

monatlich

# Compute mit

## COMMODORE & SCHNEIDER

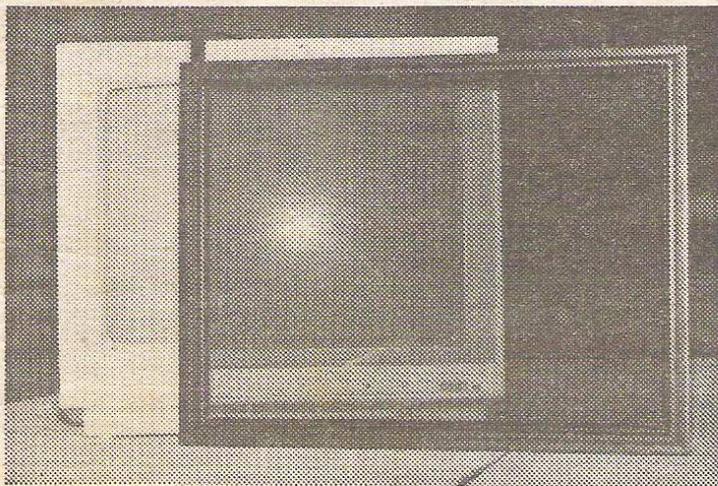
1/86

VC-64, VC-20, C-16, CPC-464  
Unabhängiges Magazin für Anwender von Commodore- und Schneider-Computern

2,80 DM  
24 öS  
2,80 sFr

**Aktuell +++ Aktuell +++ Aktuell +++ Aktuell**

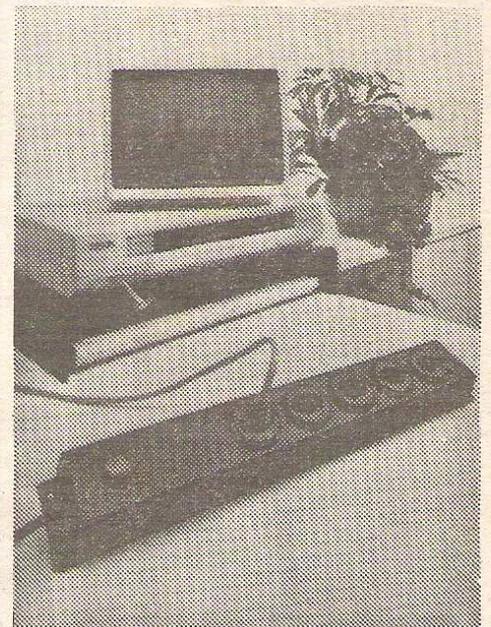
### **NEU:** Polarisationsfilter für Bildschirme und Monitore



Jetzt können Sie überall im Fachhandel die neuen Polarisationsfilter der John Hall Computerdivision Hamburg erwerben. Die Filter gibt es in verschiedenen Größen, so daß jeder Bildschirm oder Monitor auf einfache Weise mit ihm ausgerüstet werden kann.

Weiter S. 29

der solche Schwankungen oder kurzfristige Spannungsanstiege durch statische Entladung ausgleicht. Ebenfalls werden durch eine Hochfrequenzdrossel Störungen abgebaut, die via Erdleiter übertragen werden. Die Stromschiene kann überdies in den SICOS-Computerständer integriert werden, sodaß nur noch ein einziges Kabel zum Arbeitsplatz führt, und die Verteilung an die Peripheriegeräte direkt von dort aus erfolgt, was dem „Kabelsalat“ unter dem Pult ein Ende bereitet.



### **Schwankungsfreie Stromversorgung**

Schwankungen oder Störungen im Stromnetz sind oft Ursache von Computerfehlern wie Datenverlust oder – im schlimmsten Fall – Beschädigungen an

den Geräten. Die Deutsche Generalvertretung für die SICOS Computerzubehörline führt neuerdings im Sortiment eine Stromschiene mit integriertem Netzfilter,

**Report – Kleinanzeigen – Serie – Werkstatt – Tips & Tricks – Bücher  
Software-Reviews – Leser + Meckerecke – Assembler-Kurs – Software-Abo  
Astronaut-Battle – Hueycobra – Locos – Rock'me – Defender – u.v.m.**

## Report

Was gibts Neues auf dem Software-Markt?

Software-Test

Polarisationsfilter für Bildschirme u. Monitore

Die „DO IT YOURSELF“ Mailbox

Die neuen Hitrans-Modems von CDI

Btx-Commodore 64 Software per Telefon

## Bücher

Pascal

Forth

Das Computerjahrbuch '86

## Hardware

5.25" Floppy-Stationen für Schneider 464

RAM-Karten für Schneider 464

## Serie

Tastatur-Verwaltung des Commodore 64

## Software-Service

### Assembler-Kurs

Teil 8

### Bewerbung als Programmator

### Kleinanzeigen

### Leser + Meckerecke

# Software

17

## Commodore

18

Computer-Kommunikation (C-64)

29

Checksummer C Version 1.0 (C-64)

29

Astronaut Battle (C-64)

29

Hueycobra (C-16)

30

Hardcopyroutine (C-16)

Helmut und die Zeitmaschine (C-16)

Locos (VC-20)

Rock'me (VC-20)

Charly der Raketenmann (VC-20)

62

## Schneider

62

Checksummer 1.0 CPC

63

Defender

Boulder Dash

## Tips & Tricks

Commodore 64

61

VC-20

Schneider

63

## Werkstatt

Window-Technik mit dem C-64

Interrupt-Programmierung mit dem VC-20

3

Basic-Befehle auf den Funktionstasten des CPC 464

31

„Compute mit“

liegt ab

20. Januar

wieder bei Ihrem

Zeitschriftenhändler

für Sie bereit.

41

53

54

56

## Impressum

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

### Redaktion:

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

### Freie Mitarbeiter:

Volker Becker,

### Gesamtherstellung:

Druckhaus Dierichs Kassel, Frankfurter Str. 168, 3500 Kassel

### Vertrieb:

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

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

### Anzeigenverwaltung:

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

### Anzeigenteilung:

Inland: H. Wendt  
Ausland: M. Kleimann

### Anzeigenpreise:

Bitte Mediaunterlagen anfordern.

### 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 oder Bezeichnungen frei von Schutzrechten sind.

### Bezugspreis:

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

### Autoren, Manuskripte:

Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen.

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

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

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

Kopierfähige Kassette oder Diskette mit dem Programm (Computer-Bezeichnung), von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinenlistings), evtl. Bildschirmfotos von einem Probelauf und ausführliche Programmbeschreibung (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaues, Farbe, Grafik usw.). Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.

# Tastatur-Verwaltung des C-64

## Fortsetzung unserer 2-teiligen Serie über die Tastaturverwaltung des Commodore 64.

Um über die Tastatur ein Steuerzeichen drucken zu können, braucht man eben einige Mühe. In Tabelle 2 habe ich zusammengefaßt, wie man das erreichen kann.

Ein Zeichen mit einem ASCII-Wert von 0 bis 30 kann man ausnahmslos mit Hilfe der CTRL-Taste zusammen mit der passenden Normal-Taste drucken. Es scheint, daß die CTRL-Taste den ASCII-Wert um 64 herunterbringt (d. h. 6. Bit löscht). Gesamte Zeichenfarbensteuerung über CTRL-Taste zusammen mit Zifferntasten ist aber so willkürlich aufgebaut, daß man keine Ge-

setzmäßigkeit mehr findet. Man ist gezwungen, blind der Anleitung zu folgen. Obwohl in Tabelle 2 in Spalte „Taste“ darauf hingewiesen wird, welches Steuerzeichen mit welcher Taste (bzw. Tastenkombination) hinter einem Ausführungszeichen geschrieben werden kann, muß man an dieser Stelle den Anwender darauf aufmerksam machen, daß vier von diesen nicht erreichbar sind. Dies sind 'Return' = CHR\$(13), 'Shift+Return'=CHR\$(141), 'Del'=CHR\$(20) und 'Shift+Stop'=CHR\$(131). In solchen Fällen ist zu schreiben:

TABELLE - 2

ASC	ZCHN	BEMERKUNG	TASTE	ASC	ZCHN	PEEK(203)
0	" "		CTRL+@	64	"@"	46
1	"a"		CTRL+a	65	"a"	10
2	"b"		CTRL+b	66	"b"	28
3	"c"	?	CTRL+c / Stop	67	"c"	20, 63
4	"d"		CTRL+d	68	"d"	18
5	"e"	Weiss	CTRL+e / CTRL+2	69	"e"	14, 59
6	"f"		CTRL+f / CTRL++	70	"f"	21, 57
7	"g"		CTRL+g	71	"g"	26
8	"h"	Blck Sh+C	CTRL+h	72	"h"	29
9	"i"	Entr Sh+C	CTRL+i	73	"i"	33
10	"j"	Line Feed	CTRL+j	74	"j"	34
11	"k"		CTRL+k	75	"k"	37
12	"l"		CTRL+l	76	"l"	42
13	"m"	Return	CTRL+m / Return-?	77	"m"	1, 36
14	"n"	Gr/Klein	CTRL+n	78	"n"	39
15	"o"		CTRL+o	79	"o"	38
16	"p"		CTRL+p	80	"p"	41
17	"q"	Cur-unter	CTRL+q / CRSR-U	81	"q"	7, 62
18	"r"	RVS-on	CTRL+r / CTRL+9	82	"r"	17, 32
19	"s"	Home	CTRL+s / HOME	83	"s"	13, 51
20	"t"	Del	CTRL+t / Del-?	84	"t"	0, 22
21	"u"		CTRL+u	85	"u"	30
22	"v"		CTRL+v	86	"v"	31
23	"w"		CTRL+w	87	"w"	9
24	"x"		CTRL+x	88	"x"	23
25	"y"		CTRL+y	89	"y"	25
26	"z"		CTRL+z	90	"z"	12
27	"["		CTRL+[	91	"["	45
28	"\	Rot	CTRL+\ / CTRL+3	92	"\"	8, 48
29	"]	Cur-recht	CTRL+] / CRSR-R	93	"]"	2, 50
30	"^	Grn	CTRL+^ / CTRL+6	94	"^"	19, 54
31	"_	Blau	CTRL+_ / CTRL+7	95	"_"	24, 53
128	"`"			192	"`"	
129	"A"	Orange	COM+ 1	193	"A"	56
130	"B"			194	"B"	
131	"C"	?	Shift+ / COM+Stop-?	195	"C"	63
132	"D"			196	"D"	
133	"E"		Fn-Taste 1	197	"E"	4
134	"F"		Fn-Taste 3	198	"F"	5
135	"G"		Fn-Taste 5	199	"G"	6
136	"H"		Fn-Taste 7	200	"H"	3
137	"I"		Fn-Taste 2	201	"I"	4
138	"J"		Fn-Taste 4	202	"J"	5
139	"K"		Fn-Taste 6	203	"K"	6

### PRINTCHR\$(13)

oder entsprechende ASCII-Werte. Ausführung von 'Stop' bzw. 'Shift+Stop' mit PRINTCHR\$(3) bzw. 131 gelingt nicht. Um das Programm zu unterbrechen muß man direkt die 'Stop'-Taste drücken oder im Programm an geeigneter Stelle den BASIC-Befehl 'STOP' zufügen.

In der letzten Spalte der Tabelle 2 stehen die Codenummern der zu drückenden Tasten. Diese Nummern werden im Speicherplatz 203 registriert, wenn man die entsprechende Taste drückt. Nach dieser Nummer sucht das Betriebssystem aus der Tasten-Code-Matrix den ASCII heraus und druckt das entsprechende Zeichen aus. Wie Sie aus der Tabelle entnehmen können, gibt es für manche Zeichen keine Tastennummer. Diese Zeichen können also auch nicht mit einer Taste (bzw. Tastenkombination) direkt abgedruckt werden. Nur im Umweg kann man solch ein Zeichen auf Bildschirm (oder Papier) bringen. Ein Beispiel wäre inverses groß 'N', das für die Umschaltung in den Groß/Graphik Modus benötigt wird (Ersatz von POKE 53272,21 oder Commodore-Shift Tasten). Dieses Zeichen am Anfang eines Programms wird nicht selten gebraucht.

140	"█"		Fn-Taste 8	204	"L"	3
141	"█"	Sh+Return	Shift+Return-?	205	"M"	1
142	"█"	Gr/Grafic		206	"N"	
143	"█"			207	"O"	
144	"█"	Schwarz	CTRL+I	208	"P"	56
145	"█"	Cur-hoch	CRSR-H	209	"Q"	7
146	"█"	RVS-off	CTRL+0	210	"R"	35
147	"█"	Clr/Home	CLR	211	"S"	51
148	"█"	Insert	INST	212	"T"	0
149	"█"	Braun	COM+ 2	213	"U"	59
150	"█"	hell Rot	COM+ 3	214	"V"	8
151	"█"	Grau-1	COM+ 4	215	"W"	11
152	"█"	Grau-2	COM+ 5	216	"X"	16
153	"█"	hell Grn	COM+ 6	217	"Y"	19
154	"█"	hell Blau	COM+ 7	218	"Z"	24
155	"█"	Grau-3	COM+ 8	219	"+"	27
156	"█"	Violett	CTRL+5	220	"*"	16
157	"█"	Cur-links	CRSR-L	221	"!"	2
158	"█"	Gelb	CTRL+8	222	"@"	27
159	"█"	Cyan	CTRL+4	223	"#"	11

Besonders die Programme, die den graphischen Zeichensatz für das Programm benötigen (z. B. Spielprogramme) benutzen es sehr häufig. Es sind mehrere Arbeitsgänge notwendig, um solch ein Zeichen ins Programm hineinzuschreiben:

Taste 'RVS on' / Shift + 'n'-Tasten / Taste 'RVS off' / Taste "" / Return dann Cursor wieder vor das inverse 'N' bringen und Taste "" und Return

Graphische Darstellung der Arbeitsgänge:

ZeilenNr/PRINT/leer/RVS on/N/RVS off/" / Return

XX  
Cursor hier/" / Return

## TABELLE - 3

TST	ASC	ZCH	UNGESHIFT	ASC	ZCH	GESHIFTET	ASC	ZCH	COMMODORE	ASC	ZCH	CTRL
0	20	"█"	Del-?	148	"█"	Insert	148	"█"	Insert	255	"█"	?? N.N
1	13	"█"	Return-?	141	"█"	Sh+Return?	141	"█"	Sh+Return?	255	"█"	?? N.N
2	29	"█"	Cur-recht	157	"█"	Cur-links	157	"█"	Cur-links	255	"█"	?? N.N
3	136	"█"	Fn-Taste-7	140	"█"	Fn-Taste-8	140	"█"	Fn-Taste-8	255	"█"	?? N.N
4	133	"█"	Fn-Taste-1	137	"█"	Fn-Taste-2	137	"█"	Fn-Taste-2	255	"█"	?? N.N
5	134	"█"	Fn-Taste-3	138	"█"	Fn-Taste-4	138	"█"	Fn-Taste-4	255	"█"	?? N.N
6	135	"█"	Fn-Taste-5	139	"█"	Fn-Taste-6	139	"█"	Fn-Taste-6	255	"█"	?? N.N
7	17	"█"	Cur-unter	145	"█"	Cur-hoch	145	"█"	Cur-hoch	255	"█"	?? N.N
8	51	"3"		35	"#"		150	"█"	hell Rot	28	"█"	Rot
9	87	"w"		215	"W"		178	"!"		23	"█"	
10	65	"a"		193	"A"		176	"r"		1	"█"	?
11	52	"4"		36	"#"		151	"█"	Grau-1	159	"█"	Cyan
12	90	"z"		218	"Z"		173	"L"		26	"█"	
13	83	"s"		211	"S"		174	"!"		19	"█"	Home
14	69	"e"		197	"E"		177	"+"		5	"█"	Weiss
15	1	"█"	?	1	"█"	?	1	"█"	?	255	"█"	?? N.N
16	53	"5"		37	"%"		152	"█"	Grau-2	156	"█"	Violett
17	82	"r"		210	"R"		178	"T"		18	"█"	RVS-on
18	68	"d"		196	"D"		172	"█"		4	"█"	?
19	54	"6"		38	"&"		153	"█"	hell Grn	30	"█"	Grn
20	67	"c"		195	"C"		188	"█"		3	"█"	Stop
21	78	"f"		198	"F"		187	"█"		6	"█"	
22	84	"t"		212	"T"		163	"█"		20	"█"	Del-?
23	88	"x"		216	"X"		189	"█"		24	"█"	
24	55	"7"		39	"'"		154	"█"	hell Blau	31	"█"	Blau
25	89	"y"		217	"Y"		183	"█"		25	"█"	
26	71	"g"		199	"G"		185	"!"		?	"█"	
27	56	"8"		40	"("		155	"█"	Grau-3	158	"█"	Gelb
28	66	"b"		194	"B"		191	"█"		2	"█"	?
29	72	"h"		200	"H"		180	"!"		8	"█"	Blck Sh+C
30	85	"u"		213	"U"		184	"█"		21	"█"	
31	86	"v"		214	"V"		190	"█"		22	"█"	
32	57	"9"		41	")"		41	">"		18	"█"	RVS-on
33	73	"i"		201	"I"		162	"█"		9	"█"	Entr Sh+C
34	74	"j"		202	"J"		181	"!"		10	"█"	Line Feed
35	48	"0"		48	"@"		48	"@"		146	"█"	RVS-off
36	77	"m"		205	"M"		167	"!"		13	"█"	Return-?
37	75	"k"		203	"K"		161	"!"		11	"█"	
38	79	"o"		207	"O"		185	"█"		15	"█"	
39	78	"n"		206	"N"		170	"!"		14	"█"	Gr/Klein

40	43	"+"	219	"+"	166	"⌘"	255	"⌘" ?? N.N
41	80	"p"	208	"P"	175	"_"	16	"⌘"
42	76	"1"	204	"L"	182	"!"	12	"⌘"
43	45	"-"	221	" "	220	"⌘"	255	"⌘" ?? N.N
44	46	","	62	">"	62	">"	255	"⌘" ?? N.N
45	58	":"	91	"["	91	"["	27	"⌘"
46	64	"@"	186	"√"	164	"_"	0	"⌘"
47	44	","	60	"<"	60	"<"	255	"⌘" ?? N.N
48	92	"£"	163	"%"	168	"⌘"	28	"⌘" Rot
49	42	"*"	182	"-"	223	"⌘"	255	"⌘" ?? N.N
50	59	";"	93	" "	93	" "	29	"⌘" Cur-recht
51	19	"⌘" Home	147	"⌘" C1r./Home	147	"⌘" C1r./Home	255	"⌘" ?? N.N
52	1	"⌘" ?	1	"⌘" ?	1	"⌘" ?	255	"⌘" ?? N.N
53	61	"="	61	"="	61	"="	31	"⌘" Blau
54	94	"↑"	222	"⌘"	222	"⌘"	30	"⌘" Grn
55	47	"/"	63	"?"	63	"?"	255	"⌘" ?? N.N
56	49	"1"	33	"!"	129	"⌘" Orange	144	"⌘" Schwarz
57	95	"←"	95	"←"	95	"←"	6	"⌘"
58	4	"⌘" ?	4	"⌘" ?	4	"⌘" ?	255	"⌘" ?? N.N
59	50	"2"	34	""	143	"⌘" Braun	5	"⌘" Weiss
60	32	" " Space	160	" " Sh+Space	160	" " Sh+Space	255	"⌘" ?? N.N
61	2	"⌘" ?	2	"⌘" ?	2	"⌘" ?	255	"⌘" ?? N.N
62	81	"q"	209	"Q"	171	"⌘"	17	"⌘" Cur-unter
63	3	"⌘" Stop	131	"⌘" Load+Run	131	"⌘" Load+Run	255	"⌘" ?? N.N

Die Tastencode-Matrix wird in Tabelle 3 näher betrachtet. Davor müssen wir nochmal Tabelle 1 einsehen. Eben haben wir gesehen, daß alle Steuerzeichen nicht über die Tastatur zu erreichen sind. Verallgemeinert kann man sagen, daß alle ASCII-Werte nicht im Tastencodematrix des Betriebssystems enthalten sind. Auch der Zeichensatz von ASCII 96 bis 127 sowie von 224 bis 225 fehlt hier. Im Direktmodus können Sie es folgenderweise nachprüfen:

X=(ein Wert zwischen 96 und 127 oder zwischen 224 und 255)  
 ?"aS("cH(34)cH(x)cH(34)")"

Wenn Sie z. B. X=97 eingeben, wird nach 'Return':  
 ?aS("A")

auf dem Bildschirm erscheinen. Bringen Sie den Cursor nun in diese Zeile und

drücken Sie nochmal 'Return'. Nun erscheint entgegen Ihrer Erwartung 193 anstatt 97. Die Erklärung dafür ist, daß alle Zeichen, die im Bildschirm erscheinen, nach Bildschirmcode (vgl. S. 133 im Handbuch) ausgegeben werden. Es besteht ein Zusammenhang zwischen Bildschirmcode und ASCII-Wert. Beim Schreiben werden alle ASCII-Werte in Bildschirmcode umgewandelt. Beim Einlesen werden aber zur Eindeutigkeit nur bestimmte ASCII-Werte ausgegeben.

Die Tasten-Code-Matrix ist auch nur mit diesen ASCII-Werten aufgebaut.

Es sind 64 Nummern (von 0 bis 63) für die Tastatur bestimmt. Eine Taste hat, ob Sie ohne oder mit Shift-, Commodore- oder CTRL-Taste drücken, immer die gleiche Nummer. Die ASCII-Werte sind aber unter-

schiedlich. Sie sind in entsprechenden Spalten eingetragen.

Die Codenummern 15,52,58 & 61 (ASCII 1,1,4 & 2) werden nicht im Speicherplatz 203, sondern in 653 additiv eingespeichert. Dieser Speicherplatz ist für Flags für Shift- (Fl.=1), Commodore- (Fl.=2) & CTRL- (Fl.=4) Tasten bestimmt. Wenn gleichzeitig mehrere Tasten gedrückt werden, wird die Summe dieser Werte abgespeichert. Die Betriebsroutine sucht die Code-Matrix-Tabelle und gibt das passende Zeichen heraus, bzw., wie beim Shift+Commodore-Tastenkombination (Fl.=3), leitet die Graphik/Text-Umschaltung ein. Im übrigen wird in der Tabelle 3 eindeutig darauf hingewiesen, welche Taste bzw. Tastenkombination man drücken muß, um den gewünschten ASCII-Wert zu bekommen.

Eine weitere Tabelle gibt die Tastaturordnung genau so wieder, wie sie im C-64 angebracht ist. Zusammengehörige Tastatur-Code-Nummern, Zeichenbild sowie ASCII-Werte sind in der Tabelle übersichtlich dargestellt.

TABELLE - 4

P=Peek(203): T=Taste: Z=Zeichen: A=ASCII-Wert  
 T, Z, A sind ungeshiftet

P	57	56	59	08	11	16	19	24	27	32	35	40	43	48	51	00
T	←	1	2	3	4	5	6	7	8	9	0	+	-	£	HOM	DEL
Z	"+"	"1"	"2"	"3"	"4"	"5"	"6"	"7"	"8"	"9"	"0"	"+"	"-"	"£"	"⌘"	"⌘"
A	95	49	50	51	52	53	54	55	56	57	48	43	45	92	19	20

Zum guten Schluß ein Beispiel zur Anwendung:

Das schönste Programm, das ich am Anfang abgetippt habe, ist das Klavierprogramm aus dem Handbuch. Ärgerlich an diesem Programm ist, daß wenn Sie Ihre Finger schnell über die Tasten bewegen, der Klang nicht nachkommen kann. Das liegt am Programm, das in BASIC geschrieben ist. Noch schlimmer ist, daß es nachklingt, als ob der Computer sich selbstständig gemacht hat. Die Ursache dafür liegt daran, daß die Tastaturbelegung über Zeichenkette geschehen ist. Die Tastaturabfrage geschieht auch nach Zeichen (A\$). C-64 hat einen Puffer für eingegebene Zeichen, der für manche Zwecke sehr nützlich ist, aber hier störend wirkt. Weil man die Tastatur auch nach Code-

```
P 62 09 14 17 22 25 30 33 38 41 46 49 54
T q u e r t y u i o p @ * †
Z "q" "u" "e" "r" "t" "y" "u" "i" "o" "p" "@" "*" "†"
A 81 87 89 82 84 89 85 73 79 80 64 42 94
```

```
P 63 10 13 18 21 26 29 34 37 42 45 50 53 01
T STP a s d f g h j k l ; = RET
Z "STP" "a" "s" "d" "f" "g" "h" "j" "k" "l" ";" "=" "RET"
A 03 65 83 68 70 71 72 74 75 76 58 59 61 13
```

```
P 12 23 20 31 28 39 36 47 44 55 07 02 60
T z x c v b n m , . / CRu CRr SPC
Z "z" "x" "c" "v" "b" "n" "m" ", ." "/" "CRu" "CRr" "SPC"
A 90 88 67 86 66 78 77 44 46 47 17 29 32
```

```
P 04 05 06 03
T Fn1 Fn3 Fn5 Fn7
Z "Fn1" "Fn3" "Fn5" "Fn7"
A 133 134 135 136
```

nummern abfragen kann, habe ich eine kleine Korrektur vorgeschlagen. Unten sind die Korrekturzeilen, die Sie in entsprechende Stellen eintragen können. Sie haben damit den unerwünschten Nachklang verboten. Weitere Verbesserungen können Sie dadurch anbringen, daß Sie

am Anfang des Programms ein Steuerzeichen für Groß/Graphik zufügen. Das schöne Klavierbild erscheint nicht auf dem Bildschirm, wenn sie vorher Groß/Klein-Modus eingestellt haben, um andere Arbeiten (z. B. Textverarbeitung) zu erledigen.

```
10 print "
230 K$="5662590908141716221925243033323835414643494854515053"
240 for i=1 to len(K$) step 2:K=(val(mid$(K$,i,2))=(i+1)/2):next
305 K=peek(203)
310 fr=f(K(K))/m:fl=sity*7:wFl+4:iffr=z then 420
ready.
```

Nach dem Motto mehr Spaß in den „Hackerbuden“!

