,71, 24
34
2180 DATA 3a,1a,79,3c,fe,05,c2,
5b,71,3e,01,32,1e,79,3a,1d, 12
73
2190 DATA 79,3c,fe,86,c2,69,71,
3e,82,32,21,79,3a,1e,79,cd, 17
91
2200 DATA 9e,72,cd,77,72,cd,75,
bb,cd,60,bb,fe,20,c0,3a,1e, 22
73
2210 DATA 79,32,1a,79,3a,21,79,
32,1d,79,3a,18,79,67,3a,19, 11
19
2220 DATA 79,6f,cd,75,bb,3e,20,
cd,5d,bb,3a,1c,79,32,19,79, 17
23
2230 DATA 6f,3a,1b,79,32,18,79,
67,cd,75,bb,3a,1d,79,cd,5d, 16
30
2240 DATA bb,c9,c3,7e,71,3a,1a,
79,fe,01,ca,dd,71,fe,03,ca, 22
77
2250 DATA dd,71,f5,3a,21,79,3d,
fe,81,c2,ce,71,3e,85,32,21, 20
26
2260 DATA 79,f1,3d,fe,00,c2,f8,
71,3e,04,c3,f8,71,f5,3a,21, 21

```

```

[FC] 90
[AC] 2270 DATA 79,3c,fe,86,c2,e9,71,
3e,82,32,21,79,f1,3c,fe,05, 20
65
[BC] 2280 DATA c2,f8,71,3e,01,c3,f8,
71,f5,3a,1b,79,67,3a,19,79, 19
29
[AK] 2290 DATA 6f,cd,75,bb,3e,20,cd,
5d,bb,3a,1a,79,cd,9e,72,3a, 19
39
[BG] 2300 DATA 1b,79,32,18,79,3a,1c,
79,32,19,79,f1,f5,cd,9e,72, 17
09
[CC] 2310 DATA 3a,1b,79,32,18,79,3a,
1c,79,32,19,79,f1,32,1a,79, 12
42
[KI] 2320 DATA 32,1e,79,c9,3a,1a,79,
fe,01,ca,5c,72,fe,03,ca,5c, 18
21
[NI] 2330 DATA 72,f5,3a,21,79,3c,fe,
86,c2,4d,72,3e,82,32,21,79, 18
00
[IL] 2340 DATA f1,3c,fe,05,c2,f8,71,
3e,01,c3,f8,71,f5,3a,21,79, 21
91
[BJ] 2350 DATA 3d,fe,81,c2,68,72,3e,
85,32,21,79,f1,3d,fe,00,c2, 20
05
[KK] 2360 DATA f8,71,3e,04,c3,f8,71,
7d,fe,00,c2,84,72,2e,19,7d, 19
98
[CL] 2370 DATA 32,1c,79,c9,fe,1a,c0,
7c,fe,0d,ca,97,72,fe,1c,ca, 22
14
[IC] 2380 DATA 97,72,2e,19,c3,23,71,
2e,01,7d,32,1c,79,c9,07,4f, 13
37
[BP] 2390 DATA 06,00,21,fe,7c,09,3a,
18,79,86,32,1b,79,23,3a,19, 10
79
[MM] 2400 DATA 79,86,32,1c,79,6f,3a,
1b,79,67,c9,dd,6e,00,dd,66, 17
29
[CC] 2410 DATA 01,3e,ff,77,c3,7e,71,
00,00,00,00,00,00,00,00,00, 08
71
[FD] 2420 REM Maus-Daten
[OL] 2430 FOR i=30000 TO 32500
2440 POKE i,0
2450 NEXT i
[BD] 2460 RESTORE 2530
2470
[LB] 2480 FOR i=1 TO 4
2490 READ a,b
2500 POKE 31998+i*2,a
2510 POKE 31998+i*2+1,b
[AG] 2520 NEXT i
2530 DATA 0,255,1,0,0,1,255,0
2540 RETURN
[CB] 2550
2560 RANDOMIZE TIME
[LI] 2570 :IF 1(1,1)<>1 AND 1(1,1)<>
5 AND 1(1,1)<>6 THEN b=1:GOTO 2
620
[HB] 2580 :IF 1(4,1)<>1 AND 1(4,1)<>
5 AND 1(4,1)<>6 THEN b=2:GOTO 2
620
[NE] 2590 :IF 1(1,4)<>1 AND 1(1,4)<>
3 AND 1(1,4)<>4 THEN b=3:GOTO 2
620
[LG] 2600 :IF 1(4,4)<>1 AND 1(4,4)<>
3 AND 1(4,4)<>4 THEN b=4:GOTO 2
620
[IG] 2610 b=INT(RND*4)+1
2620 ON b GOSUB 2680,2690,2700,
2710
[EG] 2630 f%=0
2640 POKE 31000,x:POKE 31001,y:
POKE 31002,r
2650 POKE 31003,x:POKE 31004,y:
POKE 31006,r
[BN] 2660 POKE 31005,chr:POKE 31009,c
h
[OD] 2670 RETURN
2680 x=13:y=1:r=3:ch=132:RETURN
2690 x=28:y=1:r=3:ch=132:RETURN

```

```

[ND] 2700 x=13:y=25:r=1:ch=130:RETUR
N
[DM] 2710 x=28:y=25:r=1:ch=130:RETUR
N
[LD] 2720 REM Ende
[AM] 2730 MODE 1:GOSUB 4320
[CL] 2740 LOCATE 9,1:PRINT CHR$(24);
"H I - S C O R E "+CHR$(24);
[KB] 2750 GOSUB 3080
[PN] 2760 LOCATE 12,17:PRINT "SCORE:
";:PRINT USING "#####":SC;
[IL] 2770 IF SC<hi(6) THEN 3050
[KG] 2780 h$="ABCDEFGHIJKLMNPOQRSTUVWXYZ.
"+CHR$(24)+"!";
[BN] 2790 LOCATE 3,20:PRINT H$
[LP] 2800 LOCATE 17,21:PRINT "^";:
[AO] 2810
[JH] 2820 A=JOY(0):IF A=0 THEN 2820
[IN] 2830 FOR I=1 TO 100:NEXT
2840 IF A=8 THEN H$=RIGHT$(H$
,29)+LEFT$(H$,1):GOTO 2880
[KP] 2850 IF A=4 THEN H$=RIGHT$(H$
,1)+LEFT$(H$,29):GOTO 2880
[JH] 2860 IF A=16 THEN 2910
[BI] 2870 GOTO 2820
[AE] 2880 LOCATE 3,20:PRINT H$:
[JH] 2890 GOTO 2820
[PO] 2900
[AK] 2910 N1$=MID$(H$,15,1)
[CO] 2920 IF N1$=CHR$(242) AND LEN(
N$)>0 THEN N$=LEFT$(N$,LEN(N$)-
1):LOCATE 13,22:PEN 2:PRINT N$:
":GOTO 2950
[DA] 2930 IF N1$="!" THEN 2970
[BJ] 2940 IF LEN(N$)<16 AND N1$<>CH
R$(242) THEN N$=N$+N1$:PEN 2:LO
[GF] 2950 GOTO 2820
[GB] 2960
[AI] 2970 FOR I=5 TO 1 STEP -1
[PB] 2980 IF SC>HI(I) THEN NEXT I
[MH] 2990 FOR J=5 TO 1 STEP -1
[MB] 3000 HI(J+1)=HI(J):HI$(J+1)=HI
$(J)
[CL] 3010 NEXT J
[ED] 3020 HI(I+1)=SC:HI$(I+1)=N$
[EH] 3030
[AF] 3040 GOSUB 3080:tf=1:GOSUB 3200
[CF] 3050 LOCATE 13,25:PRINT "PRESS
FIRE";
[FB] 3060 WHILE JOY(0)<>16:WEND:f%=0
:lev=0:anf=0:iiv=4:sc=0:GOTO 41
10
[NI] 3070
[PJ] 3080 FOR I=1 TO 6
[NI] 3090 LOCATE 5,1*2+3:PRINT HI$(
I):STRING$(16,32);:LOCATE 25,1*
2+3:PRINT USING "#####":HI(I);:
[BP] NEXT I
[EB] 3100 RETURN
[AB] 3110 REM Level
[ML] 3120 anf=0:GOSUB 3830
3130 ON lev GOSUB 3350,3380,341
0,3440,3470
3140
[JJ] 3150 FOR i=350 TO 32 STEP -2
[GA] 3160 PLOT 160,i,0:DRAW 478,i,0
3170 NEXT i
[GP] 3180 pu=lev*100
[EM] 3190
3200 ENT -1,1,2,1,1,0,2,1,0,2,1
,-2,1,1,0,4
[ND] 3210 ENV 1,1,40,1,1,-3,2,1,0,1,
1,0,1,12,-1,4
[AL] 3220 RESTORE 3330
[OB] 3230 READ a,b:IF a=-1 THEN IF t
f=0 THEN 3270 ELSE RETURN
[CB] 3240 SOUND 1,a,b*10,7,1,1:SOUND
2,a*2,b*10,7,1,1:SOUND 4,a/2,b
*10,7,1,1
[ED] 3250 GOTO 3230
[FL] 3260
[AA] 3270 a$=SPACE$(20)+CHR$(133)+"-
LEVEL: "+CHR$(150)+SPACE$(8)
[PG] 3280 INK 1,26:PEN 1:FOR i=1 TO
38
[IL]

```


programme

```

3290 LOCATE 11,12:PRINT LEFT$(
a$,20);
3300 a$=RIGHT$(a$,37)+LEFT$(a$,
.1)
3310 NEXT i
3320
3330 DATA 319,7,284,1,268,7,338
,1,319,3,284,1,268,3,210,1,213,
3,319,1,268,3,213,1,225,12,268,
1,319,1,358,1,319,16,-1,-1
3340 RETURN
3350 SYMBOL 150,108,254,254,254
,124,56,16 'HERZ
3360 Iim=250
3370 RETURN
3380 SYMBOL 150,24,60,60,60,126
,255,24 'GLOCKE
3390 Iim=250
3400 RETURN
3410 SYMBOL 150,0,2,5,253,229,1
62,160 'SCHLUESSEL
3420 Iim=200
3430 RETURN
3440 SYMBOL 150,2,12,20,36,98,1
02,6 'KIRSCHEN
3450 Iim=150
3460 RETURN
3470 SYMBOL 150,0,36,60,126,219
,126,102,60 'GESICHT
3480 Iim=100
3490 RETURN
3500 REM (Level>5)
3510
3520 MODE 1:BO=5000
3530 S$=CHR$(133)+"SIE HABEN AL
LE LEVEL GESCHAFFT ...."
3540 S1$=CHR$(133)+"DAFUER BEKO
MMEN SIE 5000 PUNKTE BONUS. "
3550 S$=SPACE$(37)+S$:S1$=SPACE
$(40)+S1$
3560
3570 FOR I=1 TO 38
3580 LOCATE 3,5:PRINT LEFT$(S$,
37);S$=RIGHT$(S$,LEN(S$)-1)
3590 NEXT I
3600
3610 FOR I=1 TO 40
3620 LOCATE 1,10:PRINT LEFT$(S
1$,40)::S1$=RIGHT$(S1$,LEN(S1$)
-1)
3630 NEXT I
3640
3650 LOCATE 14,20:PRINT "SCORE:
":PRINT USING "#####":SC:PRI
NT " P.";
3660 LOCATE 14,15:PRINT "BONUS:
":PRINT USING "#####":BO:PRI
NT " P.";
3670
3680 FOR I=0 TO 5000 STEP 100
3690 SOUND 7,213,2,7,1,1
3700 LOCATE 21,20:PRINT USING
"#####":SC:I;
3710 LOCATE 21,15:PRINT USING
"#####":BO-I;
3720 NEXT I
3730
3740 SC=SC+5000
3750 GOTO 2720
3760 END
3770 REM Tod
3780 Iiv=Iiv-1:anf=0
3790 LOCATE 34,3:PRINT STRING$(
Iiv,131);STRING$(4-Iiv,151);
3800 IF Iiv=0 THEN 2720
3810 GOSUB 2560:GOSUB 3110
3820 FOR i=0 TO 5:FOR j=0 TO 5:
I(i,j)=0:NEXT j,i
3830 FOR i=1 TO 11
3840 IF f(i)=0 THEN 3860
3850 LOCATE x(i),y(i):PRINT " "
:
3860 f(i)=0
3870 NEXT i
3880 RETURN
3890 REM Gegenstaende
3900 FOR i=1 TO 11
3910 f(i)=0
3920 NEXT i
3930
3940 INK 1,26:FEN 1
3950 FOR i=1 TO 8
3960 r=INT(RND*11)+1
3970 :IF f(r)=1 THEN 3960
3980 LOCATE x(r),y(r):PRINT CH
R$(150)::f(r)=1
3990 NEXT i
4000
4010 RETURN
4020 DIM x(11),f(11),y(11)
4030 RESTORE 4100
4040
4050 FOR i=1 TO 11
4060 READ x(i),y(i)
4070 NEXT i
4080
4090 RETURN
4100 DATA 5,6,8,15,10,11,13,24,
33,11,20,2,13,2,28,24,31,11,33,
17,36,6
4110 REM Titelbild
4120 t$=CHR$(164)+" B/86 BY M.P
IES UND J.SCHULTE"+SPACE$(18)
4130 MODE 1:INK 0,0:INK 1,0:INK
2,0:INK 3,0:SPEED INK 7,7
4140 SYMBOL 240,0,0,32,64,132,1
82,126,127
4150 FOR i=1 TO 40 STEP 2:LOCAT
E i,1:FEN 1:PRINT CHR$(143)::LO
CATE i,25:PRINT CHR$(143)::NEXT
4160 FOR i=2 TO 40 STEP 2:FEN 2
:LOCATE i,1:PRINT CHR$(143)::LO
CATE i,25:PRINT CHR$(143)::NEXT
4170 FOR i=2 TO 24 STEP 2:LOCAT
E 1,i-1:FEN 1:PRINT CHR$(143)::
LOCATE 40,i-1:PRINT CHR$(143)::
NEXT
4180 FOR i=2 TO 24 STEP 2:LOCAT
E 1,i:FEN 2:PRINT CHR$(143)::LO
CATE 40,i:PRINT CHR$(143)::NEXT
4190 FEN 3:FOR i=1 TO 100:LOCAT
E INT(RND*38)+2,INT(RND*23)+2:P
RINT CHR$(240)::NEXT
4200 RESTORE 4270:FOR i=1 TO 5
4210 READ m,n,a:MOVE m,n:FOR j
=1 TO a:READ x,y:DRAW x,y,2:NE
XT j,i
4220 INK 0,0:INK 1,26:INK 2,6
,26:INK 3,6:PAPER 0:BORDER 0:PE
N 1
4230 LOCATE 15,20:PRINT LEFT$(t
$,18);
4240 t$=RIGHT$(t$,LEN(t$)-1)+LE
FT$(t$,1)
4250 IF INKEY$<>" " THEN 210
4260 GOTO 4230
4270 DATA 50,150,4,0,100,50,-50
,50,50,0,-100
4280 DATA 210,150,4,-50,0,0,50,
50,0,0,-50
4290 DATA 230,200,3,0,-50,50,0,
0,50
4300 DATA 340,200,5,-50,0,0,-25
,50,0,0,-25,-50,0
4310 DATA 400,150,5,-50,0,0,50,
50,0,0,-25,-50,0
4320 RESTORE 4340:FOR I=1 TO 6:
READ A$,A:HI$(I)=A$:HI(I)=A:NEX
T I
4330 N$="":RETURN
4340 DATA JEMI-SOFT,6000,JEMI-S
OFT,5000,JEMI-SOFT,4000,JEMI-SO
FT,3000,JEMI-SOFT,2000,JEMI-SOF
T,1000
ENDE DES LISTINGS

```

TRACER

Tolle Programmierhilfe für alle, die mit dem CPC in Maschinensprache programmieren wollen!

Wer kennt das nicht: gerade hat man seinen ersten Computer erworben und freut sich, wenn man z.B. mit PRINT "A" den ersten sichtbaren Programmiererfolg, und schon ist man an die Grenzen der Programmiersprache Basic gelangt (vor allem wegen der langsamen Geschwindigkeit). Aber vor Maschinenspracheprogrammierung scheut man sich doch sehr: schwer zu erlernen und sehr fehleranfällig. Daher muß ein Werkzeug her, welches dem Programmierer bei der Fehlersuche hilft. Und dieses Werkzeug ist das nachstehend abgedruckte Programm TRACER.

"Trace" heißt soviel wie "kleine Schritte machen". Mit diesem TRACER ist es nun möglich, nur einen einzigen Maschinenbefehl

(Opcode) auszuführen, um sich dann sofort die Auswirkungen auf die Register des Mikroprozessors oder auf bestimmte Speichereinhalte zu Gemüte führen zu können.

Sobald Sie den TRACER gestartet haben, können Sie eines der folgenden Kommandos eingeben:

- O - One Step**
Der angezeigte Opcode wird ausgeführt
- G - Go**
Sie werden zunächst nach der Adresse gefragt, bei der der TRACER sich wieder zu Wort melden soll. Bis zu dieser Adresse wird das Programm in Echtzeit durchlaufen. Wichtig: Die Adresse des Breakpoints muß in einem Bereich liegen,

in dem gerade das RAM und nicht das ROM aktiv ist.

- C - Call**
Die durch CALL aufgerufene Routine wird in Echtzeit durchlaufen. Der TRACER meldet sich dann direkt hinter dem CALL-Befehl zurück.
- R - Register**
Zunächst Eingabe des Registernamens, dann des neuen Registerinhaltes. Abbruch durch ESC.
- M - Memory**
Zunächst wird die Eingabe einer Startadresse erwartet. Dann können der Inhalt dieser und der nachfolgenden Speicherzellen verändert werden.

N - Next

Der aktuelle Opcode wird übersprungen. Wichtig: die Befehle HALT und EXX sollten grundsätzlich übersprungen werden.

B - Basic on/off

Schaltet zwischen BASIC-ROM und VIDEO-RAM im Bereich &C000 bis &FFFF um.

P - Pointer

Die vier Zeiger im "Memory-

Monitor" können verändert werden.
Abbruch durch ESC.

D - Disassemble

Zunächst muß eine Startadresse eingegeben werden. Ab dieser Adresse wird der Speicherinhalt seitenweise disassembliert. Eine neue Seite erhält man durch Betätigen einer beliebigen Taste. Auch hier wieder Abbruch durch ESC.

Q - Rücksprung ins Basic

Hinweise zum Abtippen:

Tippen Sie zunächst das kleine Basicprogramm aus Listing 1 ab und sichern es auf Kassette oder Diskette (Kassette noch nicht zurückspulen!). Geben Sie dann NEW ein. Nun kommt die Geduldprobe. Tippen Sie Listing 2 ab und lassen es laufen. Wenn Sie sich irgendwo vertippt haben, wird Ihnen das umgehend mitgeteilt, ansonsten wird der zweite Programmteil "TRACE" automatisch aufgenommen (er muß auf der Kassette direkt hinter dem

ersten Programmteil, den man zweckmäßigerweise "TRACER" nennen sollte, folgen). Lockern Sie nun Ihre Finger. Spulen Sie die Kassette zurück und Starten TRACER. Wenn alles O.K. ist, erscheint zunächst ein Titelbild und kurze Zeit später meldet sich der TRACER persönlich.

Sollten Sie den TRACER einmal über Q(uit) verlassen haben, so können Sie ihn mit dem neuen RSX-Befehl "Itrace" jederzeit wieder erreichen.

```

10 MEMORY &93FF
15 MODE 2
20 BORDER 6: INK 0,26: INK 1,2
25 LOCATE 1,10
30 PRINT "*****"
  **      **** *****
"
35 PRINT "*****"
  **** ** ** * ** *
"
40 PRINT "*****"
  ** ** ** * ** *
"
45 PRINT "*****"
  ** ** ** * ** *
"
50 PRINT "*****"
  **** ** ** * ** *
"
55 PRINT "*****"
  ** ** ** * ** *
"
60 PRINT "*****"
  ** ** ** * ** *
"
65 LOAD "Itrace"
70 CALL &7400

```

```

100 DATA 3E,C9,32,00,94,21,1C,9
4,01,11,94,CD,D1,BC,C3,20,-1665
110 DATA 94,16,94,C3,20,94,54,5
2,41,43,C5,00,00,00,00,-1188
120 DATA CD,E6,94,AF,CD,B4,BB,C
3,5B,9C,CD,DE,95,CD,99,99,-2856
130 DATA CD,F1,95,CD,BA,99,CD,6
5,99,AF,CD,B4,BB,CD,6C,BB,-2845
140 DATA 11,89,A2,CD,D2,95,CD,D
3,94,20,0B,11,8C,A2,CD,D2,-2221
150 DATA 95,CD,D3,94,2B,EA,CD,5
C,99,FE,4F,11,EA,A1,21,FC,-2467
160 DATA 9A,2B,60,FE,47,11,F4,A
1,21,AA,9C,2B,56,FE,43,11,-1860
170 DATA F8,A1,21,9B,9C,2B,4C,F
E,52,11,FE,A1,21,8B,9A,2B,-2045
180 DATA 42,FE,4D,11,0B,A2,21,7
4,9A,2B,3B,FE,4E,11,10,A2,-1510
190 DATA 21,47,9C,2B,2E,FE,42,1
1,16,A2,21,80,9C,2B,24,FE,-1514
200 DATA 50,11,24,A2,21,30,9A,2
8,1A,FE,44,11,2D,A2,21,0B,-1183
210 DATA 9A,2B,10,FE,51,11,3A,A
2,21,DF,9C,2B,06,11,9F,A2,-1578
220 DATA 21,2A,94,E5,D5,3E,07,C
D,B4,BB,CD,6C,BB,D1,CD,D2,-2430
230 DATA 95,E1,E9,FD,21,00,0B,C
D,09,BB,B7,C0,FD,2B,FD,ES,-2455
240 DATA E1,7C,B5,20,F2,C9,3E,0
2,CD,0E,BC,AF,01,0D,0D,CD,-1883
250 DATA 32,BC,3E,01,01,00,00,C
D,32,BC,DD,21,CD,A2,06,07,-1379
260 DATA 0E,0C,DD,5E,00,DD,23,D
D,56,00,DD,23,DD,6E,00,DD,-1712
270 DATA 23,DD,66,00,DD,23,CD,C
0,BB,DD,5E,00,DD,23,DD,56,-2076
280 DATA 00,DD,23,DD,6E,00,DD,2
3,DD,66,00,DD,23,C5,CD,F9,-2073

```

```

{CE} 290 DATA BB,C1,0D,20,E4,05,20,C
{DM} 8,06,0B,11,57,A1,CD,D2,95,-1736
{GA} 300 DATA 10,FB,DD,21,AD,A2,C5,7
{BB} 8,CD,B4,BB,DD,56,00,DD,23,-2308
310 DATA DD,66,00,DD,23,DD,5E,0
0,DD,23,DD,6E,00,DD,23,CD,-1942
320 DATA 66,BB,C1,04,7B,FE,0B,2
0,DD,AF,32,B5,F0,32,B5,FB,-2246
330 DATA 32,3B,A4,11,8F,A2,CD,D
2,95,CD,D2,95,21,30,00,11,-1821
340 DATA 3E,A4,01,0B,00,ED,80,D
D,21,30,00,3E,C3,11,33,33,-1326
350 DATA 21,27,9C,DD,72,00,DD,7
3,01,DD,77,02,DD,75,03,DD,-1804
360 DATA 74,04,21,E0,A4,22,0A,A
1,2A,3C,A4,7E,32,47,A4,0E,-1437
370 DATA 00,CD,0F,B9,3E,01,32,3
B,A4,3E,07,CD,B4,BB,11,79,-1520
380 DATA A2,CD,D2,95,DD,21,00,C
0,DD,2B,DD,ES,E1,7C,B5,20,-2448
390 DATA F7,C9,1A,13,CD,5A,BB,B
7,20,F8,C9,3E,03,CD,B4,BB,-2276
400 DATA CD,6C,BB,DD,21,0B,A1,D
D,4E,00,DD,46,01,CD,26,96,-1907
410 DATA C9,3E,02,CD,B4,BB,CD,6
C,BB,DD,21,0B,A1,DD,4E,0E,-2073
420 DATA 60,09,AF,CD,5A,BB,3E,2
0,CD,5A,BB,05,CB,01,7B,FE,-1831
430 DATA 06,2B,F8,FE,04,2B,F4,F
E,00,CB,3E,20,CB,09,CB,01,-1800
440 DATA 30,E1,3E,2A,1B,DD,F5,D
5,DD,ES,E5,21,A1,9D,CD,F2,-2557
450 DATA 97,7E,EE,10,E6,70,C2,3
3,9B,7E,ES,11,1E,9B,D5,E6,-2267
460 DATA 7F,FE,20,D2,AD,9B,FE,0
7,20,02,1B,04,FE,0E,20,00,-1571
470 DATA 3D,CA,F4,96,3D,CA,F9,9
6,3D,CA,F5,96,3D,CA,05,97,-2396
480 DATA 11,00,9B,D5,11,07,03,2
1,F6,9C,3D,CB,3D,CA,0F,97,-1534
490 DATA 21,24,9D,3D,CB,21,30,9
D,3D,CB,21,4A,9D,3D,CB,16,-1533
500 DATA 09,3D,CB,11,03,04,21,5
5,9D,3D,CB,21,5D,9D,3D,CB,-1374
510 DATA 21,99,9D,3D,CB,21,24,9
D,16,03,3D,CB,21,65,9D,1E,-1437
520 DATA 07,3D,CB,21,89,9D,11,0
3,06,3D,CB,3D,CB,3D,CB,D1,-1613
530 DATA 21,4A,9D,3D,CA,18,97,3
D,CA,21,97,3D,CA,6E,97,3D,-1734
540 DATA CA,6B,97,3D,CA,72,97,3
D,CA,2F,97,D1,3D,CA,79,97,-2190
550 DATA D1,03,11,70,9B,D5,11,F
2,97,D5,21,31,9F,3D,CB,21,-1864
560 DATA 3E,9F,3D,CB,D1,D1,3D,C
A,B9,97,21,EF,A0,03,3D,CA,-2149
570 DATA AC,97,3D,76,03,0A,C3,9
C,9B,03,03,0A,CD,9C,9B,0B,-1558
580 DATA 0A,03,C3,9C,9B,3E,20,C
D,AD,9B,3E,20,C3,AD,9B,3E,-1963
590 DATA 3A,9F,9D,E6,7F,C3,AD,9
B,0B,11,07,09,CD,00,9B,03,-1655
600 DATA C9,0A,CB,3F,CB,3F,CB,3
F,F6,30,E6,37,C3,AD,9B,03,-2111
610 DATA 0A,CB,7F,20,04,3E,2B,1
B,02,3E,2D,CD,AD,9B,0A,CB,-1357

```