## Computer-Kommunikation

### Spaß mit dem C-64

Das nachfolgende Kurzlisting ist ein tolles Kommunikationsprogramm mit Ihrem Computer. Tippen Sie die wenigen Zeilen ein, und warten Sie ab, was geschieht! Ein Tip: Der Text kann jederzeit durch neue Zeilen von Ihnen ersetzt werden.

```
0 data 162,223,160,0,185,0,160,153,0,160,200,208,247,238,24,10,238,27,10
1 data 232,208,236,169,54,133,1,169,85,141,117,164,169,191,141,119,164,96
80 for i=2578 to 2614:read q:poke i,q:next
90 sys 2578
100 a$="was befehlst du, mein meister ?"
110 a$=chr$(10)+chr$(13)+a$+chr$(13)+chr$(0)
130 for i=1 to len(a$)
140 poke 48980+i,asc(mid$(a$,i,1))
150 next
```

## Neu: Sonderzeichen werden durch Klartext ersetzt!

### Eingabe-Hinweise für Commodore C-64-Benutzer!

Um Ihnen die Eingabe unserer Listings zu erleichtern, werden künftig alle C-64-Listings ohne Steuerzeichen abgedruckt.

Alle Steuerzeichen werden in unseren LISTINGS durch Klartext, welcher die Taste kennzeichnet, ersetzt. Eine Tabelle der

Tastenfunktionen finden Sie in jedem Tronic-Magazin.

#### Tastenbezeichnungen für Steuercodes

(TRONIC-NORM)

DOWN	CURSOR UNTEN	TASTE NEBEN RECHTEM SHIFT
UP	CURSOR HOCH	SHIFT-TASTE & TASTE NEBEN RECHTEM SHIFT
CLR	CLEAR SCHIRM	SHIFT-TASTE & 2. TASTE GANZ RECHTS OBEN
INST	EINFUEGEN	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 (GROESSTE TASTE)
F1	FUNKTIONSTASTE	FUNKTIONSTASTE F1
F3	FUNKTIONSTASTE	FUNKTIONSTASTE F3
F5	FUNKTIONSTASTE	FUNKTIONSTASTE F5
F7	FUNKTIONSTASTE	FUNKTIONSTASTE F7
F2	FUNKTIONSTASTE	FUNKTIONSTASTE F2
F4	FUNKTIONSTASTE	FUNKTIONSTASTE F4
F6	FUNKTIONSTASTE	FUNKTIONSTASTE F6
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	GRUEN	CONTROL-TASTE & 6
BLUE	BLAU	CONTROL-TASTE & 7
YELLOW	GELB	CONTROL-TASTE & 8
RVSON	INVERSE EIN	CONTROL-TASTE & 9
RVSOFF	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	HELLGRUEN	COMMODORE-TASTE & 6
LIG.BLUE	HELLBLAU	COMMODORE-TASTE & 7
HGREY	HELLGRAU	COMMODORE-TASTE & 8
CTRL	CONTROL	CONTROL-TASTE ZUSAMMEN MIT DEM NACHFOLGENDEN ZEICHEN. (Z.B. CTRLA ENTSpricht CTRL & A)

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

Folgendes Beispiel demonstriert die Arbeitsweise unseres Systems:

STANDARD AUSDRUCK:

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

UNSER NEUER AUSDRUCK DER GLEICHEN ZEILEN:

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

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

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

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

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

tätigt werden muß. Auch einzelne oder mehrere Leerzeichen innerhalb von Anführungszeichen werden auf diese Art gekennzeichnet. Das bislang übliche Abzählen einzelner Zeichen entfällt somit völlig. Alle Zeichen außerhalb von den geschweiften Klammern werden normal abgedruckt und auch eingegeben.

Auf den ersten Blick hört sich das sicher etwas kompliziert an, ist jedoch in der Praxis ganz einfach. Wenn man sich erst an die in Klartext geschriebenen Steuerzeichen gewöhnt hat, wird man den großen Vorteil dieser Schreibweise erkennen.

## Checksummer C Version 1.0 für Commodore 64

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

### Arbeitsweise und Aufbau unseres Checksummers:

Unser Checksummer besteht aus einem kleinen Maschinenprogramm welches als Basic-Loader abgedruckt ist.

Tippen Sie diesen Loader ein und speichern ihn auf Kassette oder Diskette, denn Sie können ihn zukünftig immer wieder benutzen.

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

Auf diese Weise können Sie das gesamte Listing schnell und fehlerfrei eingeben. Interessant ist auch, daß bei der Eingabe von Zeilen die üblichen Abkürzungen benutzt werden können, ohne die Checksumme zu verändern. Leerzeichen außerhalb von Anführungszeichen werden ignoriert, da diese auf die Ausführung der einzelnen Befehle keinen Einfluß haben. Sie können Ihr Programm auch starten, denn der Checksummer und Ihr Programm beeinflussen sich nicht gegenseitig. Wollen Sie den Checksummer abschalten, geben Sie einfach POKE 1,55 ein oder betätigen Sie die Tasten-Kombina-

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

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

Bestellnummer:

CV10K Kassette

10 DM

CV10D Diskette

15 DM

```

0 REM ***** TRONIC CHECKSUMMER ***** <116>
1 REM VERSION C 1.0 BEI FRANK BRALL <10>
2 REM ----- <148>
3 FOR I=B32 TO 100B:READ A:S=S+A <68>
4 POKE I,A:NEXT I:IF S<>16397 THEN PRINT
"FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE
}{SPACE}(10-19)":STOP <32>
5 SYS B32:S=0:FOR I=5B464 TO 5B603:READ
A:S=S+A <21>
6 POKE I,A:NEXT I:IF S<>18919 THEN PRINT
"FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE
}{SPACE}(29-36)":STOP <219>
7 S=0:FOR I=48979 TO 49005:READ A:S=S+A <33>
8 POKE I,A:NEXT I:IF S<>2888 THEN PRINT"
FEHLER{SPACE}IN{SPACE}DATA-ZEILEN{SPACE}
!{SPACE}(46-47)":STOP <23>
9 PRINT "{DOWN2}AKTIVIEREN{SPACE}: {SPACE}
POKE{SPACE}1,53{SPACE18}AUSSCHALTEN: {SPA
CE}POKE{SPACE}1,55":NEW <26>
10 DATA169,0,133,254,162,1,189,137,3,133
,255,160,0,177,254,145,254,136 <176>
11 DATA20B,249,230,255,165,255,221,139,3
,20B,238,202,16,230,169,96,141,49 <147>
12 DATA165,169,228,141,50,165,169,53,133
,1,169,141,133,254,162,3,134,255 <225>
13 DATA160,0,177,254,240,7,32,202,241,20
0,76,120,3,169,2,141,32,208 <140>
14 DATA96,160,224,192,0,147,17,32,32,35,
35,35,35,32,84,82,79,78 <158>

```

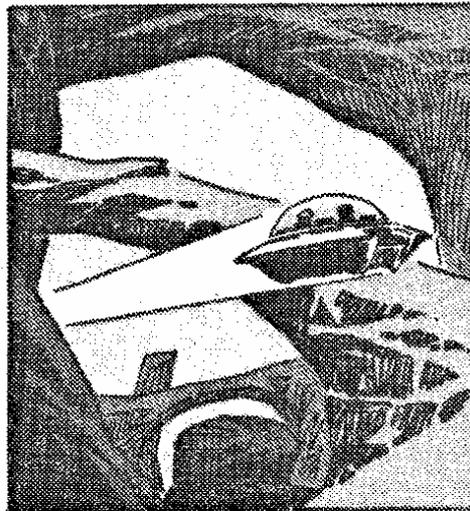
```

15 DATA73,67,45,86,69,82,76,65,71,32,67,
72,69,67,75,83,85,77 <129>
16 DATA77,69,82,32,35,35,35,35,139,13,13
,32,32,32,32,32,32 <147>
17 DATA32,32,32,32,32,86,69,82,83,73,79,
78,32,67,32,49,46,48 <79>
18 DATA13,13,32,32,32,32,32,32,40,67,41,
32,70,82,65,78,75,32 <94>
19 DATA66,82,65,76,76,32,40,49,48,46,56,
53,41,13,0 <12>
29 DATA169,0,141,107,191,141,108,191,141
,109,191,160,2,24,177,95,200,113 <7>
30 DATA95,141,109,191,160,3,200,177,95,2
40,44,201,34,208,10,173,107,191 <91>
31 DATA73,1,141,107,191,177,95,174,107,1
91,208,4,201,32,240,228,238,108 <128>
32 DATA191,174,108,191,24,177,95,109,109
,191,141,109,191,202,208,244,76,120 <244>
33 DATA228,56,76,183,228,72,32,201,255,1
70,104,144,1,138,96,32,240,255 <101>
34 DATA142,105,191,140,106,191,162,0,160
,0,24,32,240,255,169,91,32,12 <89>
35 DATA225,169,0,174,109,191,32,205,189,
169,93,76,83,191,173,33,208,145 <239>
36 DATA243,96,105,2,164,145,200,208,4,19
7,161,208,247,96 <202>
46 DATA32,12,225,32,63,171,32,63,171,172
,106,191,174,105,191,24,32,240 <213>
47 DATA255,76,128,164,24,0,0,5,200 <65>

```

## ASTRONAUT BATTLE für Commodore 64

**Schauplatz des Geschehens ist die feuchte, kühle und nur schwach erhellte Grotte Cadivan des schwarzen Planeten Dest, der aufgrund seiner Farbe seit Millionen von Jahren unentdeckt blieb und über immense Rohstoffvorkommen verfügt.**



Während Ihrer langen Forschungsreise durch die langgestreckten bizarren Felkanäle Dests treffen Sie auf ein zweites, Ihnen jedoch vollkommen fremdes Erkundungsteam. Dieses fremde Erkundungsteam eröffnet auch sogleich das Feuer auf Sie. Dank Ihres außerordentlich beweglichen und nicht gerade hilflosen Raumschiffs, entkommen Sie den ersten Schüssen Ihres Gegners und können Ihre Verteidigung einleiten. Bei diesem Spiel können Sie nur durch absolute Konzentration und genaues Steuern überleben. Hüten Sie sich also vor den gefährlichen Angriffen Ihres Gegners.

**Action für zwei Personen.**

**Gesteuert wird mit Joystick in Port 1 und Port 2.**

# programme

0 REM *****	<105>	2113 DATA182,129,191,191,129,157,157,129	<93>
1 REM *	<98>	,255	
2 REM * ASTRONAUT BATTLE *	<134>	2114 DATA183,129,189,251,247,231,231,231	<106>
3 REM * (C) 1985 *	<45>	,255	
4 REM * WRITTEN BY *	<42>	2115 DATA184,195,219,219,129,157,157,129	<160>
5 REM * CHRISTIAN WURZER *	<88>	,255	
6 REM * *	<103>	2116 DATA185,129,189,189,129,249,249,129	<152>
7 REM * FOR COMMODORE 64 *	<68>	,255	
8 REM * *	<105>	2118 DATA140,191,191,191,159,159,159,129	<224>
9 REM *****	<114>	,255	
10 :	<68>	2119 DATA137,239,239,239,231,231,231,231	<39>
50 POKE53280,0:POKE53281,0	<132>	,255	
2004 DATA120,169,51,133,1,169,0,133,95,1	<247>	2120 DATA150,189,189,189,219,219,231,231	<165>
33,90,133,88,169,208,133,96,169,240		,255	
2006 DATA133,89,169,224,133,91,32,191,16	<130>	2121 DATA133,129,191,191,135,159,159,129	<133>
3,169,55,133,1,88,96	<7>	,255	
2010 FORT=832T0832+33:READQ:POKET,Q:NEXT	<75>	2122 DATA147,129,189,191,129,249,185,129	<11>
2020 SYS832:POKE850,160:SYS832:POKE56576	<149>	,255	
,PEEK(56576)AND252:POKE53272,8		2123 DATA135,129,189,191,145,157,157,129	<143>
2030 POKE648,192	<53>	,255	
2031 PRINT"⟨CLEAR HGREY DOWN10 SPACE13⟩B	<126>	2124 DATA151,157,157,157,157,189,165,153	<166>
ITTE⟨SPACE⟩WARTEN"	<15>	,255	
2032 FRINT"⟨DOWN SPACE7⟩DATEN⟨SPACE⟩WERD	<243>	2127 DATA64,255,255,255,255,255,254,253,	<85>
EN⟨SPACE⟩EINGELESEN"	<78>	251	
2050 READX:IFX=-1THEN2060	<2>	2128 DATA65,255,255,248,199,63,255,255,2	<203>
2052 CG=57344+X*8:FORI=0TO7:READZ:POKECG	<11>	54	
+I,Z:NEXT:GOTO2050	<99>	2129 DATA66,243,192,0,236,252,252,254,25	<106>
2060 :	<110>	4	
2070 DATA27,255,255,255,255,255,255,255,	<188>	2130 DATA67,255,255,127,15,119,121,62,62	<148>
255	<121>	2131 DATA68,247,247,239,239,239,223,223,	<14>
2072 DATA28,0,223,223,223,0,251,251,251	<208>	223	
2074 DATA29,128,128,224,240,248,248,252,	<175>	2132 DATA69,30,142,192,224,224,240,248,2	<187>
255	<223>	52	
2076 DATA30,1,3,7,7,15,63,127,255	<194>	2133 DATA70,60,28,24,24,1,1,1,3	<88>
2078 DATA31,255,252,240,224,224,192,192,	<135>	2134 DATA71,127,191,223,239,239,239,247,	<176>
128	<21>	247	
2079 DATA33,255,127,15,7,7,3,3,1	<159>	2135 DATA72,191,191,191,127,127,127,127,	<34>
2080 DATA35,2,2,4,136,72,72,40,16	<6>	127	
2081 DATA129,195,219,129,157,157,157,157	<201>	2136 DATA73,255,255,255,255,224,128,39,7	<104>
,255	<252>	2	
2082 DATA142,141,181,181,149,149,149,153	<21>	2137 DATA74,254,254,255,255,0,0,255,63	<99>
,255	<147>	2138 DATA75,3,3,7,7,0,0,255,248	<239>
2083 DATA141,153,165,189,157,157,157,157	<89>	2139 DATA76,247,251,251,251,123,27,139,2	<99>
,255	<98>	03	
2084 DATA36,255,255,255,255,255,255,254,	<29>	2140 DATA77,127,127,127,127,127,191,191,	<114>
252	<132>	191	
2085 DATA37,239,207,207,151,55,119,119,2	<245>	2141 DATA78,86,121,126,127,127,127,191,1	<102>
43	<72>	91	
2086 DATA38,252,248,211,215,175,131,15,6		2142 DATA79,1,147,150,12,248,240,224,192	<178>
3		2143 DATAB0,231,210,33,63,31,15,7,3	<163>
2087 DATA40,3,3,251,249,253,241,249,253		2144 DATAB1,203,219,211,215,215,135,135,	<207>
2088 DATA148,131,239,239,231,231,231,231		159	
,255		2145 DATAB2,223,223,239,247,247,251,253,	<117>
2089 DATA146,129,189,189,129,155,157,157		254	
,255		2146 DATAB3,191,188,112,64,128,128,128,1	<208>
2090 DATA143,129,189,189,157,157,157,129		28	
,255		2147 DATAB4,57,124,124,124,56,24,24,24	<29>
2091 DATA130,131,219,193,205,205,205,129		2148 DATAB5,159,159,63,63,127,127,255,25	<151>
,255		5	
2092 DATA149,189,189,189,157,157,157,129		2149 DATAB6,64,224,240,248,252,254,255,2	<206>
,255		55	
2100 DATA176,129,189,185,149,141,157,129		2150 DATAB7,0,0,0,0,0,0,129,193	<62>
,255		2151 DATAB8,12,12,12,4,116,198,130,2	<97>
2102 DATA177,199,247,247,231,231,231,195		2152 DATAB9,255,127,127,127,255,127,127,	<112>
,255		127	
2109 DATA178,129,189,253,129,159,159,129		2153 DATA90,241,249,249,248,255,255,255,	<163>
,255		255	
2110 DATA179,193,253,253,225,249,249,129		2154 DATA91,2,1,1,129,255,255,255,255	<152>
,255		2155 DATA92,63,63,63,63,255,255,255,255	<228>
2111 DATA180,191,191,187,187,129,243,243		2156 DATA-1	<228>
,255		2157 :	<175>
2112 DATA181,193,223,223,193,249,185,129		3500 REM ----- MASCHINENPROGRAMM	<213>
,255		3501 REM -----	<205>

3520 DATA162,0,142,192,2,142,193,2,142,196,2,142,197,2	<84>	,73,201,162,1,142,33,208,32,126,201	<125>
3522 DATA206,252,207,174,252,207,224,0,208,102,162,30,142,252,207	<57>	3685 DATA162,8,142,33,208,32,126,201,162,0,142,33,208,174,68,195,224,176	<176>
3524 DATA173,0,220,41,239	<169>	3689 DATA208,1,96,174,30,208,206,68,195	<240>
3600 DATA141,254,207,162,0,142,253,207,3,2,0,200,173,0,220,201,112,176,74	<119>	3705 DATA174,13,208,224,131,208,10,162,1,142,205,2,162,255,142,204,2	<38>
3602 DATA173,21,208,41,2,201,2,240,65,16,2,0,142,194,2,142,195,2	<210>	3706 DATA174,13,208,224,160,208,10,162,2,55,142,205,2,162,1,142,204,2	<166>
3603 DATA173,16,208,41,253,141,16,208,41,1,201,1	<168>	4203 DATA162,15,169,128,141,61,3,141,60,3,222,208,2,208,44,189,176,2,157,208,2	<181>
3604 DATA208,8,173,16,208,9,2,141,16,208,174,0,208,142,2,208,174,1,208	<254>	4205 DATA189,192,2,240,33,16,12,189,0,208,8,222,0,208,40,240,7,208,19,254,0,208	<144>
3606 DATA142,3,208,162,0,32,172,200,174,0,220,224,109,240,11,32,102,201	<112>	4207 DATA208,14,173,60,3,208,9,173,61,3,77,16,208,141,16,208,173,60,3,208	<215>
3608 DATA173,21,208,9,2,141,21,208	<160>	4209 DATA3,78,61,3,73,128,141,60,3,202,1,6,191,76,24,196	<48>
3611 DATA206,251,207,174,251,207,224,0,208,104,162,30,142,251,207	<142>	4210 DATA-1	<242>
3612 DATA173,1,220,41,111,141,254,207,16,2,4,142,253,207,162,2,32,0,200	<238>	4215 DATA172,254,207,192,110,208,13,172,253,207,169,255,153,193,2,169,40	<195>
3613 DATA173,1,220	<233>	4216 DATA157,248,195,172,254,207,192,109,208,8,172,253,207,169,1,153,193,2	<252>
3614 DATA201,240,176,74,173,21,208,41,8,201,8,240,65,162,0,142,198,2,142,199,2	<2>	4217 DATA172,254,207,192,103,208,13,172,253,207,169,1,153,192,2,169,44	<179>
3615 DATA173,16,208,41,247,141,16,208,41,4,201,4	<201>	4218 DATA157,248,195,172,254,207,192,107,208,13,172,253,207,169,255,153,192,2	<230>
3616 DATA208,8,173,16,208,9,8,141,16,208,174,4,208,142,6,208,174,5,208	<200>	4219 DATA169,43,157,248,195,172,254,207,192,102,208,18,172,253,207,169,1	<182>
3618 DATA142,7,208,162,2,32,172,200,174,1,220,224,237,240,11,32,102,201	<42>	4220 DATA153,192,2,169,255,153,193,2,169,41,157,248,195,172,254,207,192,106	<49>
3620 DATA173,21,208,9,8,141,21,208	<6>	4221 DATA208,16,172,253,207,169,255,153,192,2,153,193,2,169,42,157,248,195	<91>
3622 :	<110>	4223 DATA172,254,207,192,105,208,18,172,253,207,169,255,153,192,2,169,1	<175>
3623 DATA173,30,208,141,249,207,41,2,201,2,240,12	<151>	4224 DATA153,193,2,169,46,157,248,195,17,2,254,207,192,101,208,16,172,253,207	<35>
3624 DATA173,31,208,141,255,207,41,2,201,2,208,8,173,21,208,41,253,141,21,208	<245>	4225 DATA169,1,153,192,2,153,193,2,169,4,5,157,248,195,96	<48>
3625 DATA173,249,207,41,8,201,8,240,9	<90>	4229 DATA188,248,195,192,40,208,13,172,2,53,207,169,255,153,195,2,169,47	<75>
3626 DATA173,255,207,41,8,201,8,208,8,173,21,208,41,247,141,21,208	<19>	4230 DATA157,249,195,188,248,195,192,44,208,13,172,253,207,169,1,153,194,2	<162>
3632 DATA174,0,208,142,8,208,174,4,208,1,42,10,208,174,1,208,232,232	<189>	4231 DATA169,50,157,249,195,188,248,195,192,43,208,13,172,253,207,169,255	<232>
3634 DATA142,9,208,174,5,208,232,232,142,11,208	<21>	4232 DATA153,194,2,169,51,157,249,195,18,8,248,195,192,41,208,18,172,253,207	<108>
3635 :	<123>	4234 DATA169,1,153,194,2,169,255,153,195,2,169,48,157,249,195,188,248,195	<138>
3637 DATA174,0,208,224,0,208,8,173,16,208,9,16,141,16,208,174,0,208,224,255	<108>	4236 DATA192,42,208,16,172,253,207,169,2,55,153,194,2,153,195,2,169,49	<243>
3639 DATA208,8,173,16,208,41,239,141,16,208,174,4,208,224,0,208,8,173,16,208	<204>	4237 DATA157,249,195,188,248,195,192,46,208,18,172,253,207,169,255,153,194,2	<134>
3641 DATA9,32,141,16,208,174,4,208,224,2,55,208,8,173,16,208,41,223,141,16,208	<130>	4238 DATA169,1,153,195,2,169,53,157,249,195,188,248,195,192,45,208,16	<9>
3645 :	<133>	4239 DATA172,253,207,169,1,153,194,2,153,195,2,169,52,157,249,195,96	<254>
3647 DATA206,250,207,174,250,207,224,0,208,43,162,99,142,250,207	<115>	4246 DATA162,0,142,4,212,162,13,142,5,21,2,142,6,212,162,13,142,1,212,162,255	<204>
3649 DATA174,252,195,224,54,208,17,238,2,53,195,238,252,195,162,2,142,43,208,142	<12>	4248 DATA142,0,212,162,129,142,4,212,96	<179>
3651 DATA44,208,24,144,14,206,253,195,206,252,195,162,1,142,43,208,142,44,208	<236>	4252 DATA162,0,142,11,212,162,11,142,12,212,142,13,212,162,23,142,8,212,162,129	<138>
3671 DATA173,255,207,41,1,201,1,240,9	<87>	4254 DATA142,11,212,96	<107>
3672 DATA173,249,207,41,1,201,1,208,72,1,62,40,142,0,208,142,8,208	<197>	4259 DATA162,25,160,255,192,0,240,3,136,208,249,202,224,0,208,242,96,-1	<237>
3673 DATA162,165,142,1,208,232,232,142,9,208,173,16,208,41,238,141,16,208	<236>	4260 :	<238>
3674 DATA173,21,208,41,247,141,21,208,32,73,201,162,1,142,33,208	<131>	5000 REM ---- SPRITES	<6>
3678 DATA32,126,201,162,8,142,33,208,32,126,201,162,0,142,33,208,174,41,195	<33>	5010 :	<223>
3679 DATA224,176,208,1,96,174,30,208,206,41,195	<246>	5014 DATA,,,,,,8,,,8,,,8,,,8,,,62,,,221,128,1,190,192,1,190,192,,221,128	<114>
3680 DATA173,255,207,41,4,201,4,240,9	<216>	5016 DATA,62,,,,,, 5020 DATA,,,,,,1,,,2,,,4,,,8,,,62,,,221,128,1,190,192,1,190,192,,221,128	<252>
3681 DATA173,249,207,41,4,201,4,208,72,1,62,45,142,4,208,142,10,208,162,165	<44>	5022 DATA,62,,,,,, 5026 DATA,,,,,,64,,,32,,,16,,,8,,,62	<116>
3682 DATA142,5,208,232,232,142,11,208,173,16,208,9,36,141,16,208	<24>		<2>
3683 DATA173,21,208,41,253,141,21,208,32			

# programme

```

,,221,128,1,190,192,1,190,192,,221,128 <227>
5028 DATA,62,,,,,,,,,,,,,,,,,,,,,,,,, <8>
5032 DATA,,,,,,,,,,,,,,,,,,,,,62,,,221, <134>
128,1,190,192,15,190,192,,221,128 <14>
5034 DATA,62,,,,,,,,,,,,,,,,,,,,,,,,, <203>
5038 DATA,,,,,,,,,,,,,,,,,,,,,62,,,221, <20>
128,1,190,192,1,190,248,,221,128 <96>
5040 DATA,62,,,,,,,,,,,,,,,,,,,,,,,,, <161>
5044 DATA,,,,,,,,,,,,,,,,,,,,,62,,,221, <146>
128,1,190,192,1,190,192,,221,128,1,62, <182>
5052 DATA2,,,4,,,,,,,,,,,,,,,,,,,,, <55>
5056 DATA,12,,,12,,,,,,,,,,,,,,,,,,,,, <247>
5062 DATA,,96,,,96,,,,,,,,,,,,,,,,,,,,, <26>
5068 DATA3,,,3,,,,,,,,,,,,,,,,,,,,, <182>
5074 DATA,,,,,,,,,,,,,,,,,,,,,3, <134>
,3,,,,,,,,,,,,,,,,,,,,,96,, <15>
5079 DATA,,,,,,,,,,,,,,,,,,,,,96,, <222>
,96,,,,,,,,,,,,,,,,,,,,, <74>
5083 DATA,,,,,,,,,,,,,,,,,,,,,3,,3,, <154>
,3,,3,, <3>
5087 DATA,,,,,,,,,,,,,,,,,,,,,96,,,96,, <159>
,96,,,96,, <3>
5090 DATA,,,,,,,,,,,,,,,,,,,,,20,,,46,,,50,,,46,,,38,,,28,,,28 <59>
,20,,,46,,,50,,,46,,,38,,,28,,,28 <61>
5092 DATA,,8,,,8, <54>
5096 DATA,,,,,,,,,,,,,,,,,,,,,28,,,50,,,46,,,50,,,58,,,20,,,28 <195>
,28,,,50,,,46,,,50,,,58,,,20,,,28 <54>
5097 DATA,,8,,,8, <136>
5100 DATA,,3,,,7,,,14,,,20,,,40,,,80,,,2 <198>
24,,1,192,,3,128,,5,,10,,20,,56 <143>
<205>
5102 DATA,,112,,,224,,1,64,,2,128,,5,,,1 <164>
4,,,28,,,24,, <74>
5104 : <197>
6000 REM ----- EINLESEN DER DATEN <218>
6002 : <128>
6004 FORT=0T0999:READQ:IFQ=-1THEN6008 <115>
6006 POKES0200+T,Q:NEXT <25>
6008 FORT=0T0999:READQ:IFQ=-1THEN6012 <148>
6010 POKES1200+T,Q:NEXT <169>
6012 : <79>
6020 FORT=0T062:READQ:POKES1712+T,Q:NEXT <202>
6022 FORT=0T062:READQ:POKES1776+T,Q:NEXT <223>
6024 FORT=0T062:READQ:POKES1840+T,Q:NEXT <235>
6026 FORT=0T062:READQ:POKES1904+T,Q:NEXT <0>
6028 FORT=0T062:READQ:POKES1968+T,Q:NEXT <21>
6030 FORT=0T062:READQ:POKES2032+T,Q:NEXT <33>
6032 FORT=0T062:READQ:POKES2096+T,Q:NEXT <54>
6034 FORT=0T062:READQ:POKES2160+T,Q:NEXT <247>
6036 FORT=0T062:READQ:POKES2224+T,Q:NEXT <151>
6038 FORT=0T062:READQ:POKES2288+T,Q:NEXT <175>
6040 FORT=0T062:READQ:POKES2352+T,Q:NEXT <161>
6042 FORT=0T062:READQ:POKES2416+T,Q:NEXT <115>
6044 FORT=0T062:READQ:POKES2480+T,Q:NEXT <8>
6046 FORT=0T062:READQ:POKES2544+T,Q:NEXT <220>
6048 FORT=0T062:READQ:POKES2608+T,Q:NEXT <255>
6050 FORT=0T062:READQ:POKES2672+T,Q:NEXT
6052 FORT=0T062:READQ:POKES2736+T,Q:NEXT
6054 :
7000 REM ----- VARIABLEN,DATEN
7002 :
7004 V=53248:POKE53265,PEEK(53265)AND239
7010 FORT=688T0719:POKET,0:NEXT
7014 FORT=39T045:POKEV+T,15:NEXT
7022 POKEV,40:POKEV+1,165:POKE688,1:POKE
689,1:POKES0168,41:POKEV+39,8
7024 POKE690,10:POKE691,10
7026 POKEV+4,45:POKEV+5,165:POKE692,1:PO
KE693,1:POKES0170,46 <95>
7028 POKE694,10:POKE695,10 <67>
7029 POKEV+12,143:POKEV+13,150:POKE716,1
:POKE717,255:POKE700,30:POKE701,30 <97>
7030 POKES0174,56 <243>
7031 : <204>
7040 POKEV+16,36:POKEV+21,1+4+16+32+64 <134>
7044 POKEV+27,1+4+16+32+64 <189>
7045 POKEV+30,0:POKEV+31,0 <215>
7046 : <219>
7050 POKES0172,54:POKES0173,54 <47>
8012 SI=54272:FORT=0T024:POKESI+T,0:NEXT <234>
8014 POKESI+24,31:POKESI+21,3:POKESI+22,
3:POKESI+23,241 <60>
8016 : <169>
9000 REM ---- GRAFIK <201>
9010 : <143>
9020 PRINT"(CLEAR BROWN)#####
#####" <159>
9022 PRINT"(UP)#####(SPACE23)!!
!" <177>
9024 PRINT"(UP)!(SPACE7)!!(SPACE25)!!
!" <116>
9026 PRINT"(UP)!(SPACE9)(SPACE25)^!!!" <101>
9028 PRINT"(UP)!(SPACE36)!!!" <251>
9030 PRINT"(UP)!(SPACE24)^(SPACE11)!!!" <184>
9032 PRINT"(UP)!(SPACE21)^1^(SPACE13)!!!" <206>
9034 PRINT"(UP)!(SPACE19 LIG.GREEN)#(BRO
WN)^!!!(SPACE14)!!!" <32>
9036 PRINT"(UP)!(SPACE18)!!!(SPACE10)
^1(LIG.GREEN)#(BROWN)^!!!" <85>
9038 PRINT"(UP)!!!(SPACE17)^(SPACE)!!(S
PACE10)!!!" <1>
9040 PRINT"(UP)!!!(SPACE16)!!(SPACE13)^
!!!" <60>
9042 PRINT"(UP)!!!(SPACES)^!!!(SPACES)!
(SPACE14)!!!" <123>
9044 PRINT"(UP)!!!(SPACES)^-+|-!(SPACE23
)!!!!" <61>
9046 PRINT"(UP)!(SPACE6)!!!(SPACE)
E20)!!!" <93>
9048 PRINT"(UP)!(SPACE6)^!|,|!!!(SPACE
22)!!!" <241>
9050 PRINT"(UP)!!!(SPACES)!!!(SPACE
16)^1(SPACES)!!!" <23>
9052 PRINT"(UP)!!!(SPACE4)^(SPACE)!(L
IG.GREEN)#(BROWN SPACE13)^(SPACE3
)^!!!" <241>
9054 PRINT"(UP)!!!(HGREY)\\(BROWN)#####
#####(HGREY)\\(BROWN
)^!!!" <38>
9056 PRINT"(UP)#####
#####" <17>
9058 PRINT"(UP)#####
#####" <17>
9060 PRINT"(UP)!!!(RVSON)LIVES(RVSOFF)!!
(RVSON)5(RVSOFF)!!!(RVSON)STRONAUT(RVSO
FF)!!!(RVSON)LIVES(RVSOFF)!!!(RVSON)5(
RVSOFF)!!!" <126>
9062 PRINT"(UP)#####&!!!(RVSON)BA
TTLE(RVSOFF)#####" <181>
9064 PRINT"(UP)#####
#####" <23>
9066 : <199>
9068 POKE49152+809,181:POKE49152+836,181 <103>
9070 POKE53265,PEEK(53265)OR16 <158>
9100 SYS50200 <31>
9150 POKE214,18:POKE211,15:SYS58640 <228>
9152 PRINT"(RVSON BROWN)'N'(SPACE)=(SPAC
E)NEW(SPACE)GAME" <206>
9164 GETA$:IFA$<>"N"THEN9164 <2>
9200 GOTO7000 <74>

```

In einem Programm oder während seiner Bearbeitung kommt es oft vor, daß man etwas (Anweisung, Zeichnung usw.) auf dem Bildschirm lassen möchte, wobei der Rest mehrmals gelöscht werden muß. Da bei dem C-64 diese Funktion nicht implementiert ist, wie es bei manchen Computern der Fall ist, soll das Programm „WINDOWS“ dieses umgehen, damit wir unsere unlöschbaren „FENSTER“ entwerfen können.

## WINDOW – Technik mit dem Commodore 64

Nachdem man das Programm fehlerfrei eingegeben und gestartet hat, liegt es in Speicher ab Adr.\$C000 bis \$C0C2.

Um die Wirkung dieser Funktion zu sehen, wollen wir einen kleinen Test machen und zwar:

- 1) Bildschirm löschen (Shift+CLR/HOME)
- 2) SYS49257:SYS49152 + Return
- 3) Positionieren des Cursors an eine beliebige Stelle des Bildschirms
- 4) SHIFT+SPACE drücken
- 5) Schreiben: DIESES IST EIN WINDOW!
- 6) SHIFT+SPACE drücken
- 7) SHIFT+RETURN drücken
- 8) SYS49312:POKE720,1 + Return
- 9) SHIFT+CLR/HOME

Nun stellen wir fest, daß der Satz „Dieses ist ein Window“ seine Farbe geändert hat und trotz des Löschens des Bildschirms noch angezeigt wird. Geben wir POKE720,0 ein, wird die Schriftfarbe wieder geändert in Schwarz(Farb-Code 0 ist bekanntlich Schwarz).

Die Speicherzelle dez.720, enthält also den aktuellen Code für die Farbe. Jetzt bewegen wir den Cursor an eine andere Stelle des Bildschirms und schreiben einen neuen Satz, nachdem man, vor und nach, mit den Tasten Shift+Space begrenzt hat. Mit SYS49312 wird dieser

auch ein Window. Es ist möglich, „Fenster“ in Senkrecht-Richtung zu verbreitern, indem man mehrere Zeilen untereinander schreibt. Vorausgesetzt, daß am Anfang und Ende die Tasten Shift+Space gedrückt worden sind.

Mit SYS49257 kann man den ganzen Bildschirm samt Windows löschen und mit SYS 49312 wieder neu entwerfen. Um aus dem Programm auszusteigen, muß RUN/STOP+RESTORE eingegeben werden. Der Inhalt der Windows bleibt trotz drücken der RUN/STOP+RESTORE-Taste erhalten. Dadurch kann man jederzeit diese Windows mit SYS49152 abrufen.

Eine mögliche Anwendung wäre, Windows in ein Programm einzubauen mit folgendem Format:

```
10 PRINTCHR$(160);"WINDOW";CHR$(160)
```

### Funktionsweise des Programmes „WINDOWS“ (Assembler Listing)

Das Programm ist in drei Teile aufgliedert, die im Folgenden erklärt werden sollen:

Als erstes sind die IRQ-Vektoren zu verändern, damit sie auf die Adresse \$C011 zeigen, wo eigentlich unsere Routine „Progr. Start“ anfängt.

#### ROUTINE: PROGR.START

Dieser Teil des Programmes schreibt – nachdem es geprüft hat, ob der Windows-

Inhalt am Anf. bzw. Ende mit Shift+Space gekennzeichnet ist – den Inhalt aus dem Speicherbereich \$C400-\$C7FF in den Original-Bildschirmspeicher ab \$0400. Dieser Vorgang wird durch die veränderten IRQ-Vektoren jede 1/60 sec. wiederholt.

#### ROUTINE: NEU

Durch SYS 49312 (\$C0A0) wird der Bildschirmspeicher ab Adr. \$0400-\$07FF,

nach Adr. \$C400-\$C7FF kopiert.

#### ROUTINE: AUS

Mit SYS49257 (\$C069) wird der Speicherbereich von \$C400-\$C7FF, der uns als „Hilfs-Bildschirmspeicher“ dient, auf Null gesetzt. Anschließend erfolgt das Zurückschreiben des Original-Inhaltes (\$EA31) in die Speicherzellen \$0314 bzw. \$0315.

C64 LISTING MIT CHECKSUMMEN (C V1.0)

50 REM #####	<84>	85 REM #####	<119>
55 REM #	<117>	100 PRINT " {CLEAR} ":FORI=0TO7:PRINT:NEXT:	
60 REM # W I N D O W S #	<130>	PRINTTAB(10) " ** {SPACE} W {SPACE} I {SPACE} N {	
65 REM # BY #	<239>	SPACE} D {SPACE} O {SPACE} W {SPACE} S {SPACE} **	<246>
70 REM # S.VALENTINO #	<250>	"	
75 REM # 6300 GIessen #	<47>	105 PRINT:PRINTTAB(12) " (C) {SPACE} S.VALEN	<117>
80 REM #	<142>	TINO"	
		110 PRINT:PRINT:PRINTTAB(8) "SYS {SPACE} 49	<55>
		152 {SPACE} = {SPACE} PROG.START "	

```

115 PRINT:PRINTTAB(8)"SYS{SPACE}49257{SP
ACE}={SPACE}WINDOW{SPACE}AUS" <218>
120 PRINT:PRINTTAB(8)"SYS{SPACE}49312{SP
ACE}={SPACE}WINDOW{SPACE}NEU" <58>
125 FORT=49152TO49359STEP16 <209>
130 FORI=0TO15:READA:S=S+A:POKET+I,A:NEX
T:I=+1:READB:IFB<>STHEN140 <206>
135 S=0:I=0:NEXT:END <197>
140 PRINT"FEHLER{SPACE}IN{SPACE}DATAZEIL
E{SPACE}NR.: "PEEK(63)+PEEK(64)*256:STOP <154>
1000 DATA120,169,017,141,020,003,169,192
,141,021,003,088,169,000,133,002,1388 <144>
1001 DATA096,165,002,041,007,170,189,144
,192,133,251,133,253,189,154,192,2311 <228>
1002 DATA133,252,189,149,192,133,254,160
,000,177,251,170,201,096,208,009,2574 <36>
1003 DATA165,002,073,128,133,002,076,063
,192,165,002,041,128,240,018,138,1566 <250>

```

```

1004 DATA145,253,166,254,138,024,105,212
,133,254,173,208,002,145,253,134,2599 <141>
1005 DATA254,200,192,200,208,211,230,002
,165,002,041,007,201,005,208,006,2132 <200>
1006 DATA165,002,041,000,133,002,076,049
,234,120,160,000,152,170,133,253,1690 <247>
1007 DATA169,196,133,254,177,253,041,000
,145,253,200,208,247,230,254,232,2992 <97>
1008 DATA224,004,208,240,169,049,141,020
,003,169,234,141,021,003,088,096,1810 <122>
1009 DATA000,200,144,088,032,004,004,005
,006,007,196,196,197,198,199,000,1476 <171>
1010 DATA120,160,000,152,170,133,251,133
,253,169,004,133,252,169,196,133,2428 <186>
1011 DATA254,177,251,145,253,200,208,249
,230,252,230,254,232,224,004,208,3371 <214>
1012 DATA240,088,096,000,000,000,000,000
,000,000,000,000,000,000,000,424 <59>

```

## UP Windows-Demo

Dieses Basic-Unterprogramm wurde für anspruchsvolle Anwender geschrieben. Damit ist es möglich, an jeder beliebigen Stelle und in jeder gewünschten Größe Windows zu entwerfen. Außerdem wird der Text innerhalb des Windows so geschrieben, als wenn man zusätzlich noch einen kleinen Bildschirm zur Verfügung hätte. Diese UP zu benutzen ist sehr einfach: man muß – nachdem das Maschinenprogramm im Speicher durch

den Basic-Loader geladen worden ist – sechs Variablen initialisieren und anschließend die Subroutine mit GOSUB 10000 abrufen. Folgende Variablen muß man vorher definieren:  
 X = Zeile Anfangs-Position des Windows (Werte 2 bis 38)  
 Y = Spalte Anfangs-Position des Windows (Werte 1 bis 25)  
 ZX = Zeilenlänge des Windows

(Werte 1 bis 39-X)  
 ZY = Spaltenlänge des Windows (Werte 1 bis 24-Y)  
 EL = Anzahl der Elemente des String Vektor ZL\$(KK) (Werte 1 bis 9). Falls mehr gebraucht werden soll, muß das Feld dimensioniert werden.  
 ZL\$(KK) = String, wo sich der Text befindet (max. 255 Zeichen).  
 In dem Demo-Programm sind die Zeilen von 60 bis 75, die dieses UP initialisieren.

### C64 LISTING MIT CHECKSUMMEN (C V1.0)

```

50 REM UNTERPROGR. INITIALISIEREN <201>
55 PRINT "{CLEAR}" <85>
60 X=10:Y=5:ZX=19:ZY=9:EL=3 <127>
65 ZL$(1)="===== {SPACE}W
{SPACE}I {SPACE}N {SPACE}D {SPACE}O {SPACE}W
{SPACE}S {SPACE}===== " <116>
70 ZL$(2)=" {SPACE}4 {D {SPACE}E {SPACE}M {SPA
CE}O {SPACE}4}===== {SPAC
E}BY {SPACE}7}===== " <180>
75 ZL$(3)="===== {SPACE}2}S. VALEN
TIND {SPACE}2}=====
===== " <24>
100 GOSUB10000:END <73>
200 REM UNTERPROGRAMM 'WINDOWS-DEMO' <156>
10000 YY$="":FORI=1TO25:YY$=YY$+CHR$(17)
:NEXT <39>
10005 FI=160:ER=0 <192>
10010 FOREE=1TO80:FF$=FF$+CHR$(32)+CHR$(
32):NEXT <216>
10015 X=X-1:Y=Y-1:ZX=ZX-1:ZY=ZY-1 <209>
10020 IFX+ZX>39DRX<10RZX<OTHENER=1:RETUR
N <147>
10025 IFY+ZY>24DRY<00RZY<OTHENER=1:RETUR
N <125>
10030 IFEL>9OREL<1THENER=1:RETURN <3>
10035 SYS49257 <38>
10040 FOREE=1TOEL:CK=CK+LEN(ZL$(EE)):NEX
T <61>
10045 FOREE=EL+1TO9:ZL$(EE)=FF$:NEXT:EL=
9 <131>
10050 AA=X:BB=Y:PRINT "{HOME}"LEFT$(YY$,B

```

```

B);SPC(AA-1);CHR$(FI) <207>
10055 FOREE=1TOEL <32>
10060 FORKK=1TOLEN(ZL$(EE)) <97>
10065 PRINT "{HOME}";LEFT$(YY$,BB);SPC(AA
);MID$(ZL$(EE),KK,1);CHR$(145) <192>
10070 AA=AA+1 <242>
10075 IFAA>X+ZXTHENGOSUB 10095 <101>
10080 IFBB>Y+ZYTHEN 10090 <181>
10085 NEXT:NEXT <8>
10090 SYS49312:SYS49152:RETURN <98>
10095 PRINT "{HOME}"LEFT$(YY$,BB);SPC(AA)
;CHR$(FI) <25>
10100 BB=BB+1:AA=X:IFBB<=Y+ZYTHENPRINT "{
HOME}"LEFT$(YY$,BB);SPC(AA-1)CHR$(FI) <88>
10110 RETURN <51>

```

**Das Assembler-Listing  
 zum Programm  
 „Windows“ finden Sie  
 auf der gegenüber-  
 liegenden Seite!**

# Assembler-Listing zur „Window“-Technik mit dem Commodore 64

```

.C000 78 SEI
.C001 A9 LDA #11
.C003 80 14 03 STA $314
.C004 A9 00 LDA #0
.C008 80 15 03 STA $315
.C00B 58 CLI
.C00C A9 00 LDA #0
.C00E 85 02 STA .02
.C010 60 RTS

.C011 A5 02 LDA .02
.C013 29 07 AND #07
.C015 44 TAX
.C016 8D 90 C0 LDA C090,X
.C019 65 FB STA FB
.C01B 85 FD STA FD
.C020 85 FC STA FC
.C022 8D 90 C0 LDA C090,X
.C025 85 FE STA FE
.C027 40 00 LDY #00
.C029 B1 FB LDA (FB),Y
.C02B AA TAX
.C02C C9 40 CMP #40
.C02E D0 09 BNE C039
.C030 A5 02 LDA .02
.C032 49 80 EOR #80
.C034 85 02 STA .02
.C036 4C 3F C0 JMP C03F

.C039 A5 02 LDA .02
.C03B 29 80 AND #80
.C03D F0 12 BEQ C051
.C03F 8A TAX
.C040 91 FD STA (FD),Y
.C042 A6 FE LDX FE
.C044 8A TAX
.C045 18 CLC
.C046 69 D4 ADC #D4
.C048 85 FE STA FE
.C04A AD D0 02 LDA $D0
.C04D 91 FD STA (FD),Y
.C04F 86 FE STX FE
.C051 C8 INY
.C052 C0 C8 CPY #C8
.C054 D0 D3 BNE C022
.C056 E4 02 INC .02
.C058 A5 02 LDA .02
.C05A 29 07 AND #07
.C05C C9 05 CMP #05
.C05E D0 04 BNE C066
.C060 A5 02 LDA .02

```

```

.C062 29 00 AND #00
.C064 85 02 STA .02
.C066 4C 31 EA JMP EA31
.C069 78 SEI
.C06A 40 00 LDY #00
.C06C 98 TYA
.C06D 4A TAX
.C06E 85 FD STA FD
.C070 A9 C4 LDA #C4
.C072 85 FE STA FE
.C074 B1 FD LDA (FD),Y
.C076 29 00 AND #00
.C078 91 FD STA (FD),Y
.C07A C8 INY
.C07B D0 F7 BNE C074
.C07D E4 FE INC FE
.C07F E8 INX
.C080 E0 04 CPX #04
.C082 D0 F0 BNE C074
.C084 A9 31 LDA #31
.C086 8D 14 03 STA $314
.C089 A9 EA LDA #EA
.C08B 8D 15 03 STA $315
.C08F 60 RTS

.C090 00 BRK
.C091 C8 INY
.C092 90 58 BCC C0EC
.C094 20 04 04 JSR 0404
.C097 05 06 ORA 06
.C099 07 ***
.C09A C4 C4 CPY C4
.C09C C5 C6 CMP C6
.C09E C7 ***
.C09F 00 BRK

.C0A0 78 SEI
.C0A1 40 00 LDY #00
.C0A3 98 TYA
.C0A4 AA TAX
.C0A5 85 FB STA FB
.C0A7 85 FD STA FD
.C0A9 A9 04 LDA #04
.C0AB 85 FC STA FC
.C0AD A9 C4 LDA #C4
.C0AF 85 FE STA FE
.C0B1 B1 FB LDA (FB),Y
.C0B3 91 FD STA (FD),Y

```

```

* IRG-Vektor
* neu setzen

* Zaehler auf null setzen
* ** Routine:Progr.Start **

* Schleifen fuer
* Speicherbereich
* Selektierung

* Leseschleife aus Hilfs-
* Bildschirmspeicher
* Shift+Space?
* Nein! Dann AND-sonst-EOR-Verknuepfung

* Zeichen in Akku
* Schreibeschleife in dem
* Original Bildschirmspeicher
* von $0400 bis $07FF

* dez.720 (Schriftfarbe)

* incrementiert Zaehler

* schon 5 Seiten?
* Nein! Wiederholen(zur IRG-Routine)
* Zaehler auf null setzen

```

```

* zuraeck zur IRG-Routine
* ***-Routine:Aus **

* Schreibt null in dem
* Hilfs-Bildschirmspeicher
* von $0400 bis $L7FF

* Normal Inhalt des IRG-Vektoren
* werden zurueckgeschrieben ($EA31)

* Zum Basic zurueck
* *** Tabelle ***

* Werte fuer Speicherbereich-
* Selektierung

* Kopienschleife
* Kopiert den Bildschirmspeicher
* von $0400 bis $07FF
* nach $C400 bis $7FF

* ***-Routine:Neu **

```

**Bitte achten Sie auf den Rest des  
Assembler-Listings auf der Seite 16!**

# tips & tricks

```

,COB5  C8      INV
,COB6  D0 F9   BNE COB1
,COB8  E6 FC   INC  FC
,COBA  E6 FE   INC  FE
,COBC  E8      INX
,COBD  E0 04   CPX  #04
,COBF  D0 F0   BNE COB1
,COCL  58      CLI
,COCD  60      RTS
    
```

und drückt die Play-Taste. Das Programm kann nur durch Stop/Restore abgebrochen werden. Die Tonqualität ist zwar nicht be- rauschend, jedoch ist Maschinensprache nicht schneller als Schall. Am besten sind gesprochene Wörter zu erkennen. Bei dem meisten Musik-Kassetten ist das Schlagzeug zu stark auf- genommen und ist deshalb sehr laut zu hören.

```

1000 FORK=16384TO16440:READG:P=P+G:POKEK
,B:NEXT <1>
1010 IFF<>6570THENPRINT"ERROR{SPACE}IN{S
PACE}DATA":END <76>
1020 PRINT"YOU{SPACE}CAN{SPACE}USE{SPACE
}THE{SPACE}GRAMMOPHON,IF{SPACE}YOU{SPACE
}START <32>
1030 PRINT"THE{SPACE}ROUTINE{SPACE}WITH{
SPACE}'SYS16384'{SPACE}THEN{SPACE}PUT <99>
1040 PRINT"A{SPACE}MUSIC{SPACE}CASSETTE{
SPACE}IN{SPACE}TAPE{SPACE}AND{SPACE}PRES
S{SPACE}PLAY <90>
1050 DATA169,52,141,20,3,169,0,141,32,20
8,141,33,208,32,68,229,173,13,220,41,16 <90>
1052 DATA240,249,169,13,141,24,212,162,0
,232,224,10,208,251,173,32,208,73,1,141 <122>
1054 DATA32,208,141,33,208,169,0,141,24,
212,141,13,220,76,16,64 <138>
    
```

## Tips & Tricks für den Commodore 64

### Grammophon

Mit diesem Programm, kann man sich über den Computer Kas- setten anhören. Man startet die Routine, legt eine Kassette ein

POKE's für C-64	
POKE XXXXX	BESCHREIBUNG
POKE 775,200	Listschutz EIN
POKE 775,167	Listschutz AUS
POKE 808,237	Run/Stop Restore EIN
POKE 808,225	Run/Stop Restore AUS
POKE 650,128	Dauerfunktion für alle Tasten EIN
POKE 650,64	Dauerfunktion für alle Tasten AUS
POKE 657,128	Shift-Modus EIN
POKE 657,0	Shift-Modus AUS
POKE 19,40	Return anders EIN
POKE 19,0	Return anders AUS
POKE 56325,5	Cursor schneller
POKE 56325,255	Cursor langsamer
POKE 204,0	GET-Befehl Cursor EIN
POKE 207,0:POKE 204,1	GET-Befehl Cursor AUS
POKE 819,246	Save-Schutz EIN
POKE 819,245	Save-Schutz AUS
POKE 808,49	Run/Stop EIN
POKE 808,237	Run/Stop AUS
POKE 19,64	INPUT ohne '?' (1) EIN
POKE 19,0	INPUT ohne '?' (1) AUS
POKE 792,255:POKE 793,252	Run/Stop Restore=>Reset EIN
POKE 792,71 :POKE 793,254	Run/Stop Restore=>Reset AUS

### Multi/BW

Nachdem die Routine gestartet wurde, wandelt sie ein farbiges Bild in ein einfar- biges Bild um. Bei Bildern, wo die Punkte zu eng aneinander liegen, eignet sich die Routine nicht, weil nachher alles zu un- scharf ist. Man sollte nur mehrfarbige Bil- der zum Umwandeln nehmen, da dabei das Bild größer wird.

**Das Listing finden Sie auf der gegenüberliegenden Seite!**

### Hirescopy

Wenn das Programm mit 'SYS4096' ge- startet wurde, druckt der Drucker das Bit- mapping-Bild aus. Er benötigt dafür nur 40 Zeichen pro Reihe, ist jedoch nur für einfarbige 7-Bit Printer, z. B. für den Sei- kosha GP-100VC, geeignet. Nach dem Drucken bleibt das Bild noch auf dem Bildschirm, der Druckvorgang ist aber schon beendet.