```

{ID} 620 DATA 7F,2B,04,ED,44,E6,FF,C
D,BC,97,C9,F5,C5,E5,0A,03,-2390
{AB} 630 DATA 6F,CB,7D,2B,04,26,FF,1
B,02,26,00,09,7C,CD,9C,9B,-1486
{FA} 640 DATA 7D,CD,9C,9B,E1,C1,F1,C
9,0A,E6,3B,C3,9C,9B,03,C3,-2495
{FJ} 650 DATA 4B,97,0B,0B,CD,2F,97,0
3,C9,21,55,9D,11,03,04,0B,-1165
{MN} 660 DATA 0B,CD,00,9B,03,03,C3,1
E,9B,21,2B,A0,0B,0A,03,CB,-1214
{HH} 670 DATA 6F,2B,04,3E,59,1B,02,3
E,5B,F6,80,32,9F,9D,CD,F2,-1669
{GH} 680 DATA 97,7E,E6,7F,FE,1D,C2,7
0,9B,C3,39,96,CD,F2,97,7E,-2501
{II} 690 DATA FE,9F,C2,84,9B,0B,21,2
B,9F,C3,31,9B,F5,FE,64,3B,-2188
{HD} 700 DATA 12,16,01,D6,64,F2,CB,9
7,14,1B,F8,5F,7A,F6,30,CD,-1959
{BD} 710 DATA AD,9B,7B,16,00,D6,0A,F
A,DD,97,14,1B,F8,5F,F1,FE,-2198
{JN} 720 DATA 0A,3B,06,7A,F6,30,CD,A
D,9B,7B,C6,0A,F6,30,CD,AD,-2021
{CD} 730 DATA 9B,C9,7E,23,B7,F2,F2,9
7,0A,AE,23,A6,23,20,F3,C9,-2228
{GG} 740 DATA 0A,1F,15,20,FC,A3,3C,5
7,7E,23,B7,F2,0B,9B,15,20,-1455
{JF} 750 DATA F7,7E,E6,7F,CD,AD,9B,7
E,23,B7,F2,11,9B,C9,E1,7E,-2567
{BI} 760 DATA 23,B7,F2,39,96,3E,0D,C
D,AD,9B,03,E1,DD,E1,D1,F1,-2396
{EG} 770 DATA C9,0B,0B,C5,E5,16,09,C
D,94,9B,7E,E6,7F,FE,20,30,-2002
{CJ} 780 DATA 1C,FE,02,20,0B,CD,94,9
B,CD,94,9B,1B,10,FE,01,2B,-1669
{LA} 790 DATA F7,FE,15,2B,F3,FE,17,2
B,EF,FE,1B,2B,EB,7E,23,B7,-2258
{CC} 800 DATA F2,3A,9B,3E,20,CD,AD,9
B,15,20,FB,E1,C1,C3,39,96,-2197
{CM} 810 DATA 7E,FE,9F,20,04,03,C3,B
6,97,C5,E5,0B,16,09,CD,94,-1927
{AB} 820 DATA 9B,C3,37,9B,C5,E5,0B,0
B,0B,16,09,CB,94,9B,CD,94,-1902
{JB} 830 DATA 9B,C3,37,9B,15,15,0A,C
D,9C,9B,03,C9,F5,1F,1F,1F,-1661
{DF} 840 DATA 1F,CD,AS,9B,F1,E6,0F,C
6,90,27,CE,40,27,CD,5A,BB,-2211
{DN} 850 DATA C9,AF,21,39,A4,77,23,7
7,7E,CD,9C,9B,2B,7E,CD,9C,-2072
{BD} 860 DATA 9B,11,4D,A1,CD,D2,95,C
D,06,BB,CD,5C,99,FE,FC,CB,-2525
{JD} 870 DATA FE,0D,CB,D6,30,3B,FO,F
E,17,30,EC,FE,0A,3B,06,FE,-2166
{NF} 880 DATA 11,3B,E4,D6,07,ED,6F,2
3,ED,6F,1B,CC,26,2E,2E,-1657
{CE} 890 DATA 7C,CD,5A,BB,7D,CD,5A,B
B,11,4F,A1,CD,D2,95,CD,06,-2245
{LA} 900 DATA BB,CD,5C,99,FE,FC,CB,F
E,0D,2B,04,65,6F,1B,E1,01,-2116
{EK} 910 DATA 00,0B,DD,21,3D,A1,DD,7
E,00,BC,20,05,DD,7E,01,DD,-1593
{DD} 920 DATA CB,DD,23,DD,23,0C,0C,1
0,ED,1B,C1,21,4A,A4,36,00,-1531
{JM} 930 DATA 7E,CD,9C,9B,11,4F,A1,C
D,D2,95,CD,06,BB,CD,5C,99,-2308
{DL} 940 DATA FE,FC,CB,FE,0D,CB,D6,3
0,3B,FO,FE,17,30,EC,FE,0A,-2556

```

programme

950 DATA 3B,06,FE,11,3B,E4,D6,07,ED,6F,1B,D4,FE,61,D8,FE,-2243	(BCB)	E7,FE,20,2B,6D,E6,C7,FE,C7,-2553	(BGJ)	1540 DATA 43,20,20,41,AC,53,55,42,20,A0,53,42,43,20,20,41,-1139	(KGG)
960 DATA 7B,D0,D6,20,C9,3E,05,C D,B4,BB,CD,6C,BB,3E,06,CD,-2190	(CCI)	1260 DATA 2B,70,FE,C0,2B,73,FE,C2,2B,77,FE,C4,2B,7B,C3,03,-2171	(CJI)	1550 DATA AC,41,4E,44,20,A0,5B,4F,52,20,A0,4F,52,20,20,A0,-1401	(DKJ)
970 DATA B4,BB,CD,6C,BB,ED,4B,4B,A4,16,0E,3E,05,CD,B4,BB,-2090	(CAD)	1270 DATA 9C,EB,16,00,5A,C3,E0,9B,21,15,A1,35,20,02,1E,00,-1409	(PEE)	1560 DATA 43,50,20,20,A0,4E,DA,DA,4E,C3,C3,50,CF,50,CS,DO,-2125	(PFJ)
980 DATA 7B,CD,9C,9B,79,CD,9C,9B,3E,06,CD,B4,BB,CD,26,96,-2300	(CBA)	1280 DATA 2A,0B,A1,23,23,16,00,7B,07,D2,E0,9B,16,FF,C3,E0,-1718	(EGD)	1570 DATA CD,52,4C,43,C1,52,52,43,C1,52,4C,C1,52,52,C1,44,-1B23	(EDJ)
990 DATA 3E,0A,CD,5A,BB,15,20,E3,C9,3E,01,CD,B4,BB,CD,6C,-1983	(CBK)	1290 DATA 9B,D5,2A,0A,A1,ED,5B,0B,A1,13,13,13,2B,72,2B,73,-1450	(MI)	1580 DATA 41,C1,43,50,CC,53,43,C6,43,43,C6,C2,C3,C4,C5,C8,-2271	(GI)
1000 DATA BB,06,0B,DD,21,0B,A1,DD,7E,01,CD,9C,9B,DD,7E,00,-1832	(CJB)	1300 DATA 22,0A,A1,E1,1B,CC,2A,0A,A1,5E,23,56,23,22,0A,A1,-1326	(DP)	1590 DATA CC,2B,4B,4C,A9,C1,42,C3,44,C5,4B,CC,53,DO,42,C3,-2108	(EO)
1010 DATA CD,9C,9B,DD,23,DD,23,10,EE,C9,3E,04,CD,B4,BB,CD,-2323	(CJG)	1310 DATA EB,1B,BF,7B,2A,0E,A1,1B,09,7B,2A,0C,A1,1B,03,2A,-1230	(CP)	1600 DATA 44,C5,4B,CC,41,C6,52,4C,43,20,A0,52,52,43,20,A0,-1644	(LD)
1020 DATA 6C,BB,DD,21,4C,A4,06,04,DD,7E,01,67,CD,9C,9B,DD,-19B4	(CJP)	1320 DATA 10,A1,FE,E9,C2,03,9C,1B,A9,7B,B7,C2,60,9B,1E,00,-19B8	(PA)	1610 DATA 52,4C,20,20,A0,52,52,20,20,A0,53,4C,41,20,A0,53,-1269	(GK)
1030 DATA 7E,00,6F,CD,9C,9B,DD,23,DD,23,11,52,A1,CD,D2,95,-20B6	(CEP)	1330 DATA 1B,AE,2B,2B,7E,E6,3B,1B,1B,7B,B7,20,C9,1E,FF,1B,-15B9	(AN)	1620 DATA 52,41,20,A0,AC,53,52,4C,20,A0,AC,42,49,54,20,A0,-1531	(FJ)
1040 DATA 0E,0B,ES,7E,CD,9C,9B,23,0D,20,FB,E1,3E,20,CD,5A,-1832	(CHI)	1340 DATA 9F,7B,B7,20,BC,1E,01,1B,97,7B,B7,20,A4,1E,01,1B,-1394	(CJJ)	1630 DATA 52,45,53,20,A0,53,45,54,20,A0,42,C3,44,C5,49,DB,-1669	(CK)
1050 DATA BB,0E,0B,7E,23,FE,80,30,04,FE,20,30,02,3E,2E,CD,-1453	(CME)	1350 DATA BF,ED,5B,0B,A1,1B,1B,ED,53,0B,A1,16,00,5F,1B,91,-1469	(PM)	1640 DATA 53,DO,00,FF,4E,4F,DO,01,CF,4C,44,20,04,0B,2C,26,-1392	(IA)
1060 DATA 5A,BB,0D,20,EE,10,C1,C9,AF,CD,B4,BB,CD,6C,BB,11,-2234	(CEG)	1360 DATA 19,22,0B,A1,C3,5B,9C,ES,C5,4F,2A,16,A1,ES,3E,C2,-18B5	(NH)	1650 DATA B2,03,CF,49,4E,43,04,8B,09,CF,41,44,44,04,4B,4C,-1270	(DI)
1070 DATA 26,A1,CD,D2,95,CD,B1,9B,CD,6C,BB,ED,4B,39,A4,ED,-2567	(CFD)	1370 DATA B0,47,F1,7B,32,FC,9B,79,C1,E1,06,00,00,00,9C,C9,-1967	(BD)	1660 DATA 2C,8B,0B,CF,44,45,43,04,8B,04,C7,49,4E,43,04,89,-1310	(DL)
1080 DATA 43,4B,A4,CD,65,99,CD,06,BB,FE,FC,CA,BD,94,1B,EF,-2468	(CFF)	1380 DATA 06,FF,C9,ED,73,54,A4,2A,16,A1,E5,F1,ED,7B,0A,A1,-22B8	(MC)	1670 DATA 05,C7,44,45,43,04,89,06,C7,4C,44,20,04,09,2C,26,-1025	(BD)
1090 DATA AF,CD,B4,BB,FD,21,4C,A4,AF,ED,47,3E,4D,CD,5A,BB,-2377	(CGB)	1390 DATA FD,2A,0C,A1,DD,2A,0E,A1,2A,10,A1,ED,5B,12,A1,ED,-1869	(KM)	1680 DATA B1,07,C7,8B,02,FF,4C,44,20,04,2B,42,43,29,2C,C1,-1359	(CC)
1100 DATA ED,57,CD,9C,9B,11,52,A1,CD,D2,95,CD,B1,9B,FE,FC,-2701	(CMD)	1400 DATA 4B,14,A1,00,00,00,00,ED,73,0A,A1,FD,22,0C,A1,DD,-1460	(DD)	1690 DATA 12,FF,4C,44,20,04,2B,44,45,29,2C,C1,22,FF,4C,44,-1341	(MI)
1110 DATA CA,BD,94,CD,6C,BB,2A,39,A4,FD,75,00,FD,23,FD,74,-2329	(CPG)	1410 DATA 22,0E,A1,22,10,A1,ED,53,12,A1,ED,43,14,A1,ED,7B,-1764	(MI)	1700 DATA 20,04,2B,26,02,29,2C,4B,CC,32,FF,4C,44,20,04,2B,-1002	(NI)
1120 DATA 00,FD,23,CD,BA,99,AF,CD,B4,BB,ED,57,3C,FE,04,20,-2253	(CPH)	1420 DATA 54,A4,FS,E1,22,16,A1,2A,0B,A1,3A,4B,A4,16,00,5F,-1560	(AM)	1710 DATA 26,02,29,2C,C1,0A,FF,4C,44,20,04,41,2C,2B,42,43,-1045	(IL)
1130 DATA CB,C3,BD,94,AF,CD,B4,BB,11,26,A1,CD,D2,95,CD,B1,-2641	(GL)	1430 DATA 19,22,0B,A1,2A,3C,A4,3A,47,A4,77,CD,DB,95,C5,E1,-1901	(GL)	1720 DATA A9,1A,FF,4C,44,20,04,41,2C,2B,44,45,A9,2A,FF,4C,-1458	(CH)
1140 DATA 9B,FD,2A,39,A4,AF,CD,B4,BB,CD,6C,BB,FD,E5,E1,7C,-2746	(CLC)	1440 DATA ED,4B,0B,A1,A7,ED,42,7D,4D,32,4B,A4,06,04,21,23,-1520	(KH)	1730 DATA 44,20,04,4B,4C,2C,2B,26,02,A9,3A,FF,4C,44,20,04,-103B	(EN)
1150 DATA CD,9C,9B,7D,CD,9C,9B,11,52,A1,CD,D2,95,CD,2B,99,-2376	(CBM)	1450 DATA 9C,E5,36,00,23,10,FB,2A,0B,A1,D1,ED,80,C3,ED,94,-2106	(CJJ)	1740 DATA 41,2C,2B,26,02,A9,76,FF,4B,41,4C,D4,40,C0,4C,44,-1556	(KB)
1160 DATA FE,FC,CA,5B,9C,3A,4A,A4,FD,77,00,FD,23,CD,DB,95,-24B4	(GHP)	1460 DATA 3A,3B,A4,EE,01,32,3B,A4,2B,0B,0E,00,CD,0F,B9,C3,-1455	(ID)	1750 DATA 20,04,09,2C,8A,80,C0,05,8A,C0,C7,52,45,54,04,87,-1455	(GK)
1170 DATA CD,BA,99,CD,65,99,1B,CD,AF,CD,B4,BB,CD,6C,BB,11,-2496	(FE)	1470 DATA BD,94,CD,03,B9,C3,BD,94,3A,23,9C,FE,CD,CA,03,9C,-2331	(CJB)	1760 DATA C2,C7,4A,50,20,04,07,2C,26,82,C4,C7,43,41,4C,4C,-14B1	(IJ)
1180 DATA 6E,A2,CD,D2,95,CD,EC,9B,FE,FC,CA,5B,9C,DD,21,0B,-2646	(CEL)	1480 DATA E6,C7,FE,C4,CA,03,9C,C3,ED,94,AF,CD,B4,BB,11,94,-26B4	(BM)	1770 DATA 20,07,2C,26,82,C6,C7,05,26,81,C7,C7,52,53,54,04,-1471	(LO)
1190 DATA A1,06,00,DD,09,DD,ES,CD,6C,BB,11,34,A1,CD,D2,95,-2141	(CF)	1490 DATA A2,CD,D2,95,CD,B1,9B,FE,FC,CA,BD,94,2A,39,A4,7E,-2694	(CJG)	1780 DATA 26,96,C1,CF,50,4F,50,04,8C,C5,CF,50,55,53,4B,20,-1727	(CJ)
1200 DATA CD,B1,9B,CD,6C,BB,DD,E1,2A,39,A4,DD,75,00,DD,74,-241B	(PK)	1500 DATA ED,5B,0B,A1,32,47,A4,22,3C,A4,22,0B,A1,36,F7,3E,-1606	(NE)	1790 DATA BC,C3,FF,4A,50,20,04,26,82,D3,FF,4F,55,54,04,2B,-1706	(FH)
1210 DATA 01,CD,99,99,CD,DB,95,CD,F1,95,1B,BC,2A,0B,A1,7E,-2229	(DJ)	1510 DATA C3,EB,32,23,9C,22,24,9C,AF,32,4B,A4,C3,03,9C,21,-1748	(LK)	1800 DATA 26,01,29,2C,C1,E3,FF,45,5B,20,04,2B,53,50,29,2C,-12B0	(PB)
1220 DATA E6,E7,FE,20,7E,2B,04,E6,3B,1B,02,E6,1B,47,7E,23,-1715	(BL)	1520 DATA 3E,A4,11,30,00,01,0B,00,ED,80,3E,02,CD,0E,BC,0E,-1198	(GC)	1810 DATA 4B,CC,F3,FF,44,C9,C9,FF,52,45,D4,E9,FF,4A,50,20,-2536	(ME)
1230 DATA 5E,23,56,CD,E7,9B,FE,C3,2B,37,FE,10,2B,3A,FE,1B,-1996	(IF)	1530 DATA 00,CD,0F,B9,C3,64,C0,41,44,44,20,20,41,AC,41,44,-1527	(HN)	1820 DATA 04,2B,4B,4C,A9,F9,FF,4C,44,20,04,53,50,2C,4B,CC,-1527	
1240 DATA 2B,3E,FE,CD,2B,4B,FE,C9,2B,5C,FE,DD,2B,65,FE,FD,-23B6					
1250 DATA 2B,67,FE,E9,2B,69,E6,					

8	1830 DATA DB,FF,49,4E,20,04,41,2C,2B,26,01,A9,EB,FF,45,5B,-166	{KF}	2120 DATA 9D,86,C7,05,2B,49,06,1B,A9,34,FF,49,4E,43,04,2B,-137	06,13,1A,40,1B,1B,04,00,10,-453	{MC}