**Das Listing finden Sie auf der gegenüberliegenden Seite!**

## Listing zu den Tips & Tricks

### "Multi / BW"

```

1000 FORK=4096TO4460: INPUT "MNEMONIC{SPAC
E}"; MN$: L$=RIGHT$(MN$,1): H$=LEFT$(MN$,1) <215>
1010 IFL$>"@"ANDL$<"G" THENL=ASC(L$)-55 <215>
1011 IFH$>"@"ANDH$<"G" THENH=ASC(H$)-55 <28>
1012 IFL$>"/"ANDL$<": " THENL=ASC(L$)-48 <253>
1013 IFH$>"/"ANDH$<": " THENH=ASC(H$)-48 <66>
1020 POKEK,H*16+L: NEXT: POKE19B,0: WAIT19B
,1: SYS4411 <55>
    
```

```

1030 PRINT "{DOWN}PICTURE {SPACE}TO{SPACE}
BW{SPACE}WITH{SPACE}'SYS40000' <7>
1040 DATA169,59,141,17,208,169,24,141,24
,208,169,0,141,32,208,169,32,133,251 <42>
1042 DATA169,0,133,250,170,168,169,0,133
,252,177,250,41,192,240,6,169,192,101 <61>
1044 DATA252,133,252,177,250,41,48,240,6
,169,48,101,252,133,252,177,250,41,12 <181>
1046 DATA240,6,169,12,101,252,133,252,17
7,250,41,3,240,6,169,3,101,252,133,252 <22>
1048 DATA165,252,145,250,200,208,197,232
,230,251,224,32,208,190,169,4,433,251 <69>
1050 DATA162,0,160,0,169,16,145,250,200,
208,251,232,230,251,224,4,208,244,96 <212>
    
```

## Listing zu den Tips & Tricks

### "Hirescopy"

```

A9 04 AA 20 BA FF 20 C0 FF A2 04 20 C9 FF 20 18 11 A9 3B 8D 11 D0 A9 18 8D 18
D0 A9 00 8D 20 D0 A9 00 85 FA A9 20 85 FB A9 07 85 02 A9 28 85 03 A9 0D 20 D2
FF A9 1D 85 FC A9 28 85 06 A9 08 20 D2 FF A9 80 85 FD A2 00 A0 00 B1 FA 9D 40
03 20 96 10 E8 E0 07 D0 F1 A9 01 85 FE A9 80 85 FF A2 00 BD 40 03 25 FD F0 06
A5 FF 05 FE 85 FF 06 FE E8 A5 FE C9 80 D0 EA A5 FF 20 D2 FF 46 FD A5 FD C9 01
D0 D3 C6 06 D0 BA A9 0D 20 D2 FF C6 FC D0 A8 A9 04 4C C3 FF C6 02 F0 03 4C C0
10 18 A5 FA 69 08 85 FA 90 02 E6 FB A0 06 20 D9 10 08 D0 FA A9 07 85 02 C6 03
F0 01 60 A9 28 85 03 4C D9 10 A5 FA 29 07 C9 07 F0 03 E6 FA 60 E6 FB 18 A5 FA
69 39 85 FA 90 02 E6 FB 60 A5 FA 29 07 F0 03 C6 FA 60 C6 FB 38 A5 FA E9 39 85
FA B0 02 C6 FB 60 E6 FA 60 E6 FB 18 A5 FA 69 39 85 FA 90 02 E6 FB 60 A5 FA 29
07 F0 03 C6 FA 60 C6 FB 38 A5 FA E9 39 85 FA B0 02 C6 FB 60 20 44 E5 A9 40 85
FA A9 3F 85 FB A9 00 A8 91 FA E6 FA D0 02 E6 FB A5 FB C9 40 D0 EF A5 FA C9 80
D0 E9 60 A2 08 A0 01 20 BA FF A9 09 A2 5C A0 11 20 BD FF A9 00 85 03 A9 10 85
04 A2 3B A0 11 A9 03 4C D8 FF 48 49 52 45 53 43 4F 50 59
    
```

```

1000 FORK=4000TO40113: READG:P=P+G: POKEK
,B: NEXT <64>
1010 IFP<>17124 THENPRINT "ERROR {SPACE} IN{
SPACE}DATA": END <178>
1020 PRINT "ALL {SPACE} CORRECT. {SPACE} YOU{
SPACE} CAN {SPACE} FORM {SPACE} A {SPACE} MULTI
COLOUR <74>
    
```

## Was gibts Neues auf dem Software-Markt?

### NEW RELEASE: SCHNEIDER COMPILER

Einen Compiler für den Schneider CPC 464 hat die Firma Günster Computertechnik auf den Markt gebracht. Der Compiler, lieferbar in Disketten- und Kassettenversion, besitzt einen umfangreichen Befehlssatz und ist in der Lage, Fließkommaarithmetik zu verarbeiten.

Bei Anwendung des Compilers erzielt der Anwender eine sehr große Geschwindigkeitssteigerung, die im Bereich von 10-fach bis 50-fach möglich ist. So wird, um den Bildschirmspeicher zu löschen, eine Programmausführungszeit von 1,85 Sekunden benötigt, wohingegen das normale Schneider-BASIC bei identischem Programm 55,16 Sekunden läuft. Der Compiler wird auf Kassette für 79,-- DM und für Disketten für 89,-- DM geliefert mit ausführlicher Anleitung und Handbuch.

Nähere Informationen:  
SOFTWARE TEAM Joachim Günster,  
Mühlenstr. 12, 5431 Boden

## SOFT-TEAM-BERLIN

Software und Zubehör für

**C-64 Atari Schneider**

**Spectrum MSX**

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

Gratiskatalog anfordern:

**Anton Peter & Partner, Kamminerstraße 9  
1000 Berlin 10, Telefon 453 2711**

## „New Games“

für den Commodore 16

**Endlich ist es soweit! Bei den Softwarefirmen findet auch der C16/116 immer mehr Berücksichtigung.**

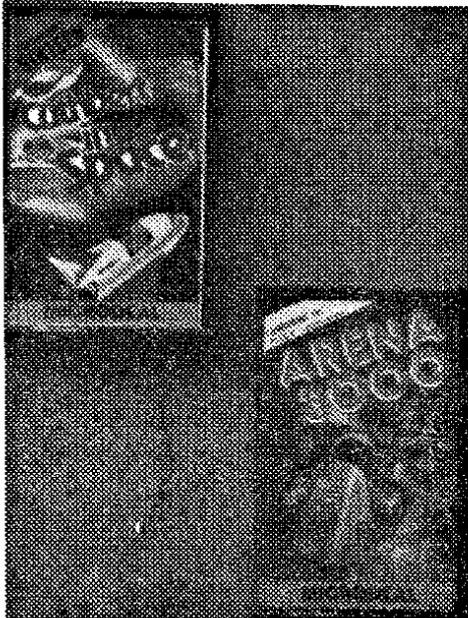
**So gibt es schon einige Programme im Fachhandel. Wir stellen Ihnen hier einige Neuerscheinungen für dieses Computersystem vor.**

### ARENA 3000

**ARENA 3000** versetzt den Spieler/die Spielerin in die Welt der Cyborgs. Zahlreiche Levels gilt es zu durchlaufen. Jedes Level – eine Arena – birgt die Gefahren der außerirdischen Angreifer, die nach Ihrem Leben trachten. Sie können sich in jede beliebige Richtung bewegen und die Feinde mit Ihrem Laser-Strahl vernichten. Dieses Spiel zeichnet sich durch seine enorme Schnelligkeit und den Variantenreichtum aus. (Beim Test dieses Programmes hatte unser Chef-Programmierer einen High Score von 1.385.455 Punkten!). Sie sehen also, dieses Spiel hat scheinbar „keine Grenzen“!

Hersteller: MICRODEAL

Preis: ca. 20 Mark



### CUTHBERT IN SPACE

**CUTHBERT IN THE SPACE** ist ein rasantes Space Shuttle-Arcade-Spiel, bei dem es darauf ankommt, das Raumschiff mit Treibstoff zu füllen, ohne aber von feindlichen Angreifern „liquidiert“ zu werden. Zu diesem „Zweck“ sind Sie mit den modernsten Waffen-Systemen ausgestattet, die es Ihnen erlauben, jeden Außerirdischen zu vernichten. Ein excellenter Spaß und viel Spielvergnügen ist Ihnen gewiß.

Hersteller MICRODEAL

Preis: ca. 28 Mark

### CUTHBERT ENTERS

### THE TOMBS OF DOOM

**CUTHBERT ENTERS THE TOMBS OF DOOM** ist ein typisches „Pyramiden-Such-Spiel“, bei dem unser Held allerlei wertvolle Kleinode aus dem „Reich der Toten“ (aus der Grabstätte eines Pharaos) sammeln muß. Dabei begegnen ihm „finstere Ge-



stalten“, die ihn bei seiner Grabräuberei ganz schön zu schaffen machen. Es ist ein Spiel mit viel Witz, einer hervorragenden Graphik und „spannenden Momenten“! Hersteller: MICRODEAL, 41, Truro Road, St. Austell, Cornwall PL25 5JE, England. Preis: ca. 28 Mark.

## Software-Test

### MASTERFILE für CPC

Die eindrucksvollen Möglichkeiten des CPC werden einem so richtig bewußt, wenn man einmal das Datenverwaltungsprogramm MASTERFILE von Campell Software benutzt. Man kann MASTERFILE ohne Übertreibung außergewöhnlich nennen, denn es enthält etliche Möglichkeiten, die man normalerweise bei vergleichbaren Programmen auf Homecomputern nicht finden wird.

Die Art der Information welche mit diesem Programm betrieben werden kann ist praktisch endlos. Natürlich speichert es die Adressen und Telefonnummern von Freunden, Verwandten, Geschäftspartner, Ärzten, Händler oder was immer Sie wollen. Und die gespeicherten Daten sind leicht zu handhaben und auf vielfältige

Weise darstellbar.

Ein File zu erstellen ist nicht so einfach, ist es doch auch die Grundlage für spätere „Kinder“-Files. Aber die vielen Seiten lange Programmbeschreibung geht sehr ins Detail und führt den Neuling bei seinen ersten Schritten sicher durch das Programm. Sind die Daten erst einmal eingegeben, können sie leicht verändert und bearbeitet werden. Verschiedene Möglichkeiten, wie die Sortierung in alphabetischer oder numerischer, in aufsteigender oder absteigender Reihenfolge sind vorhanden.

In der Suchroutine können verschiedene Kriterien miteinander verbunden werden und in den Möglichkeiten sind auch nu-

merische Vergleiche wie 'gleich' (=) oder 'größer als' (>), sowie direkte Nachweise wie 'vorhanden' oder 'nicht vorhanden' eingeschlossen. Dadurch können bei der Suche ganz komplexe Sachverhalte erfragt werden.

Das erstaunliche an diesem Programm sind die verwandtschaftlichen Verhältnisse von Files untereinander, eine Möglichkeit, die zum erstenmal auf einem Homecomputer verwirklicht wurde.

Angenommen ein File wurde errichtet und enthält die Korrespondenz des Geschäftsführers einer politischen Partei. Dabei sind etliche Briefe von Privatleuten, Schreiben von verschiedenen Organisationen und andere Dinge, die man mit der Post erhalten oder auch versandt hat.

Jedesmal den vollen Namen einer Organisation, die Anschrift und alle sonstigen wichtigen Grunddaten, z.B. Konto, Telefon,

Telex usw., einzugeben, würde eine umfangreiche aber unnötige Arbeit sein. MASTERFILE erlaubt es, ein „Eltern“-File zu erstellen, daß alle zentralen Informationen enthält, während alle anderen Eingaben, die mit dazu gehören, einem „Kinder“-File zugeordnet werden.

Das Kinder-File liest automatisch die Grunddaten vom Elternfile und stellt sie zur Verfügung, somit werden jedesmal alle passenden Daten wieder zusammengefügt.

In einigen professionellen Datenverwaltungsprogrammen – der dazu nötige Computer kostet ein Vielfaches dessen, was Sie für Ihren CPC ausgegeben haben – ist die Idee schon auf „Großeltern“- und „Enkel“-File erweitert worden. Diese werden hier noch nicht benutzt, obwohl das Thema damit noch nicht erledigt ist, denn es gibt schließlich auch „kinderlose El-

tern“-Files und „Waisen“-Files. Etwas kompliziert für den Anfang?

Das beste Datenverwaltungsprogramm, das ich kenne, liest direkt von der Diskette, so daß die Größe eines Files nur begrenzt wird von der Diskettenkapazität. Masterfile hat immerhin den respektablen Umfang von 32 k zur Verfügung und die meisten Anwendungen scheitern nicht an der Größe des Memory.

Dieses Programm ist ein Zeichen dafür, daß nun vorzügliche Software mit professionellen Eigenschaften auf den Markt kommt und es nur noch nötig ist, daß der Anwender richtig damit umgehen kann. Wieviele Geschäftsführer von Firmen, Organisationen, Vereinen usw. kennen die Möglichkeiten des Marktes und wissen diese richtig, d.h. preiswert zu nutzen.

**Hersteller:**

Campell Software; ca. 98,-- DM im Fachhandel.

## Unser Softwarefeuerwerk – zündende TITEL mit explosiven PREISEN

Für C-64	auf Tape / Disk	Für C-64	auf Tape / Disk
<i>Winter Games</i>	35,00 / 45,00	<i>Wizard &amp; Princes</i>	/ 49,00
<i>Summer Games II</i>	35,00 / 45,00	<i>Fahrenheit 451</i>	/ 56,00
<i>Elite (engl.)</i>	45,00 / 55,00	<i>Rendevouz w. Rama</i>	/ 56,00
<i>Kennedy Approach</i>	45,00 / 49,00	<i>Amazone</i>	/ 56,00
<i>Imp. Mission</i>	30,00 / 45,00	<i>Space Pilot</i>	15,00 /
<i>Shadowfire</i>	35,00 / 59,00	<i>Blue Thunder</i>	15,00 /
<i>Frank Brunos Box.</i>	28,00 / 45,00	<i>Beach Head II</i>	/ 45,00
<i>Dough Boy</i>	39,00 / 49,00	<i>Donald Ducks Pl.</i>	/ 49,00
<i>Rocky Horror Show</i>	28,00 / 38,00		
<i>Mystic Mansion</i>	39,00 / 50,00		
<i>Whyrlinurd</i>	39,00 /		
<i>Nodes of Yesod</i>	39,00 /		
<i>Nick Faldos Golf</i>	33,00 /		
<i>Ghetto Blaster</i>	37,00 /		
<i>Speed King</i>	29,00 /		
<i>Rocket Ball</i>	26,00 /		
<i>Fourth Protocol</i>	34,00 /		
<i>Cricket</i>	39,00 /		
		<b>Für C-16</b>	<b>auf Tape</b>
		<i>Las Vegas</i>	36,00
		<i>Air Combat Emulator</i>	52,00
		<i>Chess (Grandmaster)</i>	35,00
		<i>Daley Thomps. St.Ev.</i>	29,00
		<b>10 Disketten im Kunststoffalbum</b>	
		<b>zum SUPERPREIS</b>	<b>39,00</b>

Versand per NN oder Ver.-Scheck zzgl. 3,- DM. Fordern Sie unsere Preisliste mit 0,80 DM in Briefmarken an. Auslandsaufträge nur gegen Vorausscheck.

**M. M. SOFTWAREVERSAND, POSTFACH 801202 · 8000 MÜNCHEN 80, Tel. 089 – 47 23 68**

# programme

# Hueycobra

## für Commodore 16

# Helikopterflug im unterirdischen Höhlensystem

Sie müssen als Hubschrauberpilot einen Gefangenen aus dem unterirdischen Höhlensystem großer Bergketten retten. Das ist natürlich keine einfache Sache, da die Höhle sehr eng ist und mit Barrikaden geschützt wird. Zudem machen auch noch Monster die Bergungsarbeiten hinderlich.

Das Spiel erstreckt sich über drei Bilder, die automatisch weiterblättern. Ab Bild 2 kommt für jeden zählenden Treffer ein Monster hinzu. Jeder Treffer ergibt 10 Punkte, und bei 1000 Punkten erhalten Sie einen Bonus-Hubschrauber. Am Ende der Höhle nehmen Sie dann den Gefangenen auf und fliegen mit ihm zurück zur Plattform, die mit „Ziel“ markiert

ist. Steuern können Sie den Hubschrauber mit dem Joystick.

### Hinweise zum Ablippen:

Um einen korrekten Spielablauf zu gewährleisten, wurden einige Leerzeichen verwendet, die der Bildfolge dienen.

Rechte Bildseite bei Höhlenende:

„C=“ + „K“ Taste

Linke Seite bei Höhlenanfang:

„SHIFT“ + „Space“ Taste

Über der Plattform:

„SHIFT“ + „-“ Taste

```
10 REM (C)RALPH HORNEBER
100 COLOR0,2,5:COLOR4,2,3:VOL8
110 V=65280
140 POKEV+18,PEEK(V+18)AND251
```

```
150 POKEV+19,PEEK(V+19)AND30R48
155 POKE740,56:CLR
160 FORT=832TO849:READA:POKET,A:NEXT:SYS832
180 FORT=1280TO13071:READA:POKET,A:NEXT
```

```
1440 DATA162,0,189,0,208,157,0,48,189,0,209,157,0,49,202,208,241,96
1450 DATA255,6,31,47,79,127,16,32
1452 DATA240,0,193,195,254,252,128,128
1454 DATA6,6,31,47,79,127,16,32
1456 DATA0,0,193,195,254,252,128,128
1458 DATA15,0,131,195,127,95,1,1
1460 DATA255,96,248,244,242,254,16,8
1462 DATA0,0,131,195,127,95,1,1
1464 DATA96,96,248,244,242,254,16,8
1482 DATA98,82,74,70,98,82,74,70
1484 DATA0,0,0,8,8,77,239,255
1486 DATA255,239,77,8,8,0,0,0
1488 DATA255,255,239,109,77,68,64,0
1490 DATA0,64,68,77,109,239,255,255
1492 DATA7,127,15,3,31,63,7,31
1494 DATA248,128,240,252,224,192,252,224
1496 DATA3,127,15,3,3,63,31,3
1500 DATA252,128,240,224,252,192,224,252
1502 DATA254,240,168,204,200,128,64,0
1504 DATA127,15,21,51,19,1,2,0
1506 DATA0,64,128,200,204,168,240,254
1508 DATA0,2,1,19,51,21,15,127
```

```
1510 DATA24,60,126,153,255,129,129,231
1512 DATA129,129,255,195,126,60,60,66
1514 DATA0,60,60,195,195,102,60,0
1516 DATA60,0,129,153,24,0,195,24
1518 DATA102,0,8,129,0,0,0,129
1520 DATA0,0,0,0,0,0,0,0
1522 DATA255,255,255,255,255,255,255,255
1524 DATA24,153,90,60,24,24,36,66
1526 DATA0,0,0,0,0,0,0,0
1528 DATA0,0,0,0,0,60,0,0
1530 DATA0,16,16,16,16,16,16,0
1532 DATA0,0,0,0,0,0,0,0
1534 DATA0,0,0,0,0,0,0,0
1900 ZZ=0:A=68:BI=1:HU=3
1910 GOTO5000
1998 REM SPIELROUTINE
2000 X=JOY(1):Q=Q+1
2010 IFU=0ANDBI<>1THENGOSUB3800
2020 IFX>9THENX=X-128:GOSUB2300
2030 IFX=0THEN2120
2035 Q=0
```

```
2040 ONXGOTO2050,2100,2060,2100,2070,2100,2080,2100
2050 ZZ=-40:B=40:C=41:D=-40:E=-39:GOTO2100
2060 ZZ=1:A=68:B=-1:C=-1:D=2:E=2:GOTO2100
2070 ZZ=40:B=-40:C=-39:D=40:E=41:GOTO2100
2080 ZZ=-1:A=64:B=2:C=2:D=-1:E=-1:GOTO2100
2100 IFPEEK(Z+D)<>32ORPEEK(Z+E)<>32THEN4000:ELSEZ=Z+ZZ
2105 POKEZ+B,32:POKEZ+C,32
2110 POKEZ-1024,12:POKEZ-1023,12
2120 POKEZ,A:POKEZ+1,A+1
```

```

2130 SOUND3,200,2
2140 IFA=64THENA=66:GOTO2190
2150 IFA=68THENA=70:GOTO2190
2160 IFA=66THENA=64:GOTO2190
2170 IFA=70THENA=68:GOTO2190
2190 IFQ=20ANDZZ<>0THEN:Q=0:GOTO2100
2195 IFZZ=0THENQ=0
2200 GOTO2000
2298 REM RAKETE
2300 SOUND3,100,B

```

```

2310 IFA=64ORA=66THENPOKEZ,64:POKEZ+1,65
2320 IFA=68ORA=70THENPOKEZ,68:POKEZ+1,69
2330 IFA<68ANDZZ=-40THEN2400
2340 IFA<68ANDZZ=-1THEN2435
2350 IFA<68ANDZZ=40THEN2460
2360 IFA>67ANDZZ=-40THEN2490
2370 IFA>67ANDZZ=1THEN2520
2380 IFA>67ANDZZ=40THEN2550
2400 FORS=Z-40TOZ-80STEP-40

```

```

2405 IFPEEK(S)<>32THENP=S+40:GOSUB3500:RETURN
2410 POKES,95:IFS=Z-40THENPOKES,32:ELSEPOKES+40,32
2415 NEXT:POKES+40,32:RETURN
2435 FORS=Z-1TOZ-40STEP-1
2440 IFPEEK(S)<>32THENP=S+1:GOSUB3500:RETURN
2445 POKES,94:IFS=Z-1THENPOKES,32:ELSEPOKES+1,32
2450 NEXT:POKES+1,32:RETURN
2460 FORS=Z+40TOZ+80STEP40
2465 IFPEEK(S)<>32THENP=S-40:GOSUB3500:RETURN
2470 POKES,95:IFS=Z+40THENPOKES,32:ELSEPOKES-40,32
2475 NEXT:POKES-40,32:RETURN
2480 REM
2490 FORS=Z-39TOZ-80STEP-40
2495 IFPEEK(S)<>32THENP=S+40:GOSUB3500:RETURN
2500 POKES,95:IFS=Z-39THENPOKES,32:ELSEPOKES+40,32
2510 NEXT:POKES+40,32:RETURN
2520 FORS=Z+2TOZ+38
2525 IFPEEK(S)<>32THENP=S-1:GOSUB3500:RETURN
2530 POKES,94:IFS=Z+2THENPOKES,32:ELSEPOKES-1,32
2535 NEXT:POKES-1,32:RETURN
2550 FORS=Z+41TOZ+80STEP+40
2555 IFPEEK(S)<>32THENP=S-40:GOSUB3500:RETURN
2560 POKES,95:IFS=Z+41THENPOKES,32:ELSEPOKES-40,32
2565 NEXT:POKES-40,32:RETURN
3498 REM TREFFER
3500 IFPEEK(S)=72ORPEEK(S)=85ORPEEK(S)=86THEN3510:ELSEPOKEP,32:RETURN
3510 IFPEEK(P)=94ORPEEK(P)=95THENPOKEP,32
3540 SOUND3,800,22
3550 FOR T=87TO89
3560 POKE S,T:FORI=1TO20:NEXTI,T
3570 POKES,32
3580 PU=PU+10:PA=PA+10:IFPA>1000THENPA=0:HU=HU+1:GOSUB13000
3590 CHAR,13,1,"":PRINTUSING"#####";PU
3600 U=0:RETURN
3798 REM MONSTER
3800 MD=INT(RND(1)*200+(Z-120))
3810 IFPEEK(MD)<>32THEN3800
3820 POKEMD-1024,50:POKEMD,INT(RND(1)*2+85)
3850 U=1:RETURN
3998 REM KOLLISION
4000 IFPEEK(Z+D)=92ORPEEK(Z+E)=92THENRI=1:POKE3982,32:SOUND1,800,20:GOTO2000
4010 IFPEEK(Z+D)=96ANDRI=0THENBI=BI-1:GOTO5000
4020 IFPEEK(Z+D)=97ANDRI=1THENBI=BI+1:GOTO5000
4030 IFPEEK(Z+D)=96ANDRI=1THENBI=BI-1:GOTO5000
4040 IFPEEK(Z+D)=97ANDRI=0THENBI=BI+1:GOTO5000
4050 IFPEEK(Z+E)=93ANDRI=1THEN5160
4060 Z=Z+ZZ

```

**Kennen Sie  
schon die  
Tronic-Top-Hit  
Programm-  
pakete  
für den C-16?**

*Wenn Sie mehr darüber  
wissen möchten, sehen  
Sie schnell auf der Seite  
32 nach.*

*Hier finden Sie tolle An-  
gebote für weitere Com-  
putersysteme.*



```

11000 REMBILD2
11020 SCNCLR
11040 GOSUB13000
11070 PRINT"+++++
11080 PRINT"+++++
11090 PRINT"+
11100 PRINT"
11110 PRINT"
11120 PRINT"
11130 PRINT"
11140 PRINT"
11150 PRINT"
11160 PRINT"
11170 PRINT"+
11180 PRINT"+
11190 PRINT"+
11200 PRINT"+
11210 PRINT"+
11220 PRINT"+
11230 PRINT"+
11240 PRINT"+
11250 PRINT"+
11260 PRINT"+
11270 PRINT"+
11280 PRINT"
11290 PRINT" :POKE4071,91
11400 IFR1=1 THENZ=3908:A=64:GOTO11420
11410 Z=3353:A=68
11420 POKEZ-1024,12:POKEZ-1023,12:POKEZ,A:POKEZ+1,A+1
11430 GOSUB13100
11500 RETURN
12000 REM BILD3
12030 SCNCLR
12040 GOSUB13000
12070 PRINT"+++++
12080 PRINT"+++++
12090 PRINT"+++++
12100 PRINT"+++++
12110 PRINT"+++++
12120 PRINT"+++++
12130 PRINT"+++++
12140 PRINT"+++++
12150 PRINT"+++++
12160 PRINT"+++++
12170 PRINT"+
12180 PRINT"+
12200 PRINT"+
12210 PRINT"+
12220 PRINT"+
12230 PRINT"+
12240 PRINT"
12250 PRINT"
12260 PRINT"
12270 PRINT"
12280 PRINT"
12290 PRINT" :POKE4071,91
12300 PRINT"+++++
12400 Z=3834:A=68

```

# programme

```
12410 POKEZ-1024,12:POKEZ-1023,12:POKEZ,A:POKEZ+1,A+1
12500 GOSUB13100: RETURN
13000 CHAR,12,0,"HUEYCOBRA"
13010 CHAR,5,1,"PUNKTE":PRINTUSING"####";PU;
13020 PRINT"HELIKOPTER";:PRINTUSING"##";HU:COLOR1,1
13050 RETURN
13100 FORT=1T015
13120 MO=INT(RND(1)*799+3192)
13130 IFPEEK(MO)<>32THEN13120
13140 POKEMO-1024,50:POKEMO,INT(RND(1)*2+85)
13150 NEXT:RETURN
```

## Hardcopyroutine

für den C16/116 + MPS801  
(oder kompatibel) s. Listing 3

Die Grafikeigenschaften des C16/116 sind hervorstechend und werden soweit man sich mit dem Arbeiten auf dem Bildschirm zufrieden gibt voll vom Basic des C16/116 unterstützt. Doch als Besitzer eines Druckers möchte man natürlich seine grafischen Kunstwerke, mathematischen Funktionsgraphen oder was sich sonst auf dem Bildschirm befindet gerne aufs Papier bannen. Hier bietet sich zunächst die Programmierung in Basic an, da eine Hardcopyroutine in Basic recht einfach und kurz zu bewerkstelligen ist; doch hat solch eine Routine einen entscheidenden Nachteil: die Geschwindigkeit!!!

Zeiten von einer halben Stunde sind dabei keine Seltenheit.

Es bleibt also keine andere Wahl, als auf Maschinensprache zurückzugreifen. Sie finden hier eine 216 Bytes lange Hardcopy-Routine für den C16/116 in Verbindung mit einem MPS801 oder einem kompatiblen Drucker (GP100VC, etc.). Mit Hilfe dieser Routine ist der Grafikbildschirm in weniger als 3 Minuten vollständig ausgedruckt.

Leider wurde kein idealer Speicherraum für die Routine gefunden (die Benutzung des Kassettenpuffers hätte die Datasettenanwender von der Nutzung des Pro-

gramms ausgeschlossen; die 192 Bytes hinter dem Grafikspeicher werden beim Löschen des Grafikbildschirms jedesmal mit Nullen aufgefüllt), so daß nichts anderes übrigbleibt, als weitere 200 Bytes vom ohnehin knappen Speicherraum des C16/116 abzuschneiden.

Gehen Sie beim Abtippen folgendermaßen vor:

- Geben Sie das Listing ein, speichern Sie es ab und starten dann mit RUN. Datasettenanwender nehmen vorher folgende Änderungen vor:  
In der Zeile 260 ist die 08 durch 01 und in Zeile 320 die 8 durch 1 zu ersetzen.
- Haben Sie beim Abtippen einen Fehler gemacht, meldet das Programm Ihnen dieses, ansonsten wird die Hardcopyroutine unter dem Namen 'hardcopy-code' als reines Maschinenprogramm abgespeichert. Dazu printet das Programm alle nötigen Anweisungen auf den Bildschirm; Sie brauchen nur noch, wenn der Cursor blinkt, auf die RETURN-Taste drücken. Dies geschieht viermal.

Für eigene Anwendungen müssen Sie die Routine nun absolut nachladen und den benötigten Speicherraum schützen. Dies erledigen die beiden folgenden Zeilen die

Sie an den ANFANG Ihres Programms stellen müssen:

```
O IFA=OTHENA=1:LOAD"HARDCOPY.CODE",8,1
1 GRAPHIC2:GRAPHICO,1:POKE51,32:POKE52,
23:POKE55,32:POKE56,23:CLR
```

Es versteht sich, daß Ihr Programm und die Hardcopyroutine sich auf derselben Diskette befinden müssen bzw., daß auf Band die Hardcopyroutine direkt hinter Ihrem Programm sein muß. Gestartet wird die Routine dann mit SYS5928!

### Änderungshinweis:

Bei manchen Hires-Bildern empfiehlt es sich, diese revers auszudrucken, da die Darstellung auf dem Bildschirm nicht der auf Papier entspricht. Dies ist zum Beispiel bei dem Hires-Bild auf der Commodore-Demo-Diskette der Fall (s. Demoausdruck). Diese Möglichkeit ist optimal in der Routine eingebaut. Wünschen Sie also eine Hardcopyroutine, die das Bild vor dem Ausdruck invertiert, müssen Sie an dem Listing vor dem Starten folgende Änderungen vornehmen:

In Zeile 115 sind das 11. und 12. Datum (... ,234,234,...) zu ändern in 73,127. Entsprechend muß die Prüfsummenabfrage in Zeile 190 geändert werden: der Zahlenwert von 23100 muß geändert werden in 22832.

```
10 REM HARDCOPY-ROUTINE FUER DEN COMMODORE 16/116 IN VERBINDUNG MIT EINEM
20 REM MPS-801 ODER KAMPATIBELN DRUCKER
30 REM WRITTEN 1985 BY WALDEMAR RAAZ
40 GRAPHICO,1
50 REM MASCHINENCODE
```

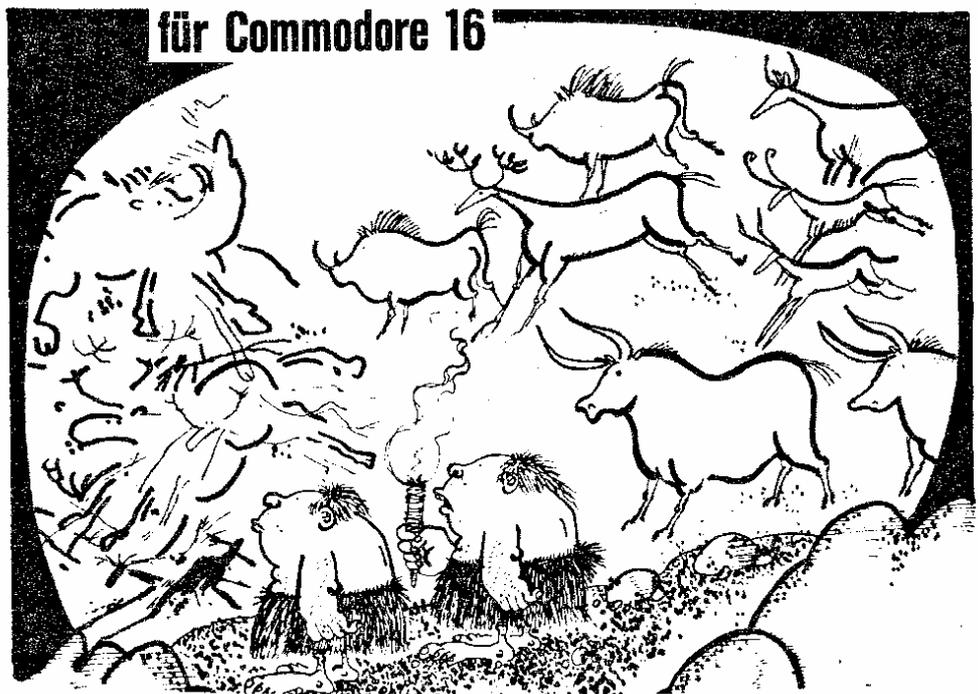
```

110 DATA169,127,133,172,169,4,133,174,169,0,133,171,32,192,255,162,127,32
111 DATA201,255,169,13,32,210,255,169,8,32,210,255,169,192,133,2,169,30,133
112 DATA3,169,0,133,4,169,32,133,5,169,28,133,8,169,7,133,6,169,40,133,7
113 DATA164,6,192,7,240,20,162,7,177,2,10,126,51,3,202,16,249,200,192,8,208
114 DATA240,165,6,240,18,160,0,162,7,177,4,10,126,51,3,202,16,249,200,196
115 DATA6,208,240,162,7,189,51,3,56,106,234,234,164,8,208,2,41,143,32,210
116 DATA255,202,16,237,24,169,8,101,2,133,2,144,2,230,3,169,8,24,101,4,133
117 DATA4,144,2,230,5,198,7,208,165,169,13,32,210,255,198,8,208,7,169,3,133
118 DATA6,76,94,23,16,18,169,15,32,210,255,169,13,32,210,255,32,204,255,169
119 DATA127,76,195,255,198,6,48,3,76,94,23,165,2,133,4,165,3,133,5,56,165
120 DATA2,233,64,133,2,165,3,233,1,133,3,76,90,23
170 SU=0:RESTORE
180 FORI=5928TD6143:READA:POKEI,A:SU=SU+A:NEXT
190 IFSUK>23100THENPRINT" FEHLER IN DATAS !!!":STOP
200 PRINT" OK! MASCHINENPROGRAMM EINGELESEN!"
210 PRINT:PRINT"JETZT MUSS ES NUR NOCH ABGESPEICHERT"
220 PRINT"WERDEN. DRUECKEN SIE HIERZU VIERMAL DIE"
230 PRINT"RETURN-TASTE:"
240 PRINT:PRINT"MONITOR"
250 PRINT:PRINT:PRINT:PRINT:PRINT
260 PRINT"S"CHR$(34)"HARDCOPY.CODE"CHR$(34)",08,1728,1800"
270 PRINT:PRINT:PRINT"X"
280 PRINT:PRINT"GOTO300"
290 PRINT"END"
300 GRAPHICO,1
310 PRINT"DAS PROGRAMM KANN JEDERZEIT MIT"
320 PRINT"LOAD"CHR$(34)"HARDCOPY.CODE"CHR$(34)",8,1
330 PRINT"NACHGELADEN UND DANN MIT"
340 PRINT"SYS 5928"
350 PRINT"GESTARTET WERDEN !"

```

## Helmut und die Zeitmaschine

Durch eine Zeitverschiebung ist Helmut ins Jahr 1000 v. Chr. zurückversetzt worden. Er muß nun durch das Einsammeln von Zeitschwertern (5 Stück pro Epoche) versuchen, wieder in die Gegenwart zu kommen. Insgesamt gibt es 8 Epochen und mit C 128 kann man in die Zukunft gelangen. Aber natürlich ist das Ganze nicht so einfach. Ein Diener des myhenischen Kaisers ist Helmut ständig auf den Fersen und versucht ihn zu erreichen. Starten können Sie das Spiel mit der SPACE-Taste und steuern mit dem Joystick. Die Puste von Helmut nimmt durch die Aufnahme eines Schwertes zu. Zur Aufnahme eines Schwertes müssen Sie auf dem Schwert stehen und den Feuerknopf drücken. Auf diese Weise können auch Steine zerschlagen werden. Der Diener des Kaisers kann nicht bekämpft werden.



Mensch Helmut wie kommen wir hier bloß wieder weg?

# programme

```
10 REM*****
20 REM*
30 REM* H E L M U T
40 REM* BY RALF ADLING
50 REM* LINGENER STR.6
60 REM* 4576 BERGE
70 REM*
80 REM*****

140 RESTORE860: FORT=832TO849: READA: POKET, A: PR=PR+A: NEXTT
150 SYS832
160 FORT=12800TO13000: READA: IFA>-1 THENPOKET, A: NEXT
170 B=3301: Z=1: X=1000: T=0
180 GOTO550: K=3321: B=3301
190 GOSUB480: COLOR1, 2: E=500: CHAR1, 1, 1, "PSTE: ": PRINTE: K=3321: B=3301
200 REM
210 REM MAINROUTINE
220 REM
230 CHAR1, 30, 1, "LEVEL: ": PRINTZ: CHAR1, 16, 1, Z$(Z)
240 IFPEEK(B+40)=32ANDPEEK(B+40)<4096THENC=40: SOUND1, 1000-B/4, 5

250 CHAR1, 6, 1, ": ": PRINTUSING"####"; E
260 IFT>4THEN1170
270 POKEK, 32
280 IFK-B>40THENL=-40: GOTO330
290 IFK-B<-40THENL=40: GOTO330
300 IFK>BTHENL=-.5: GOTO330
310 IFK<BTHENL=.5
320 IFB=K+LTHEN1110
330 IFPEEK(K+L)>4000THEN360
340 IFPEEK(K+L)=32OR65THENK=K+L: GOTO360
350 K=K-L

360 POKEK, 71
370 IFJOY(1)=3THENC=C+1
380 IFJOY(1)=7THENC=C-1
390 IFJOY(1)=128THENGOSUB490: E=E-25
400 IFJOY(1)=1THENC=-80: E=E-5
410 POKEB, 32
420 B=B+C: IFPEEK(B)=32THENPOKEB, 64: GOTO440
430 B=B-C: POKEB, 64
440 C=0: E=E-1: IFE<0THEN1140
450 REM
460 GOTO240

470 SOUND1, 800, 10: SOUND2, 770, 4: SOUND1, 854, 5: POKEB, 32: GOSUB480: E=E+400-Z*10: RETURN
N
480 N=INT(RND(1)*17)+5: M=INT(RND(1)*35)+1: CHAR1, M, N, "_": RETURN
490 IFPEEK(B+40)=70THENGOSUB470: T=T+1
500 IFB<3980THENPOKEB+40, 32: SOUND3, 1000, 5: SOUND3, 444, 5
510 RETURN
520 REM
530 REM INTRO
540 REM
550 GOSUB1940: COLOR1, 8, 7
560 CHAR1, 3, 4, "  C  C  CCCC  C  CCCC  C  C  CCCC"
570 CHAR1, 3, 5, "  C  C  C  C  C  C  C  C  C  C  C  C"
580 CHAR1, 3, 6, "CCCC  CCCC  C  C  C  C  C  C  C"
590 CHAR1, 3, 7, "  C  C  C  C  C  C  C  C  C  C  C  C"
600 CHAR1, 3, 8, "  C  C  CCCC  CCCC  C  C  CCCC  C"
610 COLOR1, 3, 7: CHAR1, 8, 12, "- BY COLOSSUS-SOFT": CHAR1, 12, 14, "1 9 8 5"
620 CHAR1, 14, 18, "\T": CHAR1, 14, 19, "/●": CHAR1, 14, 20, "Γ_"
630 Z$(1)="1000 V. CHR. ": Z$(2)="500 V. CHR. ": Z$(3)="44 V. CHR. ": Z$(4)="400 N. CHR. "
640 Z$(5)="800 N. CHR. ": Z$(6)="1750 N. CHR. ": Z$(7)="1940 N. CHR. ": Z$(8)="1985 N. CHR. "
"
650 REM
660 REM LIED
670 REM
680 DATA1, 881, 10, 2, 0, 0, 1, 834, 20, 2, 0, 0, 1, 810, 10, 2, 881, 10
690 DATA1, 796, 20, 2, 834, 20, 1, 768, 10, 2, 810, 10, 1, 739, 20, 2, 796, 20, 1, 768, 10, 2, 768, 10
700 DATA1, 796, 5, 2, 739, 10, 1, 768, 5, 1, 739, 5, 2, 644, 10, 1, 1022, 5
710 DATA1, 796, 10, 2, 739, 10, 1, 834, 20, 2, 796, 20, 1, 854, 10, 2, 810, 10
720 DATA1, 834, 20, 2, 796, 20, 1, 810, 10, 2, 768, 10, 1, 796, 30, 2, 739, 30, 1, 768, 20, 2, 721, 20
730 DATA1, 881, 10, 2, 0, 0, 1, 834, 20, 2, 0, 0, 1, 810, 10, 2, 881, 10
740 DATA1, 796, 20, 2, 834, 20, 1, 768, 10, 2, 810, 10, 1, 739, 20, 2, 796, 20, 1, 768, 10, 2, 768, 10
```

# programme

```
750 DATA1,796,5,2,739,10,1,768,5,1,739,5,2,644,10,1,1022,5
760 DATA1,796,10,2,739,10,1,834,20,2,796,20,1,854,10,2,810,10
770 DATA1,834,20,2,796,20,1,810,10,2,768,10,1,796,40,2,739,40,1,768,40,2,721,40
780 RESTORE&B0:FORU=1TO34:READD,E,F,G,H,I:SOUNDG,E,F:SOUNDG,H,I
790 NEXT:SOUND1,796,110:SOUND2,881,110
800 IFR=90THEN190
810 GETA$:IFA$=" "THEN1410
820 GOTO810
830 REM
840 REM DATAS FUER ZEICHENSATZ
850 REM
860 DATA162,0,189,0,208,157,0,48,189,0,209,157,0,49,202,208,241,96

870 DATA8,124,12,10,8,20,34,34
880 DATA255,66,36,24,24,36,66,255
890 DATA255,255,195,195,195,195,255,255
900 DATA60,66,185,161,161,185,66,60
910 DATA28,28,8,63,125,189,125,61
920 DATA61,60,36,36,36,36,36,36
930 DATA16,16,16,16,16,124,16,56
940 DATA56,56,16,255,56,56,68,68
950 DATA0,0,0,0,8,8,141,255
960 DATA255,195,195,60,60,195,195,255
970 DATA146,146,146,146,146,146,146,146
980 DATA255,0,0,255,0,0,255,0

990 DATA0,31,31,31,0,248,248,248
1000 DATA3,3,1,1,15,31,55,55
1010 DATA55,51,51,51,55,7,6,6
1020 DATA12,12,24,24,24,24,24,56
1030 DATA192,192,128,128,240,240,240,240
1040 DATA240,208,216,204,235,224,96,96
1050 DATA48,48,24,24,24,24,24,28
1060 DATA0,0,0,0,254,0,0,0
1070 DATA31,34,68,248,31,34,68,248
1080 DATA248,66,33,31,248,68,34,31
1090 DATA0,0,183,149,183,165,183,0
1100 DATA-1

1110 FORI=850TO100STEP-2:SOUND1,I,1:POKEB,70:COLOR0,3,INT(RND(1)*7):NEXT
1120 GOSUB1270
1130 GOTO90
1140 FORI=850TO100STEP-5:SOUND1,I,1:SOUND2,I,1:POKEB,72:COLOR0,3,INT(RND(1)*7):N
EXT
1150 GOSUB1270
1160 GOTO90
1170 T=0:Z=Z+1
1180 PU=PU+(2000+Z*100)--80:IFPU<0THENPU=0
1190 R=90:IFZ=2THEN1690
1200 IFZ=3THEN1590
1210 IFZ=4THEN1830

1220 IFZ=5THEN1790
1230 IFZ=6THEN1860
1240 IFZ=7THEN1550
1250 IFZ=8THEN1460
1260 IFZ=9THENGOSUB1270:GOTO1350

1270 SCNCLR:COLOR0,1:CHAR1,0,0,"-----"
1280 CHAR1,0,1,"-----"
1290 COLOR1,8,5:CHAR1,10,10,"WIR SCHREIBEN":CHAR1,13,12,"DAS JAHR"
1300 IFZ=9THENRETURN
1310 CHAR1,14,15,"":PRINT$(Z)
1320 CHAR1,18,10,"IE":CHAR1,10,18,"ALS HELMUT STARB":CHAR1,14,20,"|"
1330 FORU=1TO2000:NEXT
1340 RETURN
1350 CHAR1,13,14,"":PRINT"1985"
1360 CHAR1,10,16,CHR$(130):PRINT"HELMUT IST WIEDER DA!!!!!!"
1370 GOTO90
1380 REM
1390 REM LABYRINTHE
1400 REM
1410 GOSUB1940:COLOR1,3,6:FORI=17TO23:CHAR1,4,I,"*|||||||||||||||||||||||||
":NEXT
1420 FORI=4TO37STEP4:CHAR1,I,15,"-":CHAR1,I,14,"-":NEXT
1430 FORI=13TO23:CHAR1,I,I,"":NEXT:CHAR1,4,16,"*****"
"
1440 COLOR1,8,7:CHAR1,1,12,"┌-":CHAR1,1,11,"/●":CHAR1,1,10,"└"
1450 GOTO190
1460 GOSUB1940:COLOR1,10,3
1470 FORI=17TO23:CHAR1,I,I,"█" "":NEXT:COLOR
```

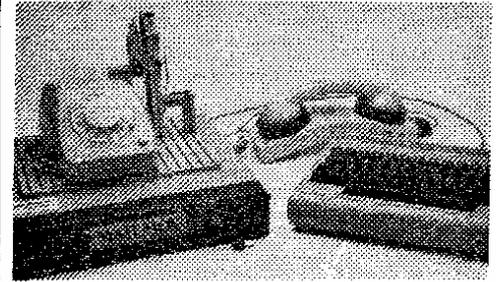


## Polarisationsfilter zu günstigen Preisen

Fortsetzung von Seite 1

Die Handhabung dieser Filter ist denkbar einfach. Die im Lieferumfang enthaltenen Befestigungsklemmen werden auf dem oberen Teil des Bildschirms oder Monitors aufgeklebt. Anschließend werden die beiden Halterstangen in den Filter gesteckt und diese vor den Schirm montiert. Sie werden sofort den Unterschied feststellen und Ihre Augen werden nicht mehr so

stark beansprucht wie vorher. Die Redaktion ist der Meinung, daß es sich bei diesem Zubehörteil um ein sinnvolles und zugleich nützliches Gerät handelt, das eigentlich vor jedem Bildschirm oder Monitor montiert sein sollte. Dadurch dürfte die Diskussion um ein zu langes Arbeiten vor dem Monitor etwas entschärft werden. Auch der Homecomputeranwender sollte an seine Gesundheit denken und von dem Angebot Gebrauch machen.



- 1 Interface für CBM 64 (Black Box)
- 1 Akustik-Koppler (V24) mit entsprechenden Kabeln

## Die „DO IT YOURSELF“ Mailbox

Alle Bastelfreunde haben die Möglichkeit die „Do it yourself“ Mailbox in Betrieb zu nehmen. Sie benötigen dazu:  
1 Commodore 64 (oder anderen Compu-

ter)  
1 VC 1541 Floppy  
1 Fischertechnik Computing Baukasten (Teilweise)

Die Fa. Spima Computer, Mannheim, hat das Programmpaket Mbox.11 speziell für den Commodore 64 zugeschnitten und bietet für einen Gesamtpreis von ca. 400,- DM den Fischertechnikbaukasten und das Interface an.

## Die neuen Hitrans-Modems von CDI

**Immer mehr Anbieter drängen sich auf dem DFÜ-Sektor. Die Firma CDI stellt nun gleich mehrere Modem's und Akustikkoppler auf einmal vor. Interessant ist auch, daß alle Geräte eine FTZ Nummer besitzen, was wohl noch immer nicht selbstverständlich ist.**

Hitrans-Modems sind als Akustikkoppler ausgelegt. Dies bedeutet, daß keine elektrische Verbindung zwischen Modem und Telefonnetz hergestellt werden muß. Der Telefonhörer wird einfach in die Gummimuffen des Hitrans-Kopplers eingesteckt. Die Stromversorgung erfolgt je nach Modell über Batterie, Netzteil oder sogar

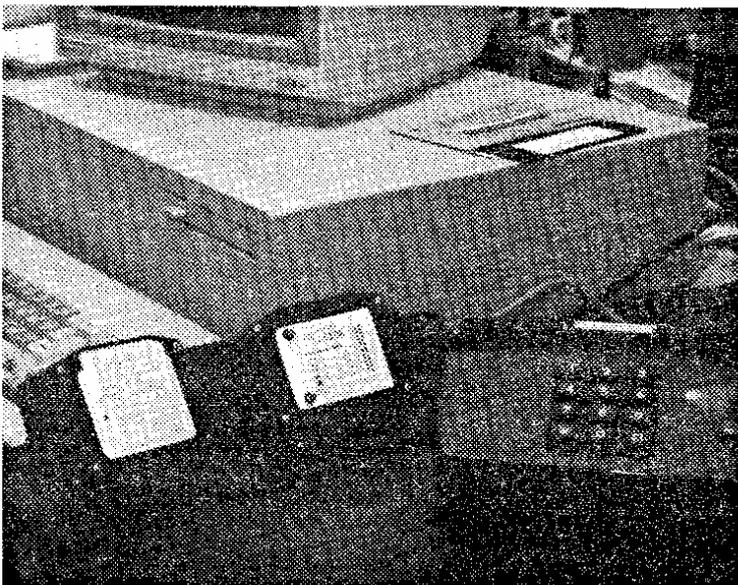
durch beide Möglichkeiten. Der uns zu Testzwecken zur Verfügung gestellte Koppler (Modell Nr. 05106) arbeitet wahlweise mit 9V Block-Batterie oder einem 9V Gleichspannungsnetzteil. Im Test machte sich jedoch die Batteriewechselung etwas negativ bemerkbar, da dazu 4 fest angezogene Schrauben gelöst wer-

den müssen, was bei einem relativ hohen Stromverbrauch öfter wiederholt werden muß. Nachdem wir unser Terminalprogramm (COM 64) auf Halbtublex umgeschaltet, und die Schalter am Koppler in die richtige Stellung (grün/ein gelb/aus) gebracht hatten, lief die Datenübertragung sehr sicher ab. Umgebungsgeräusche schienen unserem Koppler nichts auszumachen, lediglich einige etwas schwach übertragene Mailboxen sorgten für einige Übertragungsfehler.

Etwas problematisch verlief auch die Handhabung, denn legt man den Koppler flach auf einen Tisch, wie es eigentlich üblich ist, werden immer die eingebauten Tasten betätigt. Neben diesen kleinen Schönheitsfehlern scheint die Firma CDI einige hervorragende Produkte entwickelt zu haben.

### CDI Lieferprogramm

**HITRANS 300 P:**  
(Professional) Übertragungsgeschwindigkeit 300 Baud, (30 Z/sec.), FDX, Originate/Answer automatische Freischaltung der Echosperre, Stromversorgung 220 V, Batterie, Accu-Betrieb oder Versorgung über den Schnittstellenstecker möglich.  
Farbe: beige,  
ZZF Nr. 18.13.2015.00



## HITRANS 300 C:

(Consumer) Übertragungsgeschwindigkeit 300 Baud, FDX, Originate/Answer, Stromversorgung über Schnittstellenstecker, externes Netzteil 12 V, 40 mA oder über Blockbatterie. Farbe: schwarz, ZZF Nr. 18.13.2015.00

## HITRANS 1200:

Übertragungsgeschwindigkeit Senden: 1200 Baud, Empfangen: 1200 oder 75 Baud umschaltbar 1200/1200 HDX-oder

1200/75 FDX-Betrieb, Stromversorgung über Schnittstellenstecker, Netzteil 12 V, 30 mA, Batterie oder Accu, ZZF Nr. 18.13.2017.00

## HITRANS 75/1200:

BTX-zugelassener Akustikkoppler, Senden: 75 Baud, Empfangen: 1200 Baud, V.24 Schnittstelle, optional DBT 03, Stromversorgung Netzteil 12 V, 30 mA, Batterie oder Accu, ZZF Nr. 18.13.2020.00

## HITRANS U:

Universalkoppler 300 Baud FDX, Originale-Answer und 1200 Baud HDX umschaltbar, Stromversorgung über den Schnittstellenstecker, Netzteil  $\pm 15$  Volt, 1600 mA, ZZF Nr. 18.13.2016.00

## HITRANS MUFF:

300/1200 Baud Sendeteil (nur Senden möglich), ZZF Nr. 18.13.2018.00

# BTX – Commodore 64 Software per Telefon

Erfreuliche Nachrichten für die über 700.000 Commodore 64 User in der Bundesrepublik Deutschland. Jeder Besitzer eines C-64 kann durch den Erwerb eines Decodermoduls SD-64 zum Kreis der Btx-Teilnehmer gehören. Nach der Entwicklung eines Btx-Decoders durch Commodore gibt es bereits ein weiteres Modul, das nach Angaben des Herstellers den bisherigen, weitaus teureren Geräten in Intelligenz und Funktion weit überlegen ist. Dadurch wird Btx zum Kommunikations-Medium für jedermann, denn schon mit wenigen Handgriffen ist Ihr Commodore 64 zum Abrufterminal für Bildschirmtext umfunktioniert. Der Tronic-Verlag plant in Zusammenarbeit mit dem Hersteller des Moduls, Btx auch für die Zeitschriftenleser sinnvoll einzusetzen. Dabei gehen die Überlegungen dahin, daß bereits ab Anfang des Jahres 1986 ein „Telesoft-Vertrieb“ im Tronic-Verlag eingerichtet wird. Hier soll der Kunde die Möglichkeit erhalten, qualitativ hochwertige Software über Telefon abzurufen. Dabei liegen die Vorteile für den Verbraucher in der Tatsache, daß er schnell, ohne Bestell- und Lieferzeiten und günstig an gute Software gelangt. Die Telefongebühren für den Btx-Service sind sehr gering, da im gesamten Bundesgebiet zum Nah-Tarif telefoniert werden kann. Somit ist Btx eigentlich nicht nur für die professionelle Nutzung gedacht, sondern kann von jedermann genutzt werden.