5	1840 DATA 20,04,44,45,2C,4B,CC,FB,FF,45,C9,CD,FF,43,41,4C,-193	{FF}	2130 DATA 49,06,1B,A9,35,FF,44,45,43,04,2B,49,06,1B,A9,36,-115	2470 DATA 00,00,00,DC,FF,01,00,00,-626	{CH}
7	1850 DATA 4C,20,26,82,0B,FF,45,5B,20,04,41,46,2C,41,46,A7,-121	{JD}	2140 DATA FF,4C,44,20,04,2B,49,06,1B,29,2C,26,B1,76,FF,9F,-136	2480 DATA 00,00,00,22,00,01,00,FE,FF,00,00,E0,FF,00,00,00,-102	{LL}
3	1860 DATA D9,FF,45,5B,DB,10,FF,44,4A,4E,5A,20,26,95,1B,FF,-192	{MC}	2150 DATA 7B,FF,9F,70,FB,4C,44,20,04,2B,49,06,1B,29,2C,93,-144	2490 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,-0	{AP}
4	1870 DATA 4A,52,20,04,26,95,20,C7,4A,52,20,04,0E,2C,26,95,-104	{AH}	2160 DATA 46,C7,4C,44,20,04,09,2C,2B,49,06,1B,A9,00,00,9F,-973	2500 DATA 00,10,00,3E,00,00,00,56,00,4B,01,00,00,00,AA,-407	{BA}
7	1880 DATA CB,FF,9A,ED,FF,9B,DD,DD,9C,00,00,44,45,46,42,20,-216	{IL}	2170 DATA 36,FF,9F,06,C7,0F,2B,49,06,17,A9,06,07,11,14,2C,-109	2510 DATA FF,01,00,00,00,00,00,54,00,01,00,FE,FF,00,00,AC,-102	{BJ}
2	1890 DATA 26,83,37,FB,9F,3F,C0,0F,BA,FF,00,10,14,2C,8A,70,-162	{AJ}	2180 DATA 2B,49,06,17,A9,00,00,9F,00,00,00,00,00,00,00,-470	2520 DATA FF,B6,FE,00,00,01,00,FE,FF,49,01,00,00,00,FE,-152	{DB}
4	1900 DATA FF,9F,40,C7,49,4E,20,04,09,2C,2B,43,A9,71,FF,9F,-172	{CC}	2190 DATA 00,00,00,00,00,00,FF,00,42,72,65,61,6B,61,64,72,-105	2530 DATA FF,B9,FE,00,00,2E,00,9E,00,00,00,46,00,6A,00,00,-107	{EC}
0	1910 DATA 41,C7,4F,55,54,04,2B,43,29,2C,89,42,CD,53,42,43,-133	{IC}	2200 DATA 65,73,73,3A,20,00,53,74,61,72,74,61,64,72,65,73,-147	2540 DATA 00,00,00,BA,FF,01,00,00,00,00,44,00,01,00,FE,-765	{KB}
2	1920 DATA 04,4B,4C,2C,8B,63,FF,9F,43,CF,4C,44,20,04,2B,26,-138	{FJ}	2210 DATA 73,3A,20,00,44,61,74,61,3A,20,20,20,00,50,43,53,-967	2550 DATA FF,00,00,BA,FF,96,FF,00,00,FF,FF,02,00,6A,00,00,-171	{HL}
0	1930 DATA 02,29,2C,99,44,FF,4E,45,C7,45,FF,52,45,54,CE,46,-174	{KG}	2220 DATA 50,49,59,49,5B,4B,4C,44,45,42,43,41,46,0B,0B,-980	2560 DATA 00,00,00,02,00,96,FF,00,00,1E,00,DE,00,00,00,A6,-825	{HJ}
4	1940 DATA FF,49,4D,20,04,B0,47,FF,4C,44,20,04,49,2C,C1,4D,-151	{PD}	2230 DATA 0B,00,20,20,20,20,00,0E,10,0F,01,1F,0B,02,52,45,-374	2570 DATA 00,BA,00,00,00,00,00,5A,FF,01,00,FE,FF,00,00,A6,-115	{AL}
1	1950 DATA FF,52,45,54,C9,56,FF,49,4D,20,04,B1,57,FF,4C,44,-188	{KP}	2240 DATA 47,49,53,54,45,52,1F,07,03,50,43,1F,07,04,53,50,-855	2580 DATA 00,01,00,FE,FF,00,00,5A,FF,76,FF,00,00,FF,FF,02,-174	{NA}
1	1960 DATA 20,04,41,2C,C9,5E,FF,49,4D,20,04,B2,67,FF,52,52,-158	{MP}	2250 DATA 1F,07,05,49,59,1F,07,06,49,5B,1F,07,07,48,4C,1F,-634	2590 DATA 00,8A,00,00,00,FF,FF,02,00,76,FF,00,00,7A,00,4E,-122	{DE}
1	1970 DATA C4,6F,FF,52,4C,C4,A0,FF,4C,44,C9,A1,FF,43,50,C9,-244	{PK}	2260 DATA 07,0B,44,45,1F,07,09,42,43,1F,07,0A,41,46,1F,07,-553	2600 DATA 01,00,00,26,00,FB,00,00,00,00,00,DA,FF,01,00,FE,-101	{GE}
0	1980 DATA A2,FF,49,4E,C9,A3,FF,4F,55,54,C9,AB,FF,4C,44,C4,-239	{AE}	2270 DATA 0C,20,20,43,4F,4E,44,49,54,49,4F,4E,20,20,1F,0B,-85B	2610 DATA FF,00,00,26,00,01,00,FE,FF,00,00,DA,FF,0B,FF,00,-153	{RG}
9	1990 DATA A9,FF,43,50,C4,AA,FF,49,4E,C4,AB,FF,4F,55,54,C4,-240	{PJ}	2280 DATA 0E,53,20,5A,20,4B,20,56,20,4E,20,43,1F,0C,11,4D,-787	2620 DATA 00,FF,FF,02,00,FB,00,00,00,00,FF,FF,02,00,0B,FF,00,-153	{CK}
9	2000 DATA 80,FF,4C,44,49,D2,B1,FF,43,50,49,D2,B2,FF,49,4E,-230	{MK}	2290 DATA 45,4D,4F,52,59,20,4D,4F,4E,49,54,4F,52,1F,05,12,-103	2630 DATA 00,B6,00,C2,00,00,00,36,00,A4,00,00,00,00,CA,-796	{IJ}
4	2010 DATA 49,D2,B3,FF,4F,54,49,D2,BB,FF,4C,44,44,D2,B9,FF,-246	{BM}	2300 DATA 4D,30,30,1F,05,13,4D,30,31,1F,05,14,4D,30,32,1F,-664	2640 DATA FF,01,00,FE,FF,00,00,36,00,01,00,FE,FF,00,00,CA,-153	{DB}
4	2020 DATA 43,50,44,D2,BA,FF,49,4E,44,D2,BB,FF,4F,54,44,D2,-217	{JL}	2310 DATA 05,15,4D,30,33,1F,1B,17,43,4F,4D,4D,41,4E,44,4F,-870	2650 DATA FF,5C,FF,00,00,FF,FF,02,00,A4,00,00,00,FF,FF,02,-179	{DD}
8	2030 DATA 4A,CF,41,44,43,04,4B,4C,2C,8B,6B,FF,9F,4B,CF,4C,-169	{JK}	2320 DATA 20,54,41,42,4C,45,1F,02,1B,2A,4F,6E,65,20,53,74,-101	2660 DATA 00,5C,FF,00,00,5E,01,34,00,00,00,1A,01,1E,01,00,-552	{JJ}
5	2040 DATA 44,20,04,0B,2C,2B,26,02,A9,00,00,9F,09,CF,41,44,-916	{DL}	2330 DATA 65,70,00,2A,47,6F,00,2A,43,61,6C,6C,00,2A,52,65,-10B	2670 DATA 00,00,00,D0,FE,01,00,FE,FF,00,00,30,01,01,00,FE,-127	{CF}
5	2050 DATA 44,04,49,06,2C,8D,21,FF,4C,44,20,04,49,06,2C,26,-965	{KG}	2340 DATA 67,69,73,74,65,72,00,2A,4D,65,6D,6F,72,79,00,2A,-137	2680 DATA FF,00,00,D0,FE,F6,FF,00,00,00,00,02,00,0A,00,00,-123	{CB}
1	2060 DATA 82,22,FF,4C,44,20,04,2B,26,02,29,2C,49,86,23,FF,-126	{OM}	2350 DATA 4E,65,7B,74,00,2A,42,61,73,69,63,20,6F,6E,2F,6F,-135	2690 DATA 00,00,00,02,00,F6,FF,00,00,00,00,00,F7,8B,00,00,-934	{NJ}
1	2070 DATA 49,4E,43,04,49,86,2B,FF,44,45,43,04,49,86,2A,FF,-143	{DI}	2360 DATA 66,66,00,2A,50,6F,69,6E,74,65,72,00,2A,44,69,73,-131	2700 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,10,-16	{ED}
9	2080 DATA 4C,44,20,04,49,06,2C,2B,26,02,A9,E1,FF,50,4F,50,-127	{FD}	2370 DATA 61,73,73,65,6D,62,6C,65,00,2A,51,75,69,74,00,2A,-134	2710 DATA 00,20,00,30,00,00,C3,FF,3F,9F,CF,E7,F3,F9,FD,FF,-219	{MM}
1	2090 DATA 04,49,86,E5,FF,50,55,53,4B,20,49,86,E3,FF,45,5B,-189	{EL}	2380 DATA 1F,13,03,4F,50,43,4F,44,45,1F,16,04,20,20,1F,20,-679	2720 DATA C3,F3,F3,F3,F3,F3,C3,FF,E7,C3,B1,E7,E7,E7,E7,FF,-359	{PN}
3	2100 DATA 04,20,2B,53,50,29,2C,49,86,F9,FF,4C,44,20,04,53,-129	{FD}	2390 DATA 03,4D,4E,45,4D,4F,4E,49,43,1F,1B,0B,49,4E,50,55,-986	2730 DATA 00,-0	{IA}
8	2110 DATA 50,2C,49,86,E9,FF,4A,50,20,04,2B,49,06,A9,CB,FF,-175	{EL}	2400 DATA 54,20,54,41,42,4C,45,1F,01,19,1B,12,1B,00,52,65,-782	2740 MEMORY &93FF	{FJ}
5		{EL}	2410 DATA 67,69,73,74,65,72,3A,20,00,42,79,20,55,77,65,20,-130	2750 adr=&9400	{CB}
		{FD}	2420 DATA 53,63,6B,61,65,66,65,72,00,8F,0B,00,20,08,00,0E,-100	2760 FOR i=10 TO 273	{CF}
		{MI}	2430 DATA 01,0F,00,00,42,72,65,61,6B,20,61,74,3A,20,00,49,-909	2770 READ op\$:IF LEFT\$(OP\$,1)="-" THEN 2800	{BB}
		{BK}	2440 DATA 6E,70,75,74,20,43,6F,6D,6D,61,6E,64,00,1B,29,0B,-126	2780 op=VAL("&"+op\$):POKE adr,op	{FJ}
		{LO}	2450 DATA 0B,0A,0D,02,09,06,12,0E,0E,11,2C,03,03,09,29,11,-231	2790 adr=adr+1:psum=psum+op:GOTO 2770	{GD}
			2460 DATA 14,2E,31,06,13,34,4E,06,13,1A,40,1B,1B,04,00,10,-453	2800 IF psum+VAL(op\$)=0 THEN 2820	{LG}
				2810 PRINT "Checksum-Error in Zeile ";i*10:END	{GM}
				2820 psum=0:NEXT	{CJ}
				2830 SAVE "trace",b,&9400,&1070	{FI}
					{OK}
					{NI}

Endlich ist er da!

Der neue Checksummer für die Schneider-Computer CPC 464/ 664/ 6128

Worauf alle Schneider-User bisher sehnsüchtig gewartet haben, das wird jetzt an dieser Stelle veröffentlicht: Eine erheblich verbesserte Checksummerversion. Nun ist es möglich, schon bei der Eingabe einer Basiczeile festzustellen, ob ein Eingabefehler begangen wurde.

Ab dieser Ausgabe finden Sie hinter jeder Zeile eine zweistellige Buchstabenkombination, die von geschweiften Klammern eingerahmt wird. Anhand dieser Prüfsumme kann die Eingabe jetzt auf Ihre Richtigkeit hin untersucht werden.

Um ein beliebiges TRONIC-

Programm abzutippen, gehen Sie wie folgt vor: Laden Sie zuerst den Checksummer und starten Sie diesen mit "RUN". Auf die Frage nach dem verwendeten Computertyp antworten Sie bitte mit der entsprechenden Computerbezeichnung (464/ 664/ 6128) und geben Sie diese ein. Anschließend wird der Checksummer aktiviert und der BASIC-LOADER automatisch gelöscht. Nun können Sie mit der Eingabe des Programmes beginnen. Ist eine Zeile komplett eingegeben, so wird diese wie üblich mit "ENTER" abgeschlossen. Unterhalb der Zeile erscheint daraufhin die Checksum-

me. Stimmt diese mit der im Heft abgedruckten Summe überein, so wurde diese Zeile fehlerfrei eingegeben.

Stimmen diese beiden Zeilen nicht überein, liegt ein Tippfehler vor, den Sie wie folgt korrigieren müssen. Dabei ist bei der Eingabe des Listings einiges zu beachten: Starten Sie niemals ein Programm, wenn der Checksummer aktiv ist. Speichern Sie das Programm nach der Eingabe sofort ab. Die Checksummen selbst werden nicht eingetippt. Sie können bei der Eingabe sowohl Groß- als auch Kleinbuchstaben verwenden. Bis auf eine Ausnah-

me können alle Editiermöglichkeiten verwendet werden. Ausnahme: Eine Zeile kann nicht durch die Eingabe der Zeilennummer gelöscht werden. Um beispielsweise die Zeile 230 zu löschen geben Sie DELETE 200 ein. Leerzeichen die keine Bedeutung haben werden selbstständig ignoriert und gelöscht. Allerdings besitzen Leerzeichen innerhalb von Anführungsstrichen einen Wert und verändern somit die Checksumme. Für den PRINT-Befehl kann auch das Fragezeichen verwendet werden. Viel Spaß und fehlerfreies Eingeben wünscht OTTFRIED SCHMIDT!