Nähere Einzelheiten zum „Telesoft-Vertrieb“ und das neuentwickelte Modul erhalten Sie bei der Programmierabteilung

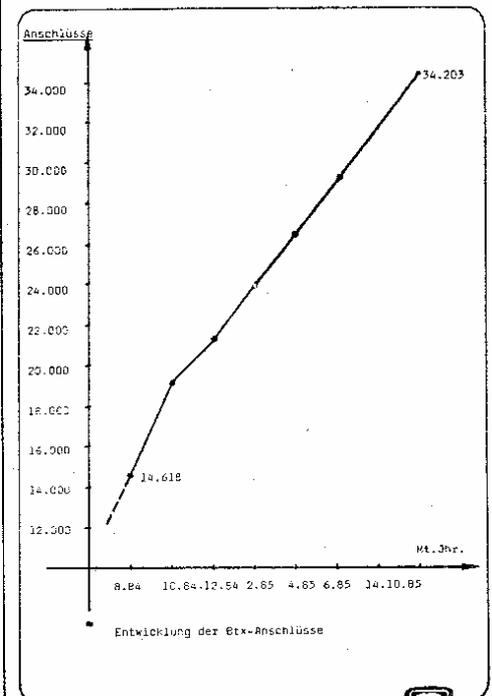
unseres Hauses bzw. in einer unserer nächsten Ausgaben.

## Siemens beurteilt Entwicklung des Btx-Dienstes positiv

Mit den ersten Prognosen zum seinerzeit erst geplanten und in der Erprobungsphase befindlichen Bildschirmtextdienst (Btx) wurden diesem neuen Kommunikationsmedium enorme Vorschußlorbeeren zuerkannt. Diese Schätzungen basierten auf der Annahme, daß über Btx die elektronische Kommunikation auf breiter Basis Einzug in den privaten Haushalt finden würde. Darüberhinaus setzten diese Prognosen einen termingerechten Start des Btx-Systems sowie das gleichzeitige Vorhandensein der notwendigen Endgerätekennzeichen voraus. Beides wurde erst mit deutlicher Verzögerung realisiert.

Die heute im Btx-Dienst angebotene Statistik zeigt über den Zeitraum des Dienstes von seinem Beginn an betrachtet folgende Entwicklung: Der Start des Dienstes im August 1984 war gekennzeichnet durch eine hohe Beteiligung von Informationsanbietern, die von Anfang an mit sehr umfangreichem und komplexem Informationsangebot vertreten waren. Die Zahl der Informationsanbieter hat sich in nur einem Jahr um über ein Viertel (27 %) kontinuierlich erhöht. Die angebotenen Informationsseiten nahmen in der gleichen

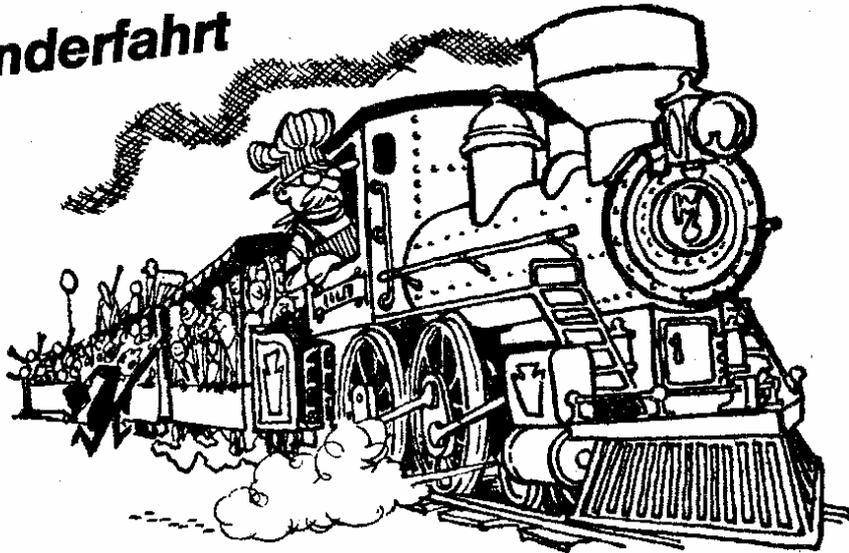
Zeit auf fast das Doppelte zu. Das zeigt, daß die Informationsanbieter heute ihr komplexes Angebot intensiv ausgebaut, strukturiert und bedienerfreundlich gegliedert haben. Trotz der technischen Anlaufprobleme hat sich die Zahl der Btx-Teilnehmer von 14 000 um rund 20 000 auf 34 000 erhöht.



## LOCOS

für die Grundversion  
des VC-20

### Der VC-20 Schnellzug auf Sonderfahrt



Haben Sie schon einmal als Signalwärter Ihren Dienst versehen? Jeder VC-20 Besitzer kommt bei diesem Programm in den Genuß, den Zugverkehr sicher zu steuern. Es klingt zwar sehr einfach, ist es aber nicht!

Der Zug fährt in der linken oberen Bildschirmhälfte los und Sie müssen nun mit Hilfe der Tastatur (Zahlen 1 - 6) den Zug steuern und an den entsprechenden Stellen die Weichen richtig stellen. Insgesamt gibt es 6 Weichen die an den entsprechenden Punkten durch Druck auf die richtige Zahlentaste um eine Drehrichtung verändert werden können.

Die Lok erhöht mit zunehmender Spieldauer automatisch die Geschwindigkeit. Die Spielbeschreibung innerhalb der Programmzeilen ist in englischer Sprache, da der Autor des Programmes aus Amerika stammt.

Wir hoffen, Sie haben trotzdem viel Spaß mit dem Programm und es gelingt Ihnen, den Zug so zu steuern, daß er nicht aus den Gleisen kommt.

#### Teil 1:

```

10 PRINT "L":PRINTTAB(222) "INSTRUCTIONS Y/N ?"
20 GETA$: IFA$="" THEN 20
30 IFA$="Y" THEN GOSUB 320
40 PRINT "L" TAB(224) "MOMENT PLEASE"
50 FOR I=7168 TO 7679: POKE I, PEEK(I+25600): NEXT
60 POKE 52, 28: POKE 56, 28: CLR
70 FOR I=7168 TO 7263: READ A: POKE I, A: NEXT
80 POKE 36869, 255: POKE 7678, 1
90 PRINT CHR$(147): FOR I=1 TO 40: PRINT "L LOCOS ";: NEXT
100 Y=128: W=36878: X=36876: M=7701: FOR I=1 TO 243: T(V)=PEEK(M+1)
110 V=V+1: IF V=2 THEN V=0
120 M=M+1: POKE M, 8: POKE M+30720, 0: POKE M-1, T(V)
130 A=A+1: IFA=1 OR A>7 THEN A=2
140 Y=Y+1: IF Y>254 THEN Y=0
150 POKE W, 15: POKE X, Y: FOR F=1 TO 10: NEXT: POKE W, 0: POKE X, 0
160 POKE M-1+30720, A: NEXT
170 PRINT "L PRESS PLAY ON TAPE": WAIT 37151, 126, 126
180 PRINT "L OK. LOADING PT. 2 L": POKE 631, 131: POKE 198, 1

```

190 END

```

200 DATA 0,0,0,0,0,0,0,0
210 DATA 0,0,0,0,3,4,9,18
220 DATA 0,0,0,224,16,200,36
230 DATA 20,20,20,36,72,200,16,224
240 DATA 20,20,20,20,18,9,4,3
250 DATA 20,62,20,20,20,20,62,20

```

```

260 DATA 0,0,0,36,255,36,255,36
270 DATA 0,0,0,0,0,0,0
280 DATA 248,252,226,255,255,36,255,36
290 DATA 31,63,71,255,255,36,255,36
300 DATA 24,60,126,102,102,126,126,126
310 DATA 126,126,126,102,102,126,60,24

```

```

320 PRINT "L" TAB(68) "YOU MUST CONTROL THE POINTS SO THAT THE ENGINES DO NOT RUN
OUT";
330 PRINT "OF TRACK.": PRINT: PRINT " YOU START WITH ONE ENGINE AND POINTS ARE AWARDED ";
340 PRINT "FOR THE TIME YOU CAN KEEP THE LOCOS FROM CRASHING."

```

Fortsetzung Seite 34

**Achtung!! Solange Vorrat reicht!! Achtung!!**

**C64**

**TOP-HIT Angebot Nr. 1**

Spielprogramme  
Karate / Firelegion / Memory / Space-Age / Planet of Death /  
COM PCK1 (Kass.) 35,-- DM  
COM PCD1 (Disk.) 40,-- DM

**TOP-HIT Angebot Nr. 2**

Spielprogramme  
Shaolin / Vampir / The Maze Graze Game / Tank-Duell / Super-Breakout /  
COM PCK2 (Kass.) 35,-- DM  
COM PCD2 (Disk.) 40,-- DM

**TOP-HIT Angebot Nr. 3 (nur Disketten)**

Anwendungsprogramme  
Sequencer / Supermoni 64 / Micro-Tools / Disk 1.0 / IML  
Bestell-Nr.: COM PCD 3 40,-- DM

**C-64-Bonus-Kassette (Top 5)**

Firebird / Projekt / Monster Attack / High Noon / Roadpainter  
Bestell-Nr.: CPK 1/85 DM 35,-

**Super-Disk II-C 64 (Top 5)**

Firebird / Projekt / Monster Attack / High Noon / Roadpainter  
Bestell-Nr.: CPD 2/85 DM 40,-

**Super-Disk-C 64 (Top 10)**

High Music / Grand Prix / Brieftaube / Cadelon / Jet-Pac / Star Battle / Painter / Skeet / Spiders / Space Comets  
Bestell-Nr.: CPD 1/85 DM 75,-

**Nachdem unsere 10 TOP-HIT-Angebote bei Ihnen so hervorragenden Zuspruch gefunden haben, bieten wir Ihnen erneut die Möglichkeit, die Restbestände dieser Aktion zu erwerben.**

**Jeder der noch nicht von unserem Angebot Gebrauch gemacht hat, bekommt also eine zweite (Bestell-) Chance. Dieses Angebot gilt nur noch solange der Vorrat reicht.**

**VC-20**

**TOP-HIT Angebot Nr. 4**

Spielprogramme  
Flowers I / Flowers II / Geo-Zaxxon / Balloon-Shoot / Stuntman  
Bestell-Nr.: COM PVK1 30,-- DM

**TOP-HIT Angebot Nr. 5**

Spielprogramme  
Einbrecher / Jupiter Rescue / Hospital / Dangerous Planet / Dino-Eggs  
Bestell-Nr.: COM PVK2 30,-- DM

**VC-20-Bonus-Kassette (Top 10)**

Düsi / Race ON / Cagy / Matron / Obst / Derrasende Malocher / Powerpack / Buffalo Bill / Prost / Fressman  
Bestell-Nr.: VPK 1/85 DM 40,-

**COMMODORE 16**

**TOP-HIT Angebot Nr. 6**

Spielprogramme  
SOS-Schiffsbruch / Schatztaucher / Car Race / Data-Maker / MERGE-Routine  
Bestell-Nr.: COM POK1 (Kass.) 30,-- DM  
Bestell-Nr.: COM POC1 (Disk.) 35,-- DM

**TOP-HIT Angebot Nr. 7**

Spielprogramme  
Shaolin / Jumping Joe / City-Runner / OLD-Routine / Zeichenformer  
Bestell-Nr.: COM POK2 (Kass.) 30,-- DM  
Bestell-Nr.: COM POC2 (Disk.) 35,-- DM

**Schneider**

**TOP-HIT Angebot Nr. 8**

Spielprogramme  
Q-Bert / Tower / Inka-Sogra\* / Duell / Alien-Hunter  
Bestell-Nr.: COM PSK1 (Kass.) 30,-- DM  
Bestell-Nr.: COM PSD1 (Disk.) 45,-- DM  
\* nur auf Kassette lauffähig

**TOP-HIT Angebot Nr. 9**

Spielprogramme  
Cave-Runner / 3 D-Labyrinth / Citylander / Etis / Demon Attack  
Bestell-Nr.: COM PSK2 (Kass.) 30,-- DM  
Bestell-Nr.: COM PSD2 (Disk.) 45,-- DM

**TOP-HIT Angebot Nr. 10**

Anwenderprogramme  
Super-File / Data-Wandler / Datenverwaltung / Tastaturbelegung / REM-Killer / \*Single-Disk-Backup / \*Discshow  
Bestell-Nr.: COM PSK3 (Kass.) 40,-- DM  
Bestell-Nr.: COM PSD3 (Disk.) 55,-- DM  
\* nur auf Diskette lauffähig

**Achtung!! Solange Vorrat reicht!! Achtung!!**

Der Tronic-Verlag bietet allen „Compute mit“ Lesern auch weiterhin die Möglichkeit, alle veröffentlichten Programme des Jahrgangs 1 zu bestellen. Geben Sie nur die Bestell-Nummer an und Sie erhalten

innerhalb  
**1 Woche**  
die von Ihnen gewünschte Software zugesandt. Beachten Sie bitte, daß bei Bestel-

lung per Telefon zusätzliche Kosten entstehen (Nachnahmegebühr bis zu 5,-- DM). Auslandslieferungen erfolgen nur gegen Vorkasse, um die Empfänger vor zu hohen Postgebühren zu schützen.

**Heft 5/85**

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

**Heft 6/85**

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

**Heft 7/85**

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

## Heft 8/85

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

## Heft 11/85

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

## Heft 9/85

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

## Heft 12/85

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

## Heft 10/85

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

## Aus diesem Heft:

<b>Commodore</b>		
Bestell-Nr. COM CK 1/2	16,-	DM
Bestell-Nr. COM CD 1/2	20,-	DM
<b>Schneider</b>		
Bestell-Nr. COM SK 1/2	16,-	DM
Bestell-Nr. COM SD 1/2	39,-	DM

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

**C64**

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

**VIC-20**

Bestellschein/Glücky	8,50 DM	15,- DM	V 61
----------------------	---------	---------	------

Multigraph/All Rammer	11,- DM	15,50 DM	V 71
Zyklo/Meteorit	11,- DM	19,50 DM	V 81
Garten/Schloß Gruselstein	14,- DM	19,50 DM	V 91
Fressman/Outlaw	14,- DM	19,50 DM	V 101
Prost/Buffalo Bill	14,- DM	19,50 DM	V 121
Joy Man/Powerpack	14,- DM	19,50 DM	V 22
Der rasende Malocher/ Frankie goes to Pharao	14,- DM	19,50 DM	V 32
Matron/Obst	14,- DM	19,50 DM	V 42
Race On/Cagy	14,- DM	19,50 DM	V 52
Nürnberg 3D/Düsi	14,- DM	19,50 DM	V 62
Breaker/Expulsion	14,- DM	19,50 DM	V 72
Zyklo/Geldsammer	11,50 DM	-	HC/V-1
Star-Wars/Punktefresser	14,50 DM	-	HC/V-2
Catch the Fish/Mister Jump	14,50 DM	-	HC/V-3
Ghost's Eggs/Fox Hunt	14,50 DM	-	HC/V-4
Collector/Break Out	14,50 DM	-	HC/V-5
Q-Bert/Salvage Crew	16,50 DM	-	HC/V-6
Dog Fight/Schwitzbad	16,50 DM	-	HC/V-1-4
Sprites/Space-Battle	13,- DM	19,- DM	UV 2/85
Grafik-Painter	10,- DM	-	UV 5/85
VIC-Block	9,- DM	-	UV 5/85
Decelerator	9,- DM	-	UV 7/85
Joypaint	12,- DM	-	UV 8/85
Disassembler, Old-Programme	10,- DM	-	9/85
Directory/Farb-JRQ	10,- DM	-	UV 10/85

**Schneider**

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

# programme

```
350 PRINT:PRINT" IF YOU REACH 1000 POINTS YOU HAVE TWO ENGINES TO KEEP GOIN
G."
360 PRINT"■■■■■PRESS ANY KEY":POKE198,0:WAIT 198,1
370 PRINT"■"TAB(67)"REACH 1000 POINTS AT THIS LEVEL AND YOU GOTO LEVEL 3 WIT
H ";
380 PRINT" THREE LOCOS."
390 PRINT:PRINT" IF YOU CRASH ON THE SECOND OR THIRD LEVELS YOUR SCORE IS ADDED
";
400 PRINT" TO YOUR PREVIOUS LEVEL'S SCORE."
410 PRINT:PRINT"■■ USE KEYS 1 - 6 TO CONTROL THE POINTS."
420 PRINT"■■■■■PRESS ANY KEY":POKE198,0:WAIT198,1:RETURN
```

## Teil: 2

```
1 GOTO54
2 FORI=1TO40:U=U+1:IFU>8THENGOTO71
3 IFI=WAND0=. THENPOKEZ,K:GOSUB31:O=I
4 IFOANDI-O>6THENPRINTJ$:GOSUB33
5 POKEY,X:FORH=1TOO-U
6 NEXT:PRINTF$:POKEY,.
7 FORS=1TOF:M=M(S):U(S)=M:P=S*2
8 GETZ$:IFZ$<"1"ORZ$>"6"THEN10
9 POKEZ,N:G=ASC(Z$)-48:GOSUB34
10 ONN(S)GOTO11,15,19,23
11 M=M+1:T=PEEK(M):L=8:IFT=6THEN26
12 IFT=2THENN(S)=3
13 IFT=3THENN(S)=4
14 GOTO26
15 M=M-1:T=PEEK(M):L=9:IFT=6THEN26
16 IFT=4THENN(S)=4
17 IFT=1THENN(S)=3
18 GOTO26
19 M=M+J:T=PEEK(M):L=11:IFT=5THEN26
20 IFT=3THENN(S)=2
21 IFT=4THENN(S)=1
22 GOTO26
23 M=M-J:T=PEEK(M):L=10:IFT=5THEN26
24 IFT=2THENN(S)=2
25 IFT=1THENN(S)=1
26 T(V)=T:IFT>6THEN74
27 V=V+1:IFV=QTHENV=.
28 POKEM,L:POKEM+C,P:POKEU(S),T(V)
:POKEU(S)+C,1
29 M(S)=M
30 NEXT:NEXT
31 W=INT(RND(O)*6+14):E=W-13
32 PRINTF$"■■ POINT "E" CHANGING":RETURN
33 G=E:I=.:O=.
34 A=C(G,3)
35 IFPEEK(A-1)>7ORPEEK(A+1)>7ORPEEK(A+J)>7ORPEEK(A-J)>7ORPEEK(A)>7THENPOKEZ,.:RE
TURN
36 R=C(G,4):R=R+1:IFG=2ORG=4THEN38
37 IFR>3THENR=1
38 IFR>6THENR=1
39 ONG GOSUB51,43,52,43,52,53
40 PRINTLEFT$(K$,C(G,1))SPC(C(G,2))Y$F$:C(G,4)=R
41 IFOTHENPOKEZ,K:RETURN
42 POKEZ,.:RETURN
43 ONRGOTO44,45,46,47,48,49
44 Y$="G■■■GAF■■■E":RETURN
45 Y$="G■■■FFF■■■G":RETURN
46 Y$="G■■■FBG■■■E":RETURN
47 Y$="E■■■GEG■■■E":RETURN
48 Y$="E■■■GDF■■■G":RETURN
49 Y$="E■■■FCG■■■G":RETURN
50 ONRGOTO44,47,48
51 ONRGOTO44,45,46
52 ONRGOTO46,47,49
53 ONRGOTO45,48,49
54 PRINT"■":CLR:M=0:T=0:C=30720:V=0:S=0:G=0:R=0:A=0:Y=7866:U=0:Q=0:B=0:Z=36876:N
=8020
55 K=0:J=22:X=188:Y=36877:F$="■":DIMC(B,4),M(4),U(4),N(4):POKE36869,255:F=PEEK(7
678)
56 FORI=1TO6:READA,B,D:C(I,1)=A:C(I,2)=B:C(I,3)=D:NEXT:K=B+F*2:POKE36879,K:K=148
57 A1$="F":A$="A":B$="B":C$="C":D$="D":E$="E":K$="■":FORI=1TO22:K$=K$+"■":NEXT
58 PRINT"■":FORA=3TO23STEP6:FORI=1TO19:IFA=15THENA=16
59 PRINT"■"LEFT$(K$,A)SPC(I)A1$:NEXTI,A
60 FORA=4TO21:FORI=1TO20STEP9:IFI=19THENI=20
61 PRINTLEFT$(K$,A)TAB(I)E$:NEXTI,A:FORI=0TO5:PRINTLEFT$(K$,10+I)TAB(1)" ":NEXT
62 PRINTLEFT$(K$,3)TAB(1)A$SPC(18)B$LEFT$(K$,9)TAB(1)D$LEFT$(K$,22)TAB(1)D$SPC(1
8)C$
63 PRINTLEFT$(K$,16)TAB(1)A$LEFT$(K$,4)SPC(11)"■1"LEFT$(K$,10)SPC(8)"2"SPC(9)"3"
64 PRINTLEFT$(K$,17)SPC(8)"4"SPC(9)"5"LEFT$(K$,21)SPC(11)"6"FF$
65 FORI=1TO6:G=I:R=INT(RND(O)*5):C(I,4)=R
```

```

66 GOSUB 36:NEXT
67 Q=F+1:D=200:W=25
68 M(1)=7724:M(2)=8142:N(1)=1:N(2)=1:M(3)=7856:N(3)=1:B=1000:N=255
69 J$="G":FORI=1TO20:J$=J$+"G":NEXT
70 POKE36878,15:GOTO2
71 FORI=1TO1000:NEXT:POKE7678,F+1:POKE36878,0:POKE7677,PEEK(7677)+1
72 IFPEEK(7677)=3THEN78
73 GOTO54
74 FORI=1TO17STEP.3:POKE36878,I:POKE36877,249:FORP=1TO60:NEXT:NEXT:POKE7678,1
75 PRINT"U"SPC(225)"CRASHED !!!"SPC(33)"YOU SCORED ";
76 PRINTSPC(31)PEEK(7677)*1000+U" POINTS"
77 FORI=1TO5000:NEXT:POKE7677,0:GOTO54
78 PRINT"U"SPC(245)"EVERY WELL DONE":FORI=1TO5000:NEXT:POKE7677,0:POKE7678,1:GOT
054
79 DATA2,10,7734,8,10,7866,8,20,7876
80 DATA15,10,8020,15,20,8030,21,10,8152

```

## Rock'me

**Super-Sound auf dem VC-20 mit min. 8 KByte-Erweiterung**

Rock'Me ist ein kurzes, aber dennoch verblüffend effektvolles Musikprogramm für den VC-20 mit min. 8kB-Erweiterung. Mit Rock'me kann eine Sequenz von 12 Takten plus Begleitung (über die Tastatur) gespielt werden.

Sind beide Listings abgetippt, wird zunächst erst Rock'me Datas geladen und gestartet. Ein Fehler wird automatisch angezeigt. Sollten alle Datas korrekt sein, meldet sich der VC-20 mit READY. Nun NEW eingeben und danach wird Rock'me Basic geladen. Mit RUN gestartet können Sie nun musizieren.

Zunächst muß erst einmal die Huellkurve programmiert werden. Mit F1 hält nun der lange Cursor an, und unten unter dem ADSR-Feld blinkt ein Balken. Sie können nun mit den Tasten 0-9 + A-F die Lautstärke für 4 Zyklen programmieren. Ein

RETURN würde den alten Wert (beim Start = 0) übernehmen. Haben Sie die 4 Zyklen programmiert, beginnt der lange Cursor wieder zu wandern und somit ist die Huellkurvengabe beendet.

Nun wird der Rhythmus programmiert. Tippen Sie nun auf die F7-Taste. Der lange Cursor verschwindet wieder, und es erscheint ein kleiner Cursor. Sie können nun mit den Cursortasten wie im Direktmodus umherwandern. Passen Sie bitte in diesem Modus besonders auf! Denn ein versehentliches Bildschirmlöschchen durch SHIFT/CLRHOMe kann nur durch erneuten Programmstart rückgängig gemacht werden.

Den Cursor setzen Sie nun an Position '1' der Snare-Taktzeile. Um nun einen Taktpunkt zu setzen, betätigen Sie bitte SHIFT-Q. Der ausgefüllte Ball stellt einen gesetz-

ten Taktpunkt, der während der unausgefüllte Ball einen ungesetzten darstellt. Probieren Sie die einmal folgende Sequenz aus.

Durch RETURN gelangen Sie wieder automatisch in den Play-Modus. Sie sollten nun den Rhythmus hören können. Besonders interessant ist der Anschluß an eine STEREO-Anlage.

Mit den oberen Tasten Q-↑ können Sie die Begleitung spielen. Sie wirkt auf Bass1 und auf Bass2, wobei Bass2 = Bass1 + 1 Oktave höher entspricht.

Beachten Sie bitte, daß Snare und Cymbel monofon sind, d. h. nur ein Instrument ist gleichzeitig hörbar. Die Cymbel ist dabei dominant.

Nun viel Spaß beim Musizieren mit ROCK'ME.

```

2 POKE37138,255:POKE36879,42:TA=11
3 PRINT"U";
5 GOSUB3000
1000 FORSC=0TOTA
1011 POKEB447,SC
1013 SYSB704
1014 SYSB448:SYSB528
1015 SYSB592:SYSB736
1016 FORT=1T04
1017 IFPEEK(8735)=1THENPOKE36878,L(T)
1018 IFPEEK(8735)=0THENPOKE36878,0

```

```

1019 NEXT
1020 POKE37136,0:FORD=36874T036877
:FOKED,0:NEXT
1021 GOTO4000
1098 NEXT
1099 GOTO1000
3000 PRINT"U_____";
3005 PRINT" 123456789ABC"
3010 FORAS=0T04:PRINT" ";:FORIS=0T04:
PRINT"o";:NEXT:READE$:PRINT" "E$:NEXT

```



## Tips & Tricks für den VC-20

### Maschinencode-Routinen:

#### Schneller Farbwechsel in Maschinensprache

```
10 for i=0 to 9 : read a : poke 7670+i,a : next
20 data 162,255,138,141,15,144,202,208,249,96
```

Die Routine wird mit 'sys 7670' aufgerufen. Sie zeigt im Bruchteil einer Sekunde alle 255 Farbkombinationen. Sie ist so schnell, daß das menschliche Auge sie kaum wahrnimmt. Deshalb ist sie nur in einer 'for/next'-Schleife wirkungsvoll.

#### Programmschutz in Maschinensprache

```
10 for i=7650 to 7675 : read a : poke i, a : next
20 data 169,116,141,50,3,169,196,141,51,3,169,0,141,145,2
30 data 169,2,141,30,145,169,194,141,20,3,169,253,162,34
40 data 141,7,3,142,6,3,96
```

Diese Routine wird mit 'sys 7650' aufgerufen. Folgender Schutz der Programme wird dabei aktiviert:

- Saveschutz
- Umschaltung zwischen Groß/Kleinschrift abschalten
- RESTORE-Taste abschalten
- STOP-Taste abschalten
- Listschutz (bei LIST wird ein RESET ausgelöst)

### Die USSR-Funktion

Viele VC-20 Benutzer werden sich sicher schon gefragt haben, wozu eigentlich die USSR-Funktion gut ist. Sobald man sie aufruft, wird ein ILLEGAL QUANTITY ERROR angezeigt. Ein Aufruf dieser Funktion führt zum Sprung des Interpreters in die Speicherstelle Null. Dort ist der Sprungbefehl JMP untergebracht. In den Speicherstellen 1 und 2 steht die Adresse, zu der beim Aufruf hingespungen wird. Wenn man nun durch Aufruf der USSR-Funktion zu einem Maschinenunterprogramm springen will, sollte man folgendermaßen vorgehen:

```
s=(startadresse)      (RETURN)
poke 2,int(s/256)      (RETURN)
poke 1, s-peek(2)mal 256 (RETURN)
```

Gibt man nun z. B. a=usr(1) ein, wird zur Startadresse gesprungen. Die Zahl in Klammern hat keinerlei Funktion, muß jedoch angegeben werden. Der Vorteil der Funktion ist, daß man mit ihr erheblich schneller ein MC-Unterprogramm anspringen kann, als dieses mit einem SYS-Aufruf möglich gewesen wäre.

### POKE-/Liste für VC-20

<b>poke 43,x:poke44,y</b>	= Der Basicanfang wird verschoben und zwar nach x+256mal y	<b>poke 788,194</b>	= Stop gesperrt
<b>poke 55,x:poke56,y</b>	= Das Basicende wird verschoben und zwar nach x+256 mal y	<b>poke 792,2</b>	= Restore gesperrt
<b>poke 120,2</b>	= Befehlsverweigerung	<b>poke 777,1</b>	= Cursor 'HOME' nach jedem Befehl
<b>poke 198,x</b>	= reguliert den Inhalt des Tastaturbuffers	<b>poke 777,199</b>	= wieder normal
<b>printpeek(197)</b>	= enthält den codierten Wert einer gedrückten Taste (normal = 64).	<b>poke 657,128</b>	= Commodore-Taste gesperrt
<b>?peek(162)</b>	= ergibt eine zufällige Zahl zwischen 0 & 100	<b>poke 794,0:poke 795,0</b>	= Kein 'OPEN' mehr möglich
<b>poke 199,1</b>	= Wirkung wie RVS ON	<b>poke 804,0:poke 805,0</b>	= Kein 'INPUT' mehr möglich
<b>poke 199,0</b>	= Wirkung wie RVS OFF	<b>poke 810,0:poke 811,0</b>	= Kein 'GET' mehr möglich
<b>poke 211,x:poke214,y:</b>	= 'print at'; positioniertes Ausdrucken	<b>poke 818,116:poke 819,196</b>	= SAVESCHUTZ
<b>sys 58640:print „text“</b>	= Listschutz:	<b>poke 816,0:poke 817,0</b>	= LOADSCHUTZ
<b>poke 774,x</b>	x=01 listet Zeilennummern	<b>?pE(152)</b>	= Anzahl der offenen Files
	x=27 listet Unsinn	<b>?pE(153)</b>	= Eingabegerät (normal 0)
	x=26 listet normal	<b>?pE(154)</b>	= Ausgabegerät (CMD); normal 3
<b>poke 775,x</b>	= Listschutz:	<b>?pE(183)</b>	= Länge des Filenamens
	x=1 Systemabsturz	<b>poke 646,x</b>	= Zeichenfarbe regulieren
	x=200 'Error' nach der ersten Zeile	<b>?pE(788)+256malpE(789)</b>	= Hardwareinterrupt (IRQ)
	x=199 listet normal	<b>?pE(653)</b>	= Flag für Shift/C T R L
<b>poke 808,126</b>	= Stop/Restore gesperrt	<b>sys 64817</b>	= Schnellreset
		<b>sys 65499</b>	= setzt ti auf null
		<b>sys 65511</b>	= schließt alle Files
		<b>sys 58234</b>	= Sprung in die NMI-Routine
		<b>wait 653,4</b>	= wartet auf CTRL
		<b>wait 653,1</b>	= wartet auf SHIFT

# programme

„Charly der Raketenmann“ fliegt mit seinem Raketenrucksack durch die Lüfte und versucht seinen Auftrag zu erledigen. Ein Programm für den VC-20 Grundversion, das uns zeigt, daß selbst bei diesem System die Programmierung in Multicolorgraphic erfolgen kann.

## Charly der Raketenmann VC-20 Grundversion

Charly und sein Raketenrucksack erhalten den Auftrag, die Dynamitstangen einzusammeln, die ein Amokläufer auf dem Bildschirm verteilt hat. Durch Druck auf den Feuerknopf zünden Sie den Raketenrucksack und Charly kann nun zwischen den einzelnen Etagen hin- und herfliegen. Aber Charly muß darauf achten, daß er nicht mit den Luftminen kollidiert. Ist es Charly gelungen zehn Dynamitstangen einzusammeln, erreicht er den nächsten Level. In diesem Level gibt es noch mehr Luftminen und der Flug wird erheblich schwieriger. Insgesamt gibt es sechs verschiedene Levels, und man muß schon ganz schön gut und schnell reagieren, um alle Levels spielen zu können. Achten Sie bitte auch auf den begrenzten Energievorrat Ihrer Rakete.

Gesteuert wird „Charly der Raketenmann“ mit dem Joystick. Die Programme müssen nacheinander auf Kassette gespeichert werden und zurückgespult werden. Jetzt SHIFT/RUNSTOP drücken und alle Programmteile werden geladen.

**Wettlauf mit der Zeit!**

**Dufte  
Grafik**



### Teil 1:

```
1 DATA162,0,189,132,28,141,12,144,201
,0,240,2,232,96,76,216,29
2 FORT=7640T07656:READA:POKET,A:NEXT
4 FORT=7300T07319:READA:POKET,A:NEXT
5 PRINT"LCATCH ALL DYNAMIT!"
6 PRINT"QBUT BECAREFUL WITH THEAIR-MINES
30 DATA225,225,225,228,231,231,228,
231,232,235,240,235,231
31 DATA225,235,232,231,228,225,0
50 PRINT"Q":POKE198,1:POKE631,131
```

### Teil 2:

```
1 PRINT"***** CHARLY *****"
2 PRINT"QCOPYRIGHT BY K.N. SOFT"
3 PRINT"QFOR UNEXPANDED VIC-20"
4 PRINT"QWRITTEN BY KAROLJ NADJ"
5 DATA169,24,141,15,144,169,25,141,15,144
6 DATA173,197,0,201,39,240,3,76,176,29,96
10 FORT=7600T07620:READA:POKET,A:NEXT
11 PRINT"QPRESS F1 TO START"
12 SYS7600:PRINT"Q":POKE198,1:POKE631,131
```

### Teil 3:

```
20 FORI= 6144 TO 6610 :READDC:POKEI,DC:NEXTI
22 DATA0,30,32,76
24 DATA86,74,34,28
26 DATA62,34,116,76
28 DATA68,136,136
30 DATA136,60,18,34
32 DATA50,44,66,66
34 DATA252,48,76,66
36 DATA66,128,128
38 DATA72,56,60,18
40 DATA33,33,33,65
42 DATA66,252,63,16
44 DATA32,60,32,64
```

```
46 DATA64,252,28,19
48 DATA32,60,32,64
50 DATA64,192,28,34
52 DATA64,64,156,132
54 DATA136,112,33
56 DATA33,66,114,78
58 DATA132,132,132
60 DATA14,4,8,8,8
62 DATA16,16,56,3
64 DATA1,2,34,66,68
66 DATA36,28,32,35
```

```
68 DATA76,112,112
70 DATA140,132,130
72 DATA64,64,64,128
74 DATA128,128,224
76 DATA28,33,35,86
78 DATA74,66,132,132
80 DATA132,32,34,84
82 DATA76,68,136,136
84 DATA136,56,68,66
86 DATA130,132,132
88 DATA72,112,56,36
90 DATA66,66,114,140
92 DATA128,128,56
```

```
94 DATA68,66,130,132
96 DATA148,136,116
98 DATA56,36,66,114
100 DATA76,136,132
102 DATA132,28,34
104 DATA65,65,56,4
106 DATA130,126,56
108 DATA14,16,16,16
110 DATA32,32,32,32
112 DATA66,68,68,132
114 DATA136,200,56
116 DATA66,66,66,132
118 DATA132,132,120
```

```
120 DATA48,33,33,66
122 DATA66,66,132
124 DATA180,252,130
126 DATA130,68,56
128 DATA56,68,130
130 DATA130,68,68
132 DATA40,16,16,32
134 DATA32,32,224
136 DATA28,4,4,24
138 DATA96,128,254
140 DATA0,60,32,32
142 DATA32,32,32,60
144 DATA0,110,112
```

## Teil 3:

```

146 DATA16,60,16,16
148 DATA12,0,60,4
150 DATA4,4,4,4,60
152 DATA8,8,8,8,42
154 DATA28,8,0,0,16
156 DATA32,127,32
158 DATA16,0,0,0,0
160 DATA0,0,0,0,0
162 DATA0,0,8,0,0
164 DATA8,8,8,8,0
166 DATA0,0,0,0,36
168 DATA36,36,0,36
170 DATA36,126,36
172 DATA126,36,36
174 DATA0,8,60,10
176 DATA28,40,30,8
178 DATA0,70,38,16
180 DATA8,100,98,0
182 DATA0,58,68,74
184 DATA48,72,72,48
186 DATA0,0,0,0,0
188 DATA16,8,4,0,4
190 DATA8,16,16,16,16
192 DATA8,4,0,32,16
194 DATA8,8,8,16,32
196 DATA0,8,42,28
198 DATA62,28,42,8
200 DATA0,0,8,8,62
202 DATA8,8,0,16,8
204 DATA8,0,0,0,0
206 DATA0,0,0,0,0
208 DATA126,0,0,0
210 DATA0,24,24,0
212 DATA0,0,0,0,0
214 DATA64,32,16,8
216 DATA4,2,0,126
218 DATA98,82,74,74
220 DATA70,70,126
222 DATA48,16,16,16
224 DATA16,16,16,124
226 DATA0,124,4,4
228 DATA124,64,64
230 DATA124,248,8
232 DATA8,56,56,8
234 DATA8,248,64,72
236 DATA72,72,124
238 DATA8,8,8,62,32
240 DATA56,4,4,4,4
242 DATA56,112,64
244 DATA64,64,92,68
246 DATA68,124,124
248 DATA4,4,8,28,8
250 DATA8,8,126,66
252 DATA66,126,66
254 DATA66,66,126
256 DATA124,68,68
258 DATA124,4,4,4
260 DATA4,0,0,8
262 PRINT"□□":POKE198,1:POKE631,131

```

## Teil 4:

```

1 DATA0,0,12,51,40,40,40,40
2 DATA56,56,124,40,40,40,124,124
3 DATA124,124,56,56,56,56,60,60
4 DATA124,124,56,56,56,56,120,120

5 DATA169,127,141,34,145,173,32,145,41,128,133,
251,169,255,141,34
6 DATA145,173,31,145,41,28,24,101,251,133,251,
173,31,145,41,32,133,252,96

7 DATA255,143,143,249,249,143,143,255
10 FORT=6512TO6519:READA:POKET,A:NEXT
11 FORT=6664TO6663+3*8:READA:POKET,A:NEXT
12 FORT=7168TO7202:READA:POKET,A:NEXT
14 FORT=6424TO6423+1*8:READA:POKET,A:NEXT
15 PRINT"□□":POKE198,1:POKE631,131

```

## Teil 5:

```

1 POKES5,0:POKE56,24:P=0:PKT=0:LE=1:AN=0
20 POKE36869,254:POKE36878,20:POKE36879,8
21 PRINT"␣PRESS F1 TO START"
22 POKE7601,8:POKE7606,9:SYS7600
26 A=7680+22*Y+X:PRINT"␣"
27 X=19:Y=13:Q=67:EE=2000:
100 GOSUB1000
101 A=7680+22*Y+X:B1=PEEK(A):B2=PEEK(A+22)
102 POKEA,65:POKEA+22,Q
103 SYS7168
104 IFPEEK(251)=140THENPOKEA,32:POKEA+22,32:X=X-1:Q=67
105 IFPEEK(251)=28THENPOKEA,32:POKEA+22,32:X=X+1:Q=66
106 IFPEEK(252)=0THENGOTO400
107 IFPEEK(A+44)=32ORPEEK(A+44)=42ANDSP=0THENPOKEA,32:Y=Y+1
108 SYS7642
200 PRINT"#####PUNKTE ";PKT
201 PRINT"#####ENERGIE ";EE;" "
202 PRINT"#####LEVEL ";LE
205 IFB1=42ORB2=42THEN2000
300 IFPEEK(A-1)=219THENX=X+1

```

```

305 IFB1=46ORB2=46THENPKT=PKT+1:AN=AN+1
306 IFPEEK(A-1)<>32ORPEEK(A+1)<>32ANDPEEK(A-1)<>46THEN999

```

```

308 IFPEEK(A)=46ORPEEK(A+22)=46THENPKT=PKT+1
400 IFPEEK(A-22)<>32ANDPEEK(A-22)<>46ANDPEEK(A-22)<>42THEN999
401 IFPEEK(252)=0THENSF=1
402 IFSP=1THENPOKEA+22,32:V=V+1:Y=Y-1:IFV=3THENV=0:POKEA,32:V=V+1:IFV=3THENY=Y+1
:SP=0:V=
410 IFEE<=0THEN2000
411 SYS7642
500 IFSP=1THENE=EE-10
501 IFB1=42ORB2=42THENGOTO2000
700 IFAN>=10THENLE=LE+1:P=0:AN=0:GOTO100
996 IFPEEK(A+44)<>32THENPOKE36877,0
997 IFSP=1THENPOKE36877,200
998 SP=0
999 GOTO100

1000 PRINT"#####";
1001 PRINT"#####";
1002 PRINT"#####";
1003 PRINT"#####";
1004 PRINT"#####";
1005 PRINT"#####";
1006 PRINT"#####";
1007 PRINT"#####";
1008 PRINT"#####";

```

```

1009 PRINT"#####";
1010 PRINT"#####";
1011 PRINT"#####";
1012 PRINT"#####";
1013 PRINT"#####";
1014 PRINT"#####";
1015 PRINT"#####";
1016 POKE36878,20:POKE646,10:IFP=1THEN1026
1017 FORE=1TO20
1018 S=INT(RND(1)*15)+1:T=INT(RND(1)*15)+6
1019 IFPEEK(7680+22*S+T)<>32THEN1018
1020 POKE7680+22*S+T,46:POKE7680+22*S+T+30720,10:NEXT
1021 POKE646,15
1022 FORE=1TO10+LE*2
1023 S=INT(RND(1)*15)+1:T=INT(RND(1)*15)+1
1024 IFPEEK(7680+S*22+T)=35THEN1023
1025 POKE7680+S*22+T,42:POKE7680+S*22+T+30720,7:NEXT
1026 P=1:RETURN
2000 PRINT"GAME OVER ":POKE36877,0
2001 PRINT"ENERGIE ";EE
2002 PRINT"TOTAL SCORE ";(2000-EE)*PKT
2003 POKE7601,8:POKE7606,15:POKE36878,0:SYS7640:
SYS7600:GOTO1
    
```

## Werkstatt:

# Interrupt-Programmierung

Der VC-20 besitzt wie viele andere Computer auch die Möglichkeit, Interrupts zu steuern. Zuerst einmal zur Bedeutung dieses Wortes: Interrupt kommt aus dem Englischen und bedeutet soviel wie 'unterbrechen'. Der Computer 'unterbricht' auch ca. 60 mal in der Sekunde seine Tätigkeit und schreibt z. B. die Systemuhr fort, läßt den Cursor blinken usw.. Wenn wir nun den Computer dazu bringen wollen, zu einer eigenen Interrupt-Routine zu springen, muß man

1. den IRQ-Interrupt abschalten;
2. in die Speicherstellen 788,789 Low- und High-Byte schreiben (Startadresse des MC-Programmes);
3. das Maschinenprogramm schreiben
4. Interrupt (IRQ) wieder anschalten;

In der Praxis sieht das folgendermaßen aus:

```

● 10 for i=0 to 12: read a : poke 7168+i,a : next : rem Lader
● 20 for i=0 to 3: read a : poke 7424+i,a : next : rem Prg.
● 30 poke 7428,peek(788):poke7429,peek(789)
● 40 data 120,169,29,141,21,3,169,0,141,20,3,88,96,206,15,144,76
● 50 sys 7168
    
```

Die Hintergrund- und Rahmenfarben wechseln ständig. Dabei kann man Programme schreiben,

1. IRQ abschalten mit MC-Befehl 'SEI'

2. Von Maschinensprache aus Lowbyte und Highbyte in 788, 789 einpoken

3. Interrupt anschalten mit 'CLI'

Damit wäre das 'Ladeprogramm' erzeugt. Aber halt, bloß noch nicht starten! Der Computer wird sofort nach dem Aufruf mit SYS 'Startadresse' dasselbe wie bei Run/Stop-Restore ausführen, da unser Maschinenprogramm noch nicht eingepoket ist. Dies wird jetzt nachgeholt. Aber nun dürfen Sie natürlich nicht den MC-Befehl RTS ausführen lassen, denn damit würden Sie alle anderen Interrupts unterbinden und der Computer würde aussteigen. Also nehmen Sie statt RTS erst einmal den Sprungbefehl 'JMP' dezimal, also 76. Nun schreiben Sie in die Speicherstellen nach diesem Befehl Low- und Highbyte der normalen IRQ-Routine.

Diese wird ermittelt durch: ?pE(788) & pE(789).

Also poken Sie gleich: poke low, peek(788);-poke high,peek(789).

Wenn Sie nun das Ladeprogramm starten, werden Sie sehen: Es läuft!

Aber die Routine darf nur recht kurz sein, da sonst die rechnerinterne Uhr verlangsamt werden würde. Dies könnte einen Absturz des Computers zur Folge haben! Also teilt man die Routine am besten in kleine Einzelstücke.

Jetzt noch ein Praxisbeispiel: Die Farben des Bildschirms wechseln ständig.

### In eigener Sache:

Wenn auch Sie Ihre selbstentwickelte Software unseren Lesern vorstellen möchten, senden Sie uns

- einen Datenträger (Kassette/Diskette)
- ein Listing (nicht unbedingt erforderlich)
- eine Programmbeschreibung
- einen ausreichend frankierten Rückumschlag

editieren usw.. Jetzt erkennen Sie sicher die nahezu unbegrenzten Möglichkeiten der Interruptprogrammierung.

# 6502-Assembler-Kurs

## Teil 9

In unserer letzten Ausgabe haben wir die Funktionsweise eines Assemblers kennengelernt. Jetzt wollen wir den Aufbau eines Assemblerlistings durch das nachfolgende Beispiel verdeutlichen.

### Unser erstes Assembler-Programm:

Das nachfolgende Listing demonstriert die Arbeitsweise des CMP-Befehls:

#### Beschreibung der einzelnen Zeilen:

10	<i>Start des Assemblers</i>	10 SYS36864 ; AUF RUF PROF1-ASSEMBLER
		20 ;
20 - 40	<i>Dokumentation (gekennzeichnet mit „;“)</i>	30 ; **** BEISPIEL FUER CMP-BEFEHL ****
		40 ;
50	<i>Befehl zur Bildschirmausgabe und Objektcode an Original-Adresse</i>	50 .OPT P,00
		60 ;
60	<i>Dokumentation</i>	70 *=49152
		80 ;
70	<i>Definierung der Startadresse</i>	100 START LDA #20
		110 CMP #21
80	<i>Dokumentation</i>	120 BCS PLUS
		130 LDA #0 ; ZEICHEN '@' AUF DEM SCHIRM AUSGEBEN
100-190	<i>Assembler-Listing mit Dokumentation</i>	140 STA 1024
		160 RTS
1000	<i>Kennzeichnung „Ende des Assembler-Listings“</i>	170 FLUS LDA #1 ; ZEICHEN 'A' AUF DEM SCHIRM AUSGEBEN
		180 STA 1024
1110	<i>Basic-Zeile; Bildschirm löschen und MC-Programm aufrufen.</i>	190 RTS
		1000 .END
		1010 PRINT"Q":SYS 49152

Ändert man die Zeile 50 wie folgt, wird das Listing auf einem Drucker ausgegeben:

50 .OPT P1,00

Ausdruck auf einem Printer:	;	**** BEISPIEL FUER CMP-BEFEHL ****
	;	
50:	C000	.OPT P1,00
	;	
70:	C000	*= 49152
	;	
100:	C000 A9 14	START LDA #20
110:	C002 C9 15	CMP #21
120:	C004 B0 06	BCS PLUS
130:	C006 A9 00	LDA #0 ; ZEICHEN '@' AUF DEM SCHIRM AUSGEBEN
140:	C008 BD 00 04	STA 1024
160:	C00B 60	RTS
170:	C00C A9 01	PLUS LDA #1 ; ZEICHEN 'A' AUF DEM SCHIRM AUSGEBEN
180:	C00E BD 00 04	STA 1024
190:	C011 60	RTS

## Funktion des Programmes:

Der Akkumulator wird mit dem Wert 20 geladen und mit dem Wert 21 verglichen. Wird durch den Vergleich das Carry-Flag gesetzt, springt der Computer in Zeile 170, wo der Akkumulator mit dem Wert 1 (Bildschirmcode für A) geladen wird. Wird das Carry-Flag gelöscht, wird der Akkumulator mit 0 (Bildschirmcode für Schnecke) geladen. Anschließend legt der Rechner den Akkumulatorinhalt in Adresse 1024 ab. Da ab Adresse 1024 der Bildschirmspeicher beginnt, wird das Symbol in der linken oberen Ecke sichtbar.

An dieser Stelle werden wir die alphabetische Auflistung aller Befehle weiterführen und anhand von kleinen Assemblerprogrammen demonstrieren. Um nicht jedesmal die Assembler-Kopfzeilen wiederholen zu müssen, beginnen wir unsere Beispiel-Listings immer ab Zeile 100. Die Zeilen 10 bis 80 müssen aus unserem oben abgedruckten Beispiel-Listing entnommen werden.

### INX Increment Memory by One

(Inkrementiere Index Register X um 1)

Dieser Befehl besitzt die gleichen Eigenschaften wie der Befehl INC, mit der Ausnahme, daß hier nur das X-Register beeinflusst wird. Oft wird dieser Befehl zum Erhöhen eines Schleifenzählers benutzt.

HEX	Adressierung	Bytes	Syntax
C8	Impliziert	1	DEX

7	6	5	4	3	2	1	0
N	V	B	D	I	Z	C	
X							X

### Beispiel als Assemblerlisting:

100 ;Zeichensatz des Commodore ausgeben  
 110 ;  
 120 VIDEO = 1024 ;Anfangsadresse des  
 Bildschirmspeichers  
 130 ;  
 140 START LDX #0 ;Zähler und Zeichencode  
 150 LOOP TXA ; Zähler als Zeichen an Akku

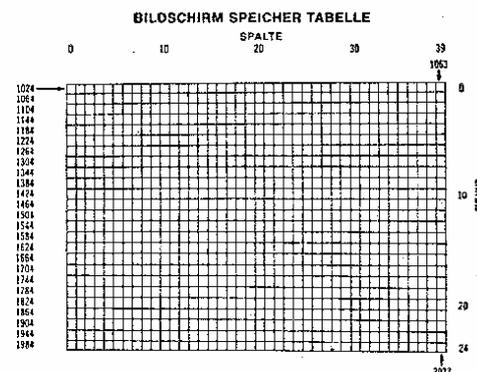
160 STA VIDEO,X ;Zeichen in Bildschirmspeicher  
 170 INX ;Zähler um 1 erhöhen  
 180 BNE LOOP ;nächste Zeichen  
 190 RTS ;Ruecksprung nach Basic  
 200 .END  
 210 SYS 49152

Das oben aufgeführte Beispiel gibt alle 255 Zeichen des Commodore-Zeichensatzes auf dem Bildschirm aus. Der Befehl INX wird für zwei Funktionen verwendet:

1. als Zähler von 0 bis 255,
2. als Bildschirmcode

Um den Ablauf des Programmes genauer zu verstehen, muß man die Speicherbelegung des Bildschirms kennen. Wir haben Ihnen diese in einer Skizze grafisch dargestellt.

sh. nebenstehende Bildschirm-Speicher-Tabelle



In der Tabelle ist sehr leicht die Aufteilung der Speicherstellen in Spalten und Zeilen zu erkennen. Um beispielsweise in Zeile 4 und Spalte 4 den Buchstaben „A“ zu bringen, speichert man einfach den Bild-

schirmcode für A in die Speicherstelle 1147. Nun, das hört sich einfach an, aber wie ist der Bildschirmcode für das ASCII-Zeichen „A“?

Die nachfolgende Tabelle gibt uns auch hierüber Auskunft:

sh. Bildschirm-Code-Tabelle auf der nächsten Seite

Wir werden uns in einem der nächsten Teile noch näher mit dem Bildschirm beschäftigen.