```

1 *****
2 * SCHNEIDER CPC *
3 * CHECKSUMMER V 2.0 *
4 * (C) 1984 *
5 * OTTFRIED SCHMIDT *
6 *****
7
8
9000 DATA 3E,FF,32,00,AC,CD,22,BF,F5,C5
1010 DATA D5,E5,2A,20,BF,CD,61,DD,B7,2B
1020 DATA 54,E5,2A,20,BF,CD,6E,A0,E1,30
1030 DATA 4A,CD,04,EE,CD,A3,E7,CD,63,E1
1040 DATA 2A,20,BF,06,FF,AF,4F,7E,FE,00
1050 DATA 2B,05,B1,AB,23,10,F5,B1,AB,4F
1060 DATA 3E,0D,CD,5A,BB,3E,0A,CD,5A,BB
1070 DATA 3E,7B,CD,5A,BB,79,E6,FO,1F,1F
1080 DATA 1F,1F,C6,41,CD,5A,BB,79,E6,0F
1090 DATA C6,41,CD,5A,BB,E1,D1,C1,F1,3E
1100 DATA 7D,C3,5A,BB,C9,E1,D1,C1,F1,C9
1110 DATA CD,61,DD,B7,37,CB,CD,04,EE,DO
1120 DATA 7E,FE,20,20,01,23,CD,D2,E6,37
1130 DATA 9F,C9,00,00,00,00,00,00,00
1140 MEMDRY &9FFF:FOR i=&A000 TO &A0B5
1150 READ a$:POKE i,VAL("&"+a$):NEXT i
1160 CLS:INPUT"BENUTZEN SIE EINEN CPC 46
4/664 oder 6128";v$
1170 IF v$<>"464" AND v$<>"664" AND v$<>
"6128" THEN PRINT CHR$(7):GOTO 1160
1180 IF v$="464" THEN adr=&BD3A:pufa=&A4
ELSE IF v$="664" THEN adr=&BD5B:pufa=&B
A ELSE adr=&BD5E:pufa=&BA
1190 POKE &BF20,pufa:POKE &BF21,&AC
1200 POKE &BF22,PEEK(adr):POKE &BF23,PEE
K(adr+1):POKE &BF24,PEEK(adr+2)
1210 POKE adr,&C3:POKE adr+1,0:POKE adr+
2,&A0
1220 IF v$="664" THEN GOTO 1250
1230 IF v$="6128" THEN GOTO 1300
1240 GOTO 1340
1250 POKE &A010,&52:POKE &A011,&DE:POKE
&A06F,&52:POKE &A070,&DE
1260 POKE &A020,&D4:POKE &A021,&EE:POKE
&A075,&D4:POKE &A076,&EE
1270 POKE &A07F,&AA:POKE &A080,&E7:POKE
&A023,&96:POKE &A024,&EB
1280 POKE &A026,&59:POKE &A027,&E2
1290 GOTO 1340
1300 POKE &A010,&4D:POKE &A011,&DE:POKE
&A06F,&4D:POKE &A070,&DE
1310 POKE &A020,&CF:POKE &A021,&EE:POKE
&A075,&CF:POKE &A076,&EE
1320 POKE &A07F,&A5:POKE &A080,&E7:POKE
&A023,&64:POKE &A024,&EB
1330 POKE &A026,&54:POKE &A027,&E2
1340 CLS:PRINT"CHECKSUMMER V2.0 IST JETZ
T AKTIV!"
1350 NEW

```


Leser- & Meckerecke

Sehr geehrte Damen und Herren.

Ich besitze einen Commodore C 16 Computer und möchte vielleicht mal zum 128 aufsteigen. Deshalb möchte ich mir gerne wissen, ob man das Diskettenlaufwerk 1571 auch mit dem C 16 betreiben kann.

Bärwaldt, Remscheid

Redaktion: Im Prinzip geht dies, aber wissen Sie schon genau welchen Computer Sie in einem Jahr kaufen wollen?

Liebe Redaktion

Seit längerem schon hatte ich vor, Ihnen einmal zu schreiben, jedoch ließ mir meine Umschulung ... keine Zeit dazu. Aber jetzt. Es fing damit an, daß ein Bekannter sich einen C-16 gekauft hatte und auch so langsam auf den Gedanken kam, mir auch einen zu-

zulegen und es dann auch in die Tat umsetzte.

Zu fast dem gleichen Zeitpunkt wurde ich auf die "Compute mit" aufmerksam. ... Interessant was in "Compute mit" zu lesen war und zu was mein "Kleiner" alles im Stande ist. Ihr Artikel in der Sonderausgabe 3/86 "Beim C-16/116 ist noch viel drin" war dann hauptsächlich Auslöser dieses Schreibens und zum Öffnen meines C-16. Ja, ich habe nicht schlecht gestaunt.

Das Öffnen gestaltete sich aufgrund einer wohl überdrehten Schraube ein wenig schwierig, doch kam ich dann zum Erfolg. Er war offen. Eigentlich sollte sich über der Platine ein Lochblech zur Wärmeableitung und zum Schutz gegen Störstrahlen befinden, welcher sich bei meinem C-16 als Pappe mit einem Hauch von Metall und lediglich 6

Löchern über dem Festspannungsregler VR1 erwies!

Beim Abziehen des Steckers mit den Leitungen von der Tastatur kommend, zeigte sich mir eine Leitung als derbe gequetscht. Der Stecker der Leuchtdiode wurde abgezogen und ich machte mutig weiter.

Die Pappe ließ sich nach Abziehen einer Halteklammer am Gehäuse nach vorn klappen, was dann folgte, dürfte jedem Elektroniker einen Schlag versetzen. Extrem verbogene Bauteile (Dioden, Kondensatoren), deren Beinchen so dicht zusammen waren, daß es einen wundert, daß es nicht zu Kurzschlüssen kommt. Als erfreulich stellte sich heraus, daß der TED-CHIP ein 8360/R2 ist. Bei näherer Betrachtung stellte ich fest, daß das Gehäuse zu ca. 3/5 unter der Rechnerplatine Lüftungsschlitze besitzt, die jedoch

nutzlos sind, weil oben genannte Pappe vollständig dicht auch zwischen Gehäuse und Rechnerplatine zu finden ist.

Weiterhin waren dann noch einige Chips zu finden, deren Bezeichnung im Stromlaufplan, der ja zu meinem Gerät gehören soll, etwas von den tatsächlichen Bezeichnungen abgewichen....

... Mai 86 gekauft, Aufdruck auf Platine: 1984.

Was habe ich bloß für ein olles Ding erwischt?

...

Baut man ein Gerät, in dem sich Bauteile befinden, vor allem der TED-CHIP derart dicht, wundert es mich gar nicht, wenn diese nach kurzem "abkacken". Da helfen auch Kühlkörper nicht mehr viel.

**Harald Hollstein,
Recklinghausen**

Kleinanzeigen

Kontakte

Suche C16-User aus dem Kreis Offenbach (Keine Spieler). Manfred Sorg, Tel. 0 61 02 / 3 42 57, Di-Fr von 15.00-18.00 Uhr

C-16, -116, +4-Anwender, Einsteiger Endlich ein Club speziell für uns. Kostenlose Information bei hrc e.V., Lashinsky, Bauernland 15, 4800 Bielefeld 1

Hallo C-16-Besitzer in Moers und Umgebung. Suche Möglichkeit zum Erfahrungsaustausch - bzw. gemeinsames Arbeiten am C16. Benutze Floppy. Bin selbst Einsteiger. Tel. 028 41 / 5 59 12

Commodore mini-club Postfach 1314, 4150 Krefeld 29 Wir suchen auch Inserenten für unsere Zeitung. Infos kostenlos!

Der Commodore mini-club mit seinen 80 Mitgliedern sucht immer weiter Commodore-User, die viel für wenig Geld verlangen. Bei uns bekommt man für 2,50 DM im Monat eine Menge! Wir bieten: mtl. Zeitung, Kontakte, Tips & Tricks u.v.m. Holt Euch das kostenlose Info oder/und schaut mal in unsere Mailbox. Tel.-Nr. 028 45 / 87 14 tgl. von 22-6 Uhr ONLINE! Info gibt's bei Cmc, Postfach 1314, 4150 Krefeld 29.

Ab sofort ist eine neue Mailbox in Betrieb. Die Mailbox arbeitet mit 8 Datenbytes, 1 Stopbyte sowie keine-Parität und ist unter der Nummer 091 91 / 1 44 72 zu erreichen. Ich benutze einen C-64-Rechner, der mit bis zu 4 Floppys ausgebaut werden kann. Die Box ist 24 Stunden Online im Einsatz und, soweit bekannt, die einzige in Oberfranken. Der Sysop der Hellas-Box freut sich auf zahlreicher Anrufe. Anton Beyer, Wilhelm-Busch-Str. 10, 8550 Forchheim, Tel. 0 91 91 / 8 99 47

An alle Atari-Besitzer!!! Unser Club sucht noch Mitglieder. Jeder muß zwar einen Beitrag bezahlen, wird aber dafür entschädigt, da er die Club-Zeitung GOTO erhält. Wir verfügen über 67 Spiele. Derjenige, der nicht eintreten möchte, kann auch die Zeitung bestellen. Anschrift: Gruner Udo, Dietrichstraße 22, 5650 Solingen 1

Wer hat billige Software für den +4? (Diskette) Wer ist an einer Clubgründung mit Softwareaustausch interessiert? Wer möchte in diesem Rahmen Erfahrungen austauschen? Infos wenn möglich mit Rückporto an Gilbert Kunz, 6600 Saarbrücken, Dudweiler Landstr. 115

Wer hätte Lust mit mir seine besten Programme für CPC 464 zu tauschen? Bitte meldet Euch bei: Stefan Basfeld, Dresdener Str. 6, 4134 Rheinberg

Suche

ATARI XL ★ SOS ★ ATARI XL ★ SOS Suche dringend DOS-2.0 für Atari-XL. Wer kann mir helfen? Dieter Müller, Rottanger 2, 3402 Scheden, Tel. 055 46 / 602

C-16 Preiswerte Anwendungsprog. gesucht. Mit ausführlicher Beschreibung. Ingo Weinhausen, Weddelbrook 39, 2354 Hohenwedstedt, Tel. 0 48 71 / 39 33

Suche geb. guterhaltenen C64 mit Datasette und Joysticks. Angebote an Thomas Tauber, Tel. 0 60 87 / 6 84

HALLO VC20 FANS Ich suche Spiele für die VC20 Grundversion. Zahle gut. Tausche auch Spiele. Schickt Eure Angebote an Mike Hoffmann, Winkelweg 8, 8521 Grossenseebach

Tronic-Verlag
Postfach 870
3440
Eschwege

NEU:

Hardwareservice des Tronic-Verlags

Unser 1. Angebot:

64 K-Speichererweiterung für C16 + C116

Bausatz - bestehend aus: allen Bauteilen und ausführlicher Anleitung, umschaltbar 64 K/16 K - geliefert in stabiler Hartbox.

Bitte Computertyp angeben! Bestell-Nr. HWS-1-786 45,- DM + NN (bei Vorkasse + 2,- DM Versand)

kleinanzeigen

Achtung C16, C116, Plus 4-User!!
Verkaufe Disk vollgepackt mit 24 Spielen und 9 Anwenderprogr. für 10 DM (Poker, Mastermind, Helikopter, Cannon, Kniffel, usw.). Schickt einen 10-Mark-Schein an Michael Grahneis, Stefanstr. 1, 5551 Monzelfeld, und Ihr erhaltet umgehend die Diskette. Tel. 0 65 31 / 39 49

Suche ROM-Listing für C16
Keuchel, Pillauer Str. 14, 2300 Kiel 14, Tel. 04 31 / 20 33 53 nach 16.30 Uhr

C-16/116 Software gesucht ★★
(Tausch oder Ankauf) Melden bei F. Mesters, Am Gerkenstein 3, 5788 Winterberg 7, Tel. 02981 - 1858 P.S.: Nur Tape!!!!!!!

Biete Software

Software für C64 und Schneider
Software Tape und Disk für 14 DM, z.B. 1942 - Paperboy - Jack the Nipper - Flightdeck - Thai Boxing - Hole in One - Ghost'n Goblins - Art Studio - Mission Elevator - Now Games II - Alter Ego - Biggles - Fist II - Super Cyrl - Dracons Lair - War - Icups - Parallax - Trap - Arac - Outlaws - Roomio - Iridisia - Hacker II - Knight Games. Thomas-Soft, Tsamadou 4, Athina - Greece

Verkaufe CPC 464 Originalspiele
wie Gyroscopoe, Locomotion u.a. Ingo Massenber, 4044 Kaarst 2, Aachener Str. 7, Tel. 02101 / 514817 Suche CPC 464 Floppy!!!

★ AMIGA SOFTWARE AMIGA ★
Suche und tausche Software. Suche Kontakte zu Amiga User. Liste an A. Palmes, 5 Köln 60, Schmiedegasse 24, Tel. 02 21 - 740 6194, 100% Antwort.

HALLO C-16-FANS!!! SUPER-SOFTWARE, Biete LOTTOSPIEL, CAR-RACING, FLIPPER und andere Programme an. Alle garantiert selbstgeschrieben. Gebe auch Programmierhilfen!!!
Kontaktadresse:
Armin Rasche, Am Nocken 40, 5885 Schalksmühle.
Also Info anfordern oder sofort bestellen!!!

ATARI 800 XL
Suche und biete Software für Atari (Disk bevorzugt). Schickt Liste an: Reiner Kuen, Schmiedstr. 8, 8901 Wehringen

★★★★ **SPRITE-KING** ★★★★★
★ Über 40 Funktionen (Trick, ★
★ Negativ ...) max. 64 Sprites ★
★ aus Prg. kopieren + bearbei- ★
★ ten. Disk + Anl. nur 35,- DM + ★
★ NN Info gegen Freiumschlag ★
★ bei WISH-Software, Weseler ★
★ Landweg 2, 4290 Bocholt ★
★★★★ **SPRITE-KING** ★★★★★

C-16, C-116 Superspiele!
Info gegen Rückporto bei Hannes Kallenbach, Prielmayerstr. 16, 7990 Friedrichshafen 1

C 64, C 128, C 16/116, VC 20, 4+
Textverarbeitung ab 9,90
6 Spiele 19,90
Denktraining, 2 Stck 19,90
Vokabeltrainer 19,90
Adressverwaltung 29,90
Karteikasten 19,90
Katalog gegen 80-Pf.-Briefmarke.
Computerservice T. Hofstede,
A.d. Windmühle 8, 5010 Bergheim 5

Verkaufe über 20 Originalspiele für C16, z.B. SOCCER, SKRAMBLE, ROCKMAN, und so. Spielekassette von Robtek. Bitte melden bei René Reinthal, Tel. 02223 / 2890 ab 14 Uhr

C16/116-Grundversion
TEXTADVENTURE (Eigenproduktion) Werner Barwig - TAPE-NACHLADER Bodelschwingstr. 19 - Vorkasse - 6418 Hünfeld 1 - 10-DM-Schein

! SCHNEIDER CPC !
3"-Disketten!!! Braxx Bluff 30.- / 3D-Boxing 30.- / GAL 90.- / Der blaue Kristall 60.- / Hacker 45.- / Tomahawk 50.- / Fairlight 53.- ! und ANDERE!!! Tel. 07231 / 48429 (18 Uhr)

C-16/116 Besitzer aufgepaßt!
Für nur 20 DM versende ich 20 Spiel- und Anwenderprogramme! Schein oder Scheck bitte an: Daniel Beittlich, Kleine Höllbergstr. 5, 6000 Frankfurt 50. Oder Infoblatt für 80 Pf. Rückporto anfordern. Alle 29 verschiedenen Programme sind auf Kasette!

C64/C16 - preisw. Progr.-Pakete für Schule + Beruf - z.B. C64: Techn. Hydrau. usw. 1 Disk ★ Simons Basic-Prg. Pak. 2 Disk voll - Super Graf. + Math. Prg. 20 DM Vorkasse o. Mehrkosten! Für C16: Techn. Mathe + Schulnaw. + Super Grafik-Progr. Disk/Cass. Katalog 1 Dm m. Comp.-Angabe. Softw.-Vers. A. Ristau, Peetzweg 9, 3320 Salzg. 1

★★★★ **C16/116/+4 Software** ★★★★★
Yie are Kung Fu 24,90
Winter Olympiade 24,90
Projekt Nova 22,90
Mercenary 26,90
Bomb Jack 19,90
Frank Bruno's Boxing 19,90
Ghost and Goblins 27,90
A.C.E. (auch 64) 29,-
Comp. Service, PF 1304, 7913 Senden

★★★★ **C16/116/+4 Software** ★★★★★
Favourite 4 (4 Programme) 24,90
Plus Paket (4 Programme) 30,-
Superhits (6 Programme) 24,90
Classics I (4 Programme) 27,90
Classics II (4 Programme) 27,90
King Size (50 Programme) 24,90
International Karate 19,90
Comp. Service, PF 1304, 7913 Senden. Aktuellen Katalog anfordern!!!!

★★★ **C16/116/+4 Software** ★★
Robo Knight - Street Olympics - Tazz - Varmint - Finders Keepers - Runner - Kickstart - Dark Tower - Vox - Climb It - Shoot It - Shark Munch It - Trizones - Guzzler - Autobahn - Water Grand Prix - Golf. Alle Programme je nur DM 9,90!! 24-Std.-Bestellservice 073 07 / 62 30. Comp. Service, PF 1304, 7913 Senden

★★★ **Elite plus V1.0 ★ Ka/Di** ★★
★ fantastische neue Program- ★
★ miersprache für strukturier- ★
★ tes Programmieren auf Ihrem ★
★ PLUS/4 // C16, C116 (64 KB) ★
★ 29.- DM mit ausf. Handbuch ★
★ incl. Porto & Verp. Info geg. Rp ★
★ bei TEDsoft / Bohlingerstr. 2 ★
★★★★ 7700 Singen ★★★★★

C-64 Super-Spielepaket C-64
Nur auf Disk: Knight Rider, Miami Vice, Boulder Dash 3, Gilligans Gold, James Bond, Snoopy, Artellerie Duel, Protektor 2, Bond Jack, Comando. Das ganze nur 10 DM + 2 DM Porto. Tausche auch. Schreibt an: Christian Doeres, Friedrichstr. 5, 5550 Bernkastel-Kues!

FRANZÖSISCH auf dem C16/116
Vocteach Plus mit den franzö. Accents u. dt. Sonderzeichen. Ein Dateiensystem garantiert optim. Speicherplatzausnutzung. Mit Menüsteuerung, Auswertung usw. Auch ohne frz. Erw. u. in Sets mit Grafikprogrammen u. Grafikdemos zum Superpreis erhältlich. Info: B. Jakob, Kehl 4, 8832 Weibenburg

GRAFIKPROGRAMME - GRAFIK-DEMOS für C 64 mit Simons' Basic und C 16/116! Grafiken versch. Art und Technik! Initialen in Grafikgroßdarstellung. Grafikdemos mit detaillierten Erklärungen und vielen Beispielen. 64er auch auf Disk! Info gegen Rückporto! Info mit Cassette: 3,80 DM! B. Jakob, Kehl 4, 8832 Weibenburg

★ **CPC Software zu Toppreisen!** ★
★ tel. best. bei M+K 07022 / ★
★ 6815. Dan Dare C/D 30 90 / ★
★ 40 - 90 DM; Johnny Reb II ★
★ 35.90/46 - 90 DM; Jack the ★
★ Nipper 32.90/42 - 90 DM; ★
★ Werner 34.90/43 - 90 DM; ★
★ Knight Games 31.90/42 - 90 ★
★ DM; Split Personal 31.90/42- ★
★ 90 DM; Dragon's Lair 32.90 / ★
★ 42 - 90 DM; Uchi Mata 32.90 / ★
★ 42 - 90 DM ★

C-116, C-116 + PLUS 4 Spitzenprogramme Info = 80 Pfennig Rückporto. Anfordern bei Michael Hasen, 2800 Bremen 21, Ritterhuderstr. 55

★★ **Top-Spiele für C-16** ★★ Spielpaket: Mover-Poker 2-Tankwars-Diam SDI - usw. insgesamt 10 Programme bei Vorkasse 10 DM, bei Nachnahme 13,60 DM. Außerdem noch ein Anwenderpaket mit 8 Programmen zum gleichen Preis. Jürgen Arns, Jügendheimstr. 1, 5550 Obernkastel-Kues

Verkaufe „Compute mit“ CPC 464 mit ORIGINAL CASSETTEN von 6/85 bis 120+6 für CPC 464 wegen Systemwechsel. Preis VB. Tel.: 07066 / 6111 ab 16.00 Uhr

Top-Games für den C-16 auf einer Kasette zu nur 20 DM. Z.B. Pac-Mac, Sos usw. Infos gegen Rückporto bei: A. Röhlinger, Julius-Brecht-Str. 2, 4990 Lübbecke 1

Schneider CPC!!! Tausche Software auf Disc/Tape. Listen an: Thorsten Olbrich, Eichenau Nr. 36, 6402 Großenluder 1, Tel. 06648 / 2667. Antworten sofort!

MSX - Verkäufe Supersoftware zu billigen Preisen. Biete über 120 Programme an. Liste gratis anfordern!!! Ralph Thuli, Ringstraße, CH-9477 Trübbach. Tel. 085 / 51519 (18.00-19.00)

C16/116-Spiel- u. Anwenderprogr. z.B. Flugsimulator, Haushaltsbuch, Kniffel usw. Info gegen Rückporto Ralf Steinhauer, Damaschkestr. 563, Remscheid 1

Commodore C-16 116** plus 4**
Hardcopy - Hardcopy - Hardcopy druckt Ihre Bildschirmgrafik in max. 60 Sek. auf ca. DIN A4 Größe aus. Für 16K und 64K Version geeignet. Als Kasette oder Diskette für nur 19 DM + Porto per NN bei: Frank Schmitz, Kurzer Weg 1, 5206 Neunkirchen 2, Tel.: 02247 / 8111

C64 Programme billig abzugeben ca. 1500 Stück. Liste anfordern. Bin auch an Tausch interessiert, aber bitte nur gute Programme. Jürgen Klein, Kürtnnerstr. 46, 5060 Bergisch Gladbach 2

Biete Hardware

C16/116/+4 Hardware + Zubehör
Joystick/Adapterkabel 8,90
Joystick-Verlängerung 3M 10,25
Daten-Cassetten C15 5 Stk 7,80
Disketten 5 1/4" 10 Stk 12,80
Kunststoff-Pinzette 125 mm 0,80
Feinlötkolben 220V/25W 10,20
PC-HC-ELEKTRONIK-TECHNIK
Fitz-Reuter-Str. 3, 2914 Barßel
Computer / Zubehör
EDV-Endlospostk. 250 St 22,50
Endlospapier 1000 Bl. 28,50
Blano-Etiketten 4000 Stk 41,50
Kopiehalter Format A4 47,20
Druckerstände Acrylglas 99,50
Disketten Pen Set 3 Stk 8,20
Fordern Sie bitte unseren Hauptkatalog an gegen 80 Pf.
PC-HC-ELEKTRONIK-TECHNIK
Fritz-Reuter-Str. 3, 2914 Barßel

Verk. 15 VC-20 Module zum Preis pro Modul: 50,- DM Spitzenspiele! R. Clobes, Hohenkirchner Str. 10, 3501 Espenau

Farbbänder Seikosha 08761/4245
 GP50-100-500-550 nur **DM 19,50**
 GP700 schw./vierf. **DM 28,-/44,-**
 SP800/1000 **DM 22,-** MS15 **DM 9,90**
Restposten Drucker m. Garantie
 Farb-Dr. GP700 Centr. **DM 529,-**
 Drucker GP500A Centr. **DM 299,-**
 Drucker GP100AS (RS232) **199,-**

★★ **Plus 4 (64 KB) + Datensette** ★★
 für nur DM 195 (Fr. 195) zu verkaufen.
 Dazu gratis Software. J. Kottmann,
 Alpenquai 42, CH-6005 Luzern. Tel.
 00 41 41 / 44 33 43

Commodore C-16/116 C16/116
 Speichererweiterung 64 KByte Bau-
 satz mit allen benötigten Teilen *
 Schalter für 16 KByte - 64 KByte um-
 zuschalten und ausführlicher
 Bauanleitung für nur 49,50 DM (incl.
 MwSt.) per Nachnahme. Frank
 Schmitz, Kurzer Weg 1, 5206 Neun-
 kirchen 2, Tel.: 0 22 47 / 81 11

Verkaufe Atari 13 OXE + Floppy +
Software + Bücher + Zeitschriften. 8
 Monate alt. NP 1750.- DM für 1200.-
 DM Computerkurs komplett (84
 Hefte in Ordern) für 300.- DM. Tel.
 ab 18 Uhr 0 22 43 / 47 22, Uwe Zim-
 mermann, Probacher Str. 52, 5208
 Eitorf.

★★★★★★★★★★★★★★★★★★★★
 ★ **Verkaufe:** TI 99/4A + Exbasic ★
 ★ + Handbücher + Joyst.-Adap- ★
 ★ ter. + Software (Cassetten) ★
 ★ VB: 300,-. Tel. 05 91 / 4 91 27 ★
 ★★★★★★★★★★★★★★★★★★★★★★

ZU VERKAUFEN !!!!
 C-16 + Datensette + Joystick + 18 Top-
 spiele z.B. Grandmaster + Basickurs
 + Turbo Tape für nur 300 DM. Bitte
 melden bei Tel: 077 20 / 13 67

★★★★★ **C-16/116** ★★★★★★
 64 KByte Speichereinb. Service nur
 59.- DM. User Port Modul 45.- DM
 RS-232-Schnittstelle 42.- DM. Kas-
 settenkopiermodul auch für norma-
 le Recorder ab 55.- DM. Info von Fa.
 Willi Keßler - Elektronik - Ru-
 prechtstr. 14, 6736 Edesheim, Tel.
 0 63 23 / 71 14 (nach 17 h)

SUPERGELEGENHEIT!!!!
 Verkaufe VC 1520 für nur 200 DM
 und VC 200 Für 60 DM. Der Comput-
 er für Kinder und Anfänger. Tel.:
 077 20 / 13 67

VERKAUFE AMIGA (neu + 6 Monate
 Gara.) mit Monitor für: 3000.- + Er-
 weiter. Ohne Monitor mit Antennen-
 kabel für: 2300.- + Erweiter. Alle mit
 Deutsch Tastatur. Tel: 0 52 50 / 2 83!!

COMMODORE HARDWARE UND
ZUBEHÖR
 Farbmonitor Philips 8500 - 549,-;
 Drucker MPS 801 - 349,-; Thermo-
 drucker TXP 1000 - 398,-; Quick-
 shot II - 15,90; Competition Pro
 (Microsch.) - 39,-; Joystickadapter
 C16/116/+4 - 9,90; Abdeckhaube
 C16 Hartplastik - 12,-. Preise Stand
 10/86 zzgl. Versand!
 Comp. Service, PF 1304, 7913 Sen-
 den

Rennersoft Computer-Distribution

ACTIONS-GAMES-ADVENTURES-BUISNESS-SOFTWARE

Wir haben für Ihren Commodore C 16/116 und Ihren Commodore C 64

Projekt Nova, Quiwi, They sold a Million 2, Nexus, Mission Elevator, Dragon's Lear, Parallax, Miami Vice, Grandmaster, Grafik-Designer
Gratis-Liste anfordern. **Achtung:** Computersystem angeben!

Renner-Soft Abteilung A 1 · Postf. 920 · 4440 Rheine

C-16 Großer Bruder des C-64
 Datenfernübertragung wird zum
 Kinderspiel! Erweiterungsplatine
 >USMO I< für C16/116 mit **RS-232**
 und zwei Userports für **DM 138,-**.
Modem steckfertig mit passender
 Software ■ Wahl von der Computer-
 Tastatur ■ Autoanswer ■ Mailbox-
 betr. **DM 148,-** ■ Speichererw. **DM**
49,- ■ Guss Data Connection
0 27 23 / 66 77

DOPING für den 64er: SUPERBE-
 TRIEBSSYSTEME als Modul f.d.
 Userport: Speeder, Monitor, Freezer,
 Tools, Hardcopies, Gamekiller,
 Centr.-Interface
 Final Cartridge (neu!) 125 DM
 Power Cartridge (neu!) 135 DM
 (NN + 3,50 o. Vork.) - Bestellen: 24 h
 Tel.: 05 61 / 88 01 11 oder ASTRO,
 Postf. 1330, 3502 Vellmar

Verschiedenes

Tausche / Verkäufe Spiele für C64
 (Disk) Habe Paperboy, World Games
 usw. Listen an Oliver Nazareth,
 Feldstr. 37, 2304 Laboe, Tel. 0 43 43 /
 78 47. Antwort 100%. Suche Speed-
 osx

ASTRO-VERSAND sucht noch:
 Für den gesamten C64/128+ Zub-
 Bereich: Pfiffige Produkte, Spitzen-
 Hard- u. Software zum Exklusiv-Ver-
 trieb. Angebote, Beschreibungen,
 Mustersendungen von FREAKS,
 PROFIS, IMPORTEUREN u. HER-
 STELLERN allzeit erwünscht. Bitte
 schriftlich an:
 ASTRO, Postf. 1330, 3502 Vellmar

Wir haben die Software, die Sie suchen!

Commodore 64/128

	Kassette	Diskette
Alleykat	29,95 DM	38,95 DM
Asterix	39,95 DM	44,95 DM
Dan Dare	38,95 DM	
Deactivators	38,95 DM	48,95 DM
Die Erbschaft	48,95 DM	58,95 DM
Hacker II	38,95 DM	44,95 DM
Jack the Nipper	32,95 DM	42,95 DM
Knight Games	38,95 DM	38,95 DM
Leather Goddesses of Phobos		78,95 DM
1942	32,95 DM	44,95 DM
Parallax	34,95 DM	
Room 10	29,95 DM	
Sanxion	32,95 DM	
Tass Times		58,95 DM
Trinity		98,95 DM
Vera Cruz	48,95 DM	58,95 DM
Way of the Expl. Fist II	32,95 DM	

Schneider CPC (464, 664, 6128)

	Kassette	Diskette
Batman	29,00 DM	44,95 DM
Jack, the Nipper	32,95 DM	32,95 DM
Movie	25,95 DM	44,95 DM
Room 10	29,95 DM	44,95 DM
Saboteur	32,95 DM	32,95 DM
Spindizzy	25,00 DM	
Star Strike II	32,95 DM	
Tau Ceti		32,95 DM
Winter Games	32,95 DM	54,95 DM

Commodore 16/116/PLUS4

	Kassette
ACE	32,00 DM
ACE (+64K)	34,95 DM
Bandits at Zero	14,95 DM
Bomb Jack	29,00 DM
Formula One Sim.	9,95 DM
Frank Bruno's Boxing	25,00 DM
Ghosts 'n' Goblins	29,00 DM
International Karate	25,00 DM
Jack Attack (Modul)	14,95 DM
Kikstart	9,95 DM
Kiwi (+64K)	29,00 DM
Legionnaire	19,95 DM
Matrix/Laser Zone	29,00 DM
Monty on the Run	25,00 DM
Paint Box	25,00 DM
Project Nora	25,00 DM
ROM-Listing	29,00 DM
Space Pilot	19,95 DM
Text Manager (Disk)	25,00 DM
Text Manager (Kass.)	25,00 DM
Winter Olympiade	29,00 DM
Yie Ar Kung-Fu	29,00 DM

ATARI ST (260, 520, 1040)

	Diskette
Arena	89,00 DM
Deep Space	89,00 DM
Leader Board	75,00 DM
Winter Games	72,00 DM

Fordern Sie kostenloses Informationsmaterial über unser Liefer-Programm für folgende Computer an: COMMODORE 64/128, COMMODORE 16/116/PLUS 4, SCHNEIDER 464/664, ATARI 130XE/260ST/520ST/800XL, SINCLAIR SPECTRUM und MSX. Bitte geben Sie immer Ihren Computertyp an!



So bestellen Sie:

Der Mindestbestellwert von Software liegt bei **15.00 DM!**

Sie können das Geld in bar oder per Scheck Ihrer schriftlichen Bestellung beilegen.

Bei Nachnahme-Versand werden 4.00 DM für Porto-Kosten erhoben.

Auslandsbestellungen werden nur per Vorauskasse entgegengenommen!

Rufen Sie uns doch einfach einmal an! Unsere Telefone sind wochentags ab 9.30 Uhr für Sie zu erreichen!

SOFTWARE-VERSAND

ANDREAS BACHLER
 POSTFACH 429
 D-4290 BOCHOLT

☎ (0 28 71) 18 30 88

Bewerbung als Programmautor

Name: _____ Vorname: _____ Alter: _____

Straße: _____ Wohnort: _____

Titel des Programmes: _____ Computersystem: _____

Erforderlicher Speicherplatz: _____ KBytes

Erforderliche Peripherie: _____

Ich versichere hiermit, daß ich der Autor des oben genannten Programmes bin und alle Rechte besitze. Ich bin damit einverstanden, daß Sie mein Programm in einer Zeitschrift abdrucken und/oder dieses verkaufen. Das einmalige Honorar beträgt 120 DM pro abgedruckter Seite. Sollte das Programm nicht veröffentlicht werden, bitte ich um Rücksendung meiner Unterlagen. Sollte mein Programm in der Zwischenzeit von anderer Seite erworben worden sein, werde ich Sie unverzüglich benachrichtigen. Ich bestätige durch meine Unterschrift, daß ich Kenntnis davon habe, daß der Tronic-Verlag gegen Personen, die Programme einschicken, deren Urheberrechte sie nicht besitzen, rechtliche Schritte einleitet.

Unterschrift des Programmautors (bei Minderjährigen auch die eines Erziehungsberechtigten): _____

Folgende Unterlagen liegen diesem Schreiben bei:

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

Unterschrift des Programmautors (bei Minderjährigen auch die eines Erziehungsberechtigten): _____

(Bitte zweimal unterschreiben!)

Einsenden an: **Tronic-Verlag Abt. Programmentwicklung, Am Stad, 3440 Eschwege**



Auftrag für Gelegenheitsanzeigen in „Compute mit ...“

**An „Compute mit ...“
Tronic-Verlag
Postfach
3440 Eschwege**

Unter der Rubrik „Kleinanzeigen“ - veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

Unser Preis für „Kleinanzeigen“:
Private Kleinanzeigen
bis 5 Zeilen nur DM 5,- DM
bis 10 Zeilen nur DM 10,- DM

Jede gewerbliche Gelegenheitsanzeige
bis 10 Zeilen nur 15,- DM!
(dürfen nicht unter Chiffre erscheinen)

Eine Veröffentlichung erfolgt **nur** gegen Vorkasse!

Kleinanzeigen jetzt noch preiswerter

Name und Adresse Oliver Müller, Nelkenstr. 11, 8300 Landshut

Oliver Müller

Unterschrift

Datum

Ich wünsche folgenden Text zu veröffentlichen:

**An alle VC-20 Fans!!!
Verkaufe Actionspiele im BASIC
und Maschinensprache! Viel Spaß-
2 DM. Ist das ein Angebot? Einmal
liste mit 80 Pf Rückp. anfordern! 5,-**

Zutreffendes ankreuzen

- 10,-
- Biete Hardware biete Software suche Kontakte Versch.

C-16: Action + Spannung frei Haus

Dies sind Superspiele aus unserem Programm von über 150 C-16 Spielen:

Robo Knight Cass. 9,95

Ein Geschicklichkeitsspiel der Superklasse. In einem rätselhaften Schloß sind 15 Schilde verteilt. Über Brücken und Treppen müssen Sie diese Schilde einsammeln. Lassen Sie sich nicht durch dunkle Gänge und mit Fackeln beleuchtete Räume oder dem Verlies beirren.

European Games Cass. 19,95

Hier kommen neue originelle Sportdisziplinen auf Ihren Monitor: Schwimmen, Hammerwerfen, Weitsprung, Rudern und Gewichtheben. Eine Topgrafik sorgt für ein ungetrübtes Sportereignis.

Frank Brunos Boxing Cass. 29,95

3 Boxer der Spitzenklasse stehen Ihnen als Gegner zur Verfügung. Mit ausgeklügelter Bewegungstaktik und Ihrem Siegeswillen sollte es aber zu einem glatten K.O. reichen. Die Spitzengrafik tut das ihrige, um dieses Spiel in unsere Top Ten einzureihen!

Operation Thunderstorm Cass. 29,95, Disk. 29,95

Mindestens 16 K Erweiterung notwendig!!!

Sie sind Jim Alton, ein bekannter und gewitzter Privatdetektiv. Das Management eines großen Warenhauskonzerns bittet Sie um schnellste Hilfe: Eine skrupellose Gangsterbande hat im bekannten City-Kaufhaus eine Reihe von Bomben gelegt und verlangt ein hohes Lösegeld.

Nur Sie allein können den Millionenverlust verhindern, indem Sie alle Bomben finden und unschädlich machen.

Dies ist der Einstieg in ein neues, deutschsprachiges Textadventure mit über 30 Räumen! Ideenreichtum und logisches Denken führen Sie durch ein Labyrinth von Räumen, zu verschlossenen Türen und geheimnisvollen Kisten. Und der Fahrstuhl scheint ein seltsames Eigenleben zu haben.

Kombi-Angebot K-4 Cass. 99,00, Disk. 99,00

Ein Super-Angebot: 64 K Erweiterung und »Operation Thunderstorm«. Der preisgünstige Einstieg in die neue Leistungsklasse des C-16.

Project Nova Cass. 24,95

Dieses phantastische Weltraum-Strategie-Action-Superspiel zählt zu den Spitzenspielen für den C-16. In der Galaxis ist der Teufel los. Und Sie sind mitten drin. Aber durch geschicktes Navigieren (Die Galaxis ist auf einer Karte dargestellt) und Ausnutzen der 10 Geschwindigkeitsstufen sollten Sie heil durchkommen. Notfalls hilft Ihnen Ihr Schutzschirmgenerator. Und wenn die Energie knapp wird, fliegen Sie Ihren Tankplaneten an. Vielleicht schaffen Sie es vom Anfänger zum Superhelden. Das Spiel hat 8 Schwierigkeitsstufen.

Yie Ar Kung Fu Cass. 33,95

Um Grandmaster zu werden, die höchste Kung Fu Auszeichnung, müssen Sie 8 Gegner besiegen. Da diese raffinierteste Angriffstechniken wie z. B. tödliche Kampfsterne werfend oder aber waagrecht auf Sie zu fliegend verwenden, müssen Sie die Ihnen zur Verfügung stehenden 6 Abwehrbewegungen perfekt beherrschen. Nur vollendet ausgeführte Kombinationen genügen, um wie Ihr Vater, die Grandmasterwürde zu erlangen.

Unsere Lieferbedingungen

Sie bestellen – wir liefern

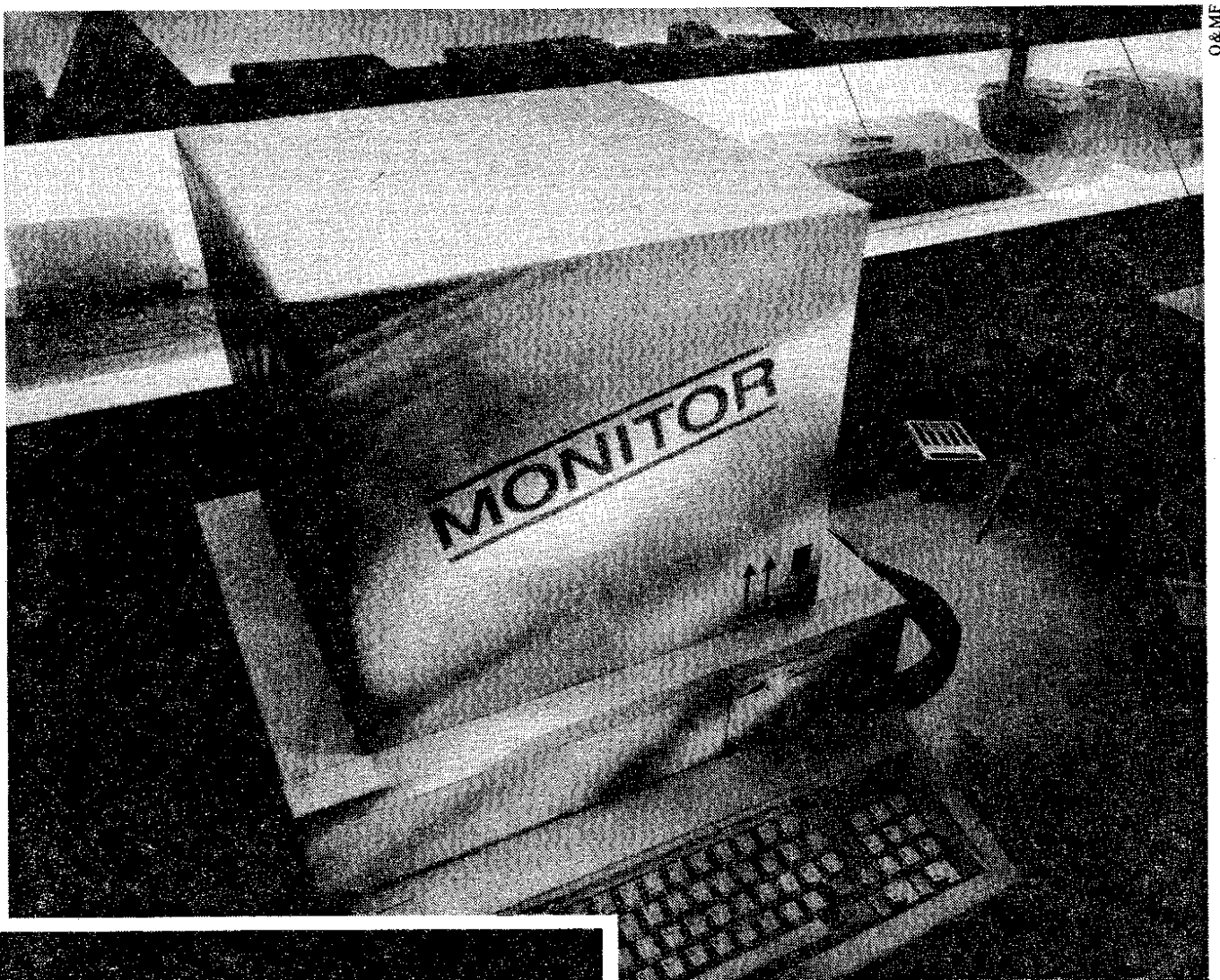
und zwar ohne zusätzliche Portokosten, ohne Nachnahmegebühr, ohne umständliches Hinzuaddieren irgendwelcher Kosten, ohne Mindestbestellmenge. Denn unsere Preise sind Endpreise. Sie wissen also sofort, wieviel ein Spiel kostet. Und haben keinen Ärger, daß ein Programm mehr kostet als Sie dachten. Unser großes C-16-Lager ermöglicht uns eine schnelle Lieferung und somit zufriedene Kunden.

Fordert unser mehrseitiges, kostenloses Info an!!!

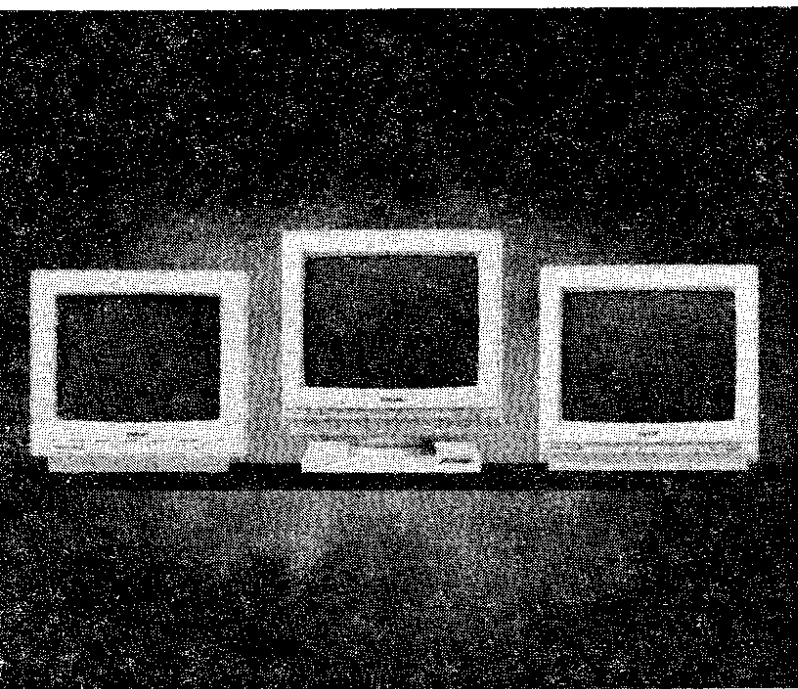
Versandhandel R. Lindenschmidt · Schulstr. 14 · Postfach 1328 · 4972 Löhne 2 · Tel. (05732) 72849

Das nächste Heft erscheint am
15.12.1986

Das C-16/116-Sonderheft
von »Compute mit« erscheint am
15.12.1986



O&MF



Wunderbar, wenn ein echter Philips Monitor drin wäre...

Wenn Ihnen jemand ein supergünstiges Paket-Angebot macht, fragt es sich, ob die Qualität nicht auf der Strecke geblieben ist. Beim Computer ist dies weniger zu befürchten. Ins Auge geht es jedoch, wenn am Monitor gespart wurde. Am besten, Sie entscheiden sich deshalb gleich für einen echten Philips Monitor: Der bringt Ihnen durch die hohe Auflösung eine erhebliche Verbesserung der Wiedergabe und der Lesbarkeit. Der schont die Augen, statt sie zu ermüden - auch bei längerer Nutzung. In Farbe oder Monochrom (Bernstein, Grün oder Papierweiß) bieten Ihnen die Philips Monitore Kompatibilität mit fast allen Systemen und optimale Zukunftssicherheit. Klar und deutlich zeigen sie, daß sie mindestens so gut sind wie Ihr Personal-Computer. Und das sollte Ihnen etwas wert sein.

Mehr darüber von Philips GmbH, Geschäftsbereich Neue Medien, Postfach 101420, 2000 Hamburg 1.

 Philips paßt
in unsere Zukunft



PHILIPS