### INY Increment Index Y by One

(Inkrementiere Index Register Y um 1)

Der Befehl INY entspricht dem INX Befehl mit der Ausnahme, daß hier das Y-Register beeinflusst wird.

HEX	Adressierung	Bytes	Syntax
C8	Impliziert	1	INY

7	6	5	4	3	2	1	0
N	V	B	D	I	Z	C	
X							X

## BILDSCHIRM CODES

Satz 1	Satz 2	Poke	Satz 1	Satz 2	Poke												
@		0	V	v	22	,		44			64		V	86			108
A	a	1	W	w	23	-		45		A	65		W	87			109
B	b	2	X	x	24	.		46		B	66		X	88			110
C	c	3	Y	y	25	/		47		C	67		Y	89			111
D	d	4	Z	z	26	0		48		D	68		Z	90			112
E	e	5	[		27	1		49		E	69			91			113
F	f	6	£		28	2		50		F	70			92			114
G	g	7	]		29	3		51		G	71			93			115
H	h	8	↑		30	4		52		H	72			94			116
I	i	9	←		31	5		53		I	73			95			117
J	j	10	SPACE		32	6		54		J	74	SPACE		96			118
K	k	11	!		33	7		55		K	75			97			119
L	l	12	"		34	8		56		L	76			98			120
M	m	13	#		35	9		57		M	77			99			121
N	n	14	\$		36	:		58		N	78			100		<input checked="" type="checkbox"/>	122
O	o	15	%		37	;		59		O	79			101			123
P	p	16	&		38	<		60		P	80			102			124
Q	q	17	'		39	=		61		Q	81			103			125
R	r	18	(		40	>		62		R	82			104			126
S	s	19	)		41	?		63		S	83			105			127
T	t	20	.		42					T	84			106			
U	u	21	+		43					U	85			107			

Die Codes 128-255 ergeben die invers dargestellten Zeichen der Codes 0-127.

An den freien Stellen in Spalte 2 stimmen die beiden Sätze überein.

### Literatur:

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

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

## CHECKSUMMER 1.0 CPC

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

### Wie funktioniert der Checksummer?

Tippen Sie zunächst das Programm ab, wobei sie darauf achten müssen, daß alle

Zwischenräume mit dem Listing übereinstimmen. Auch REM-Zeilen müssen original übernommen werden.

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

Nach dem Speichern wird das Programm mit „RUN 65200“ aktiviert. Es erscheint die Meldung: CHECKSUMMER AKTIV FUNKTIONSTASTE 0. Betätigen Sie nun die Taste 0, wird der Checksummer aufge-

rufen. Befindet sich kein zweites zu prüfendes Programm im Speicher, wird ein Selbsttest durchgeführt. Ertönt am Ende dieses Selbsttestes ein Signal, ist das Programm noch nicht fehlerfrei. Sie sollten in diesem Fall, die auf dem Bildschirm ausgegebenen Checksummen mit denen in Ihrer Zeitschrift vergleichen und korrigieren. Arbeitet der Checksummer überhaupt nicht, ist der Fehler so gravierend, daß Sie das Programm neu eingeben

# programme

müssen.

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

Wie benutzt man den Checksummer?

Wollen Sie ein Programm mit CHECK überprüfen, laden Sie zunächst das zu

prüfende Programm. Dann geben Sie den Befehl MERGE"CHECKV1" ein, um das Prüfprogramm anzuhängen. Wichtig ist, daß das zu prüfende Programm keine Zeilennummer besitzt, die größer als 65199 ist. Jetzt aktivieren Sie den Checksummer mit RUN 65200 und Sie können jederzeit die Prüfung mit der Funktionstaste 0 überprüfen. Vergleichen Sie nun die im Heft abgedruckten Prüfsummen mit denen, die Ihr Programm ausgibt.

Falls Sie einen Fehler feststellen, müssen Sie nicht das Programm neu laden, sondern können ihn normal berichtigen und

dann wieder mit 0 das Check-Programm starten. Möchten Sie den Checksummer löschen, geben Sie einfach den Befehl DELETE 65200- ein.

Anmerkung zu Checksummer 1.0 CPC:

Groß-Kleinschreibung beachten! Leerzeichen innerhalb einer Zeile müssen dem ORIGINAL-Listing entsprechen! Leerzeichen am Ende einer Zeile werden ignoriert!

Umgewandelte TOOKEN werden automatisch in den Ursprungswert zurückgesetzt! REM-Zeilen müssen eingegeben werden!

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

**Bestellnummer:**  
**CVS 10K Kassette 10 DM**  
**CVS 10D Diskette 20 DM**

```
65200 '*** Tronic CPC Checksummer ***
65210 '*** Autor Frank Brall 1985 ***
65220 '          VERSION 1.0
65230 KEY 12B,"65530 REM TRONIC"+CHR$(13
)+ "run 65250"+CHR$(13)
65240 PRINT:PRINT"CHECKSUMMER AKTIV FUN
KTIONSTASTE 0":PRINT:END
65250 CLS:PRINT"Tronic-Verlag Checksumme
r Version 1.0"
65260 PRINT:PRINT:PRINT:PRINT"<B>ILDSCHI
RM <D>RUCKER"
65270 E$=INKEY$:E$=LOWER$(E$):IF E$="" T
HEN 65270
65280 IF E$="b" THEN a=0:GOTO 65310
65290 IF E$="d" THEN a=8:GOTO 65310
65300 GOTO 65200
65310 ADRESS=368:GCHECK=0:CLS:PRINT#A,"C
HECK V1,PRUEFSUMMEN:":PRINT#A:PRINT#A,"Z
EILENNR.: SUMMEN"
65320 LAENGE=(PEEK(ADRESS+1)*256)+PEEK(A
DRESS)
65330 ZNUM=(PEEK(ADRESS+3)*256)+PEEK(ADR
ESS+2)
65340 IF ZNUM=65200 THEN IF F=0 THEN 654
80 ELSE PRINT #A, USING "#####";ZNUM1;:P
RINT#A,"-";:PRINT#A,":":PRINT #A,
USING "#####";ZCHECK:F=0:GOTO 65480
65350 IF ZNUM=65500 THEN 65510
65360 ZCHECK=0:FOR I=0 TO LAENGE-6
```

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

## CHECKSUMMER SELBSTTEST !

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

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

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

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

## DEFENDER

für Schneider CPC 464

### KHARR – Invasoren bedrohen die Erde

Sie sind der letzte Kampfpilot einer glorreichen und heldenhaften Raumflotte, die keinen Gegner fürchten mußte. Dank eines Omega-Lasers, die z. Zt. stärkste Waffe innerhalb des Universums, waren und sind Sie Ihren Gegnern kämpferisch überlegen. Leider hat diese Wunderwaffe aber einen erheblichen Nachteil. Nach jeweils vier Schüssen müssen Sie eine kurze Feuerpause einlegen, um die Kanone nicht zu zerstören. Achten Sie beim Abschub der feindlichen Invasoren also immer auf die Anzahl der abgegebenen Schüsse. Ist es Ihnen gelungen ein KHARR-ALIEN zu zerstören, verwandelt sich dieser in einen Mutanten der zu Boden stürzt. Dieser Mutant darf aber unter keinen Umständen den Boden berühren, da die Druckwelle des Aufpralls den eigenen Jäger zerstören würde. Die Steuerung der drei Jäger erfolgt mit dem Joystick.

```

1 REM *****
2 REM *   DEFENDER   *
3 REM *****
4 SYMBOL AFTER 237:SYMBOL 238,137,74,2,8
,65,4,66,145
5 MEMORY &9FFF
6 GOSUB 9000
9 REM Farben setzen
10 MODE 0:BORDER 0:INK 0,0:CLS
20 INK 1,6:INK 2,24:INK 3,11:INK 4,6,24
30 INK 5,26:INK 6,15:INK 7,18:SPEED INK
5,5
35 INK 8,25:INK 9,16:INK 10,7
40 REM Jaeger Rechts
50 DATA 0,0,0,0,12,12,12,0,64,192,0,0,0,
192,128,0
60 DATA 16,192,200,192,0,192,128,0,64,19
2,0,0,12,12,12,0
70 REM Jaeger Links
80 DATA 0,0,0,0,0,12,12,12,0,0,192,128,0
,64,192,0
90 DATA 192,196,192,32,0,64,192,0,0,0,19
2,128,0,12,12,12
100 REM Alien
110 DATA 0,0,168,0,0,84,252,0,0,252,252,
168,0,184,184,168
120 DATA 84,252,252,252,0,252,124,168,0,
80,80,0,0,160,0,160
130 REM Mutant
140 DATA 0,8,8,0,0,64,0,0,0,64,0,0,0,192
,128,0

```

```

150 DATA 0,192,128,0,0,64,0,0,0,64,0,0,0
,0,0,0
199 REM Graphik einlesen
200 RESTORE 50:a=&A030
210 FOR i=0 TO 31:READ c
220 POKE i+a,c:NEXT i
230 RESTORE 80:a=a+i
240 FOR i=0 TO 31:READ c
250 POKE i+a,c:NEXT i
270 RESTORE 110:a=a+i
280 FOR i=0 TO 31:READ c
290 POKE i+a,c:NEXT i
300 RESTORE 130:a=a+i
310 FOR i=0 TO 31:READ c
320 POKE i+a,c:NEXT i
330 a=&A000:REM M-Code Start adresse
340 b=&A00B:REM Ziel fuer Zeichenauswahl
350 c=&30:d=&50:e=&70:f=&90
360 REM c-f Werte fuer Zeichenauswahl
499 REM Spielfeld Aufbau
500 hsc=0
505 sh=0:sho=0
510 sc=0:x=11:y=12:POKE b,c:ship=3
515 zaehler=0:level=1
520 MOVE 1,390:DRAW 640,390,9
530 PEN 8:LOCATE 1,1:PRINT"SCORE":LOCATE
10,1:PRINT"HIGH":LOCATE 16,1:PRINT"SHIP
S"
540 LOCATE 1,2:PRINT sc:LOCATE 10,2:PRIN
T hsc:LOCATE 16,2:PRINT ship
550 MOVE 1,365:DRAW 640,365,9
560 MOVE 1,1:DRAW 30,10,10:DRAW 70,5,10:
DRAW 120,8,10
570 DRAW 130,2,10:DRAW 200,10,10:DRAW 27
0,2,10
580 DRAW 340,9,10:DRAW 410,1,10:DRAW 420
,6,10
590 DRAW 500,2,10:DRAW 560,10,10:DRAW 60
0,3,10
600 DRAW 640,2,10
610 LOCATE x,y:CALL a
620 SOUND 1,0,30,15,1,1,30
630 SOUND 4,10,30,15,1,1,10
635 REM Hauptprogramm
640 z=INT(RND*2)+1
650 IF z=1 THEN u=1:u1=1
660 IF z=2 THEN u=20:u1=-1

```

**Wichtig:** Nachdem das Programm abgebrochen wurde, können Sie es mit GOTO 10 neu starten. Ein Neustart mit RUN würde zur Fehlermeldung „Improppor Argument“ führen.

# programme

```
670 v=INT(RND*15)+4
680 IF v<7 THEN v1=1 ELSE v1=-1
690 POKE b,e:LOCATE u,v:CALL a
700 u2=u:v2=v
710 u=u+u1:v=v+v1
720 IF u>=20 OR u<=1 THEN LOCATE u2,v2:PRINT " ":GOTO 640
730 IF v>22 THEN v1=-1
740 IF v<5 THEN v1=1
745 CALL &BD19
750 POKE b,e:LOCATE u2,v2:PRINT " "
760 LOCATE u,v:CALL a
770 x1=x:y1=y
775 IF sh=1 THEN sho=sho-1:IF sho<=0 THEN sh=0
780 IF JOY(0) AND 4 THEN c=&50
790 IF JOY(0) AND 8 THEN c=&30
800 IF JOY(0) AND 1 THEN IF y>5 THEN y=y-2
810 IF JOY(0) AND 2 THEN IF y<22 THEN y=y+2
815 CALL &BD19
820 POKE b,c:LOCATE x1,y1:PRINT " "
830 LOCATE x,y:CALL a
835 IF x=u THEN IF y=v THEN GOTO 3500
840 IF sh=0 THEN IF JOY(0) AND 16 THEN GOTO 2000
850
860 zuf=12-level:IF INT(RND*zuf)+1=1 THEN GOTO 3000
870 IF mu=1 THEN GOTO 4000
900 GOTO 700
2000 sho=sho+1:IF sho>=4 THEN sh=1:sho=6
2005 IF c=&50 THEN GOTO 2100
2010 MOVE 355,400-(16*y)+6:DRAW 640,400-(16*y)+6,2
2020 SOUND 1,10,15,7,0,2
2025 expl=0
2030 IF y=v THEN IF x<u THEN expl=1
2035 expl1=0:IF mu=1 THEN IF y=vv THEN IF x<uu THEN expl1=1
2040 MOVE 355,400-(16*y)+6:DRAW 640,400-(16*y)+6,0
2060 IF expl=1 THEN GOTO 2500
2065 IF expl1=1 THEN GOTO 4200
2070 GOTO 850
2100 MOVE 320,400-(16*y)+6:DRAW 1,400-(16*y)+6,2
2110 SOUND 1,10,15,7,0,2
2115 expl=0
2120 IF y=v THEN IF x>u THEN expl=1
2125 expl1=0:IF mu=1 THEN IF y=vv THEN IF x>uu THEN expl1=1
2130 MOVE 320,400-(16*y)+6:DRAW 1,400-(16*y)+6,0
2140 IF expl=1 THEN GOTO 2500
2145 IF expl1=1 THEN GOTO 4200
2150 GOTO 850
2500 LOCATE u,v:PEN 12:PRINT CHR$(238)
2510 SOUND 1,5,50,7,0,1,31
2520 FOR i=1 TO 27:INK 12,i:NEXT i
2530 FOR i=27 TO 0 STEP-1:INK 12,i:NEXT i
2535 LOCATE u,v:PRINT " "
```

```
2540 IF mu=0 THEN:mu=1:vv=v:uu=u
2550 PEN 8:LOCATE 1,2:sc=sc+10:PRINT sc
2560 zaehler=zaehler+1:IF zaehler=10 THEN N zaehler=0:level=level+1:GOTO 2600
2570 GOTO 640
2600 IF level>=7 THEN level=7
2610 PEN 8:LOCATE 4,10:PRINT"LEVEL"level
2620 FOR i=1 TO 4:FOR l=1 TO 26
2630 INK 0,l:SOUND 1,22,1,7,0,1,31
2640 NEXT l,i
2650 INK 0,0:LOCATE 3,10:PRINT " "
2660 GOTO 640
3000 IF x>u THEN GOTO 3100
3010 MOVE 32*u-32,400-(16*v)+6:DRAW 1,400-(16*v)+6,1
3020 SOUND 2,100,15,7,0,2
3030 expl=0:IF y=v THEN expl=1
3040 MOVE 32*u-32,400-(16*v)+6:DRAW 1,400-(16*v)+6,0
3050 IF expl=1 THEN GOTO 3500
3060 GOTO 700
3100 MOVE 32*u,400-(16*v)+6:DRAW 640,400-(16*v)+6,1
3110 SOUND 2,100,15,7,0,2
3120 expl=0:IF y=v THEN expl=1
3130 MOVE 32*u,400-(16*v)+6:DRAW 640,400-(16*v)+6,0
3140 IF expl=1 THEN GOTO 3500
3150 GOTO 700
3500 INK 2,24,6:INK 1,6,24
3510 SOUND 1,4,60,7,0,1,31
3520 FOR i=1 TO 31:SOUND 4,0,5,7,0,2,i
3530 NEXT i
3540 INK 1,6:INK 2,24
3550 ship=ship-1:IF ship=0 THEN GOTO 1000
3560 PEN 8:LOCATE 16,2:PRINT ship
3570 GOTO 700
4000 vv1=vv:uu1=uu
4010 vv=vv+1:IF vv>23 THEN LOCATE uu1,vv1:PRINT " ":mu=0:vv=0:uu=0:GOTO 3500
4020 LOCATE uu1,vv1:PRINT " "
4030 LOCATE uu,vv:POKE b,f:CALL a
4040 GOTO 700
4200 PEN 11:LOCATE uu,vv:PRINT CHR$(238)
4210 SOUND 1,4,45,7,0,1,12:FOR i=1 TO 27:INK 11,i:NEXT i
4220 sc=sc+20:LOCATE 1,2:PEN 8:PRINT sc
4230 LOCATE uu,vv:PRINT " ":mu=0
4235 vv=0:uu=0
4240 GOTO 700
8999 REM Multi-Color M-Code
9000 a=&A000
9005 RESTORE 9040
9010 FOR i=0 TO 39:READ c
9020 POKE a+i,c
9030 NEXT i
9040 DATA &cd,&7B,&bb,&25,&2d,&cd,&1a,&b
c,&1f,&0B
9050 DATA &01,&30,&a0,&0a,&77,&03,&23,&0
a,&77,&03,&23,&0a,&77,&03,&23
9060 DATA &0a,&77,&03,&c5,&01,&fd,&07,&0
9,&c1,&15,&c2,&0d,&a0,&c9,201,201
```

```

9070 ENT 1,5,20,1,10,15,2,5,-10,3
9075 ENT 2,4,30,2,6,15,1,3,30,1
9080 :
9090 :
9100 MODE 1:BORDER 12:INK 0,1:INK 1,6
9110 LOCATE 14,1:PEN 1:PRINT"D E F E N D
E R"
9120 PRINT"Die Invasoren vom Planeten KH
ARR haben die Erde besetzt"
9130 PRINT"Du bist als letzter Kampfpilo
t der Raum-flotte unterwegs um die Invas
oren zu vertreiben."
9140 PRINT"Dein Omega-Laser ist die stae
rkste Waffedie zur Zeit im Universum ver
fuegbar ist"
9150 PRINT"Doch er hat einen Nachteil.Na
ch jeweils 4 Schuessen muss eine kurtze
Feuerpause eingelegt werden um ihn nicht
zu Zersto-ehren.Dein Auftrag ist es so
viele Invasoren wie moeglich abzusc
hiesen"
9160 PRINT"Doch Vorsicht,hast du eines d
er KHARR- ALIENS zerstoehrt so verwande
lt er sich in einen MUTANTEN der zu
Boden faellt"
9170 PRINT"Dieser MUTANT darf unter kein
en Um- staenden der Boden beruehren

```

```

da die Druckwelle sonst deinen Jaege
r vernich- tet."
9180 PRINT"Du hast drei Jaeger zur Verfu
egung"
9190 PRINT"Gesteuert wird mit dem Joysti
ck"
9200 PRINT"ALLES KLAR? Taste Druecken"
9210 CALL &BB18:CLS:RETURN
9900 RETURN
10000 PEN 8:CLS:LOCATE 5,10:PRINT"GAME O
VER"
10010 PEN 9: LOCATE 1,12:PRINT"PUNKTE="s
c
10020 IF sc>hsc THEN hsc=sc:LOCATE 1,14:
PRINT"NEUER HIGH-SCORE"
10030 SOUND 4,100,70,7,0,2,31
10040 SOUND 2,150,100,7,0,2,15
10050 SOUND 1,200,130,7,0,1,20
10060 LOCATE 1,21:PRINT"Fuer neues Spiel
10070 PRINT"Taste Druecken"
10075 CALL &BB03
10080 CALL &BB18
10085 CLS
10090 LOCATE 1,10:INPUT"NEUES SPIEL";a$
10100 IF a$="j" THEN CLS:mu=0:GOTO S10
10110 MODE 1:PEN 2:INK 0,1:CLS:END

```

CHECK VI:PRUEFSUMMEN:	330- 340: 106077	815- 820: 29318	2640- 2650: 11382	9050- 9060: 75853
ZEILENNR.: SUMMEN	350- 360: 96805	830- 835: 56941	2660- 3000: 13622	9070- 9075: 23766
1- 2: 14399	499- 500: 20173	840- 850: 39730	3010- 3020: 37075	9080- 9090: 2
3- 4: 36657	505- 510: 25588	860- 870: 51419	3030- 3040: 84625	9100- 9110: 25963
5- 6: 2653	515- 520: 36900	900- 2000: 27932	3050- 3060: 17541	9120- 9130: 72117
7- 10: 17224	530- 540: 9645	2005- 2010: 24176	3100- 3110: 32616	9140- 9150: 56401
20- 30: 32832	550- 560: 33975	2020- 2025: 8869	3120- 3130: 80167	9160- 9170: 35378
35- 40: 18878	570- 580: 38670	2030- 2035: 109670	3140- 3150: 17541	9180- 9190: 116595
50- 60: 107108	590- 600: 20940	2040- 2060: 30224	3500- 3510: 10313	9200- 9210: 41360
70- 80: 55581	610- 620: 16025	2065- 2070: 17372	3520- 3530: 27279	9900-10000: 19434
90- 100: 64412	630- 635: 18187	2100- 2110: 10294	3540- 3550: 60731	10010-10020: 70794
110- 120: 64418	640- 650: 40973	2115- 2120: 68757	3560- 3570: 15503	10030-10040: 12718
130- 140: 41266	660- 670: 49093	2125- 2130: 52812	4000- 4010: 59047	10050-10060: 35306
150- 199: 53623	680- 690: 69040	2140- 2145: 33664	4020- 4030: 47480	10070-10075: 14928
200- 210: 22808	700- 710: 63275	2150- 2500: 24607	4040- 4200: 32254	10080-10085: 1384
220- 230: 36948	720- 730: 21696	2510- 2520: 32690	4210- 4220: 35180	10090-10100: 55135
240- 250: 33190	740- 745: 16216	2530- 2535: 47382	4230- 4235: 38265	10110- : 8296
270- 280: 32164	750- 760: 38754	2540- 2550: 48647	4240- 8999: 19387	
290- 300: 37148	770- 775: 60579	2560- 2570: 22341	9000- 9005: 4340	
310- 320: 33190	780- 790: 35600	2600- 2610: 61047	9010- 9020: 24530	
	800- 810: 118815	2620- 2630: 37643	9030- 9040: 50978	
				GESAMTCHECKSUMME: 52962

**Das Programm „Boulder Dash“ ist ein Action-Spiel, bei dem Sie Ihre Spielbilder selbst entwerfen können. Der Gamedesigner im Spiel ermöglicht einen individuellen Bildschirmaufbau der einzelnen Levels. Sie bestimmen also in diesem Programm selbst das Spielgeschehen.**

## Boulder Dash für den Schneider CPC 465

Das Ziel des Spieles ist es, mit Ihrer Spielfigur alle auf dem Bildschirm verstreuten Diamanten einzusammeln. Allerdings müssen Sie darauf achten, daß Ihnen nicht bei diesem Einsammeln Felsbrocken auf den Kopf fallen oder das Sie von den

umherfliegenden Schmetterlingen berührt werden. Insgesamt steht Ihnen nur ein Leben, jedoch unbegrenzt viel Zeit für diese Aufgabe zur Verfügung.

Die vom Programm vorgegebenen Bilder rufen Sie mit „S“ auf. Den Gamedesigner

können Sie mit jeder beliebigen Taste ausser „S“ aufrufen. An dieser Stelle erklären wir Ihnen die Menue-Punkte und Ihre Funktionen:

1. Editor

Sie gelangen in ein Menue, indem man ein

# programme

Symbol auswählt. Editiert man ein neues Bild, füllt das erste Symbol das Feld, ansonsten steuert man den Zeiger mit dem Joystick. Ein Symbol wird durch drücken des Feuerknopfes gesetzt. Die Spielfigur und die Schmetterlinge können nicht überschrieben werden. Mit der ENTER-Taste kehren Sie in das Menue zurück.

## 2. Laden

Mit dieser Funktion können Sie selbstgestellte Spielfelder laden, die auch an-

schließend über den Editor editiert werden können. Für den Namen können Sie auch Wildcards verwenden.

## 3. Speichern

Dieser Menüpunkt speichert selbsteditierte Bilder ab.

## 4. Spielen

Die selbstentworfenen Spielbilder können gespielt werden.

## 5. Ende:

Sie gelangen zurück in das Titelbild und das editierte Bild wird gelöscht.

**Anmerkung:** Die im Programm vorgegebenen Bilder sind als Anregung gedacht. Es dürfen pro Bild nur neun Schmetterlinge gesetzt werden. Ein Neustart des Programms erfolgt über die Taste Null des Zehnerblocks.

### Teil 1:

```

10 REM *****
20 REM ***      BOULDER DASH      ***
30 REM ***      1985 BY          ***
40 REM ***      Ralf Moehse      ***
50 REM ***      Kirchweg 2      ***
60 REM ***      2848 Vechta-Dythe ***
70 REM ***
80 REM ***      Shaperoutinen von ***
90 REM ***      Volker Everts    ***
100 REM ***
110 REM ***      Poke fuer Modes ***
120 REM ***      von             ***
130 REM ***      Helmut Tischer  ***
140 REM *****
150 REM shape routine
160 REM es werden die befehle
170 REM pshape,@shape$,x,y,m
180 REM gshape,@shape$,x,y
190 REM und xshape,@a$,x1,y1,@shape$,x2,
y2,m zur verfuegung gestellt
200 MODE 1
210 MEMDRY &9FFF:adr=&A000:zeile=140
220 FOR i=1 TO 21
230     summe=0
240     FOR j=1 TO 16
250         READ byte$:byte=VAL("&"+byte$)
260         POKE adr,byte:summe=summe+byte:a
dr=adr+1
270     NEXT j
280     READ check:IF check<>summe THEN PR
INT"Data Fehler in Zeile";zeile:END
290 NEXT i
300 CALL &A000
310 BOTO 540
320 DATA 01,09,a0,21,39,a0,c3,d1,bc,1d,a
0,c3,3d,a0,c3,44,1880
330 DATA a0,c3,9b,a0,c3,28,a1,c3,38,a1,c
3,41,a1,47,53,48,2125
340 DATA 41,50,c5,50,53,48,41,50,c5,58,5
3,48,41,50,c5,47,1575
350 DATA 45,d4,50,55,d4,53,57,d0,00,00,0
0,00,00,01,12,7e,1181
360 DATA 16,00,18,0a,16,00,01,77,1a,fe,0
4,cc,f5,a0,fe,03,1348
370 DATA c0,21,81,a0,70,23,72,23,71,dd,6

```

```

6,02,dd,6e,00,cd,1784
380 DATA d0,a0,dd,56,05,dd,5e,04,1a,3d,c
8,d8,4f,eb,23,46,1921
390 DATA 23,66,68,46,eb,13,f3,dd,68,dd,4
5,79,90,38,1a,4f,1849
400 DATA e5,1a,ae,77,13,2c,cc,1e,a1,10,f
6,e1,7c,c6,08,67,1926
410 DATA e6,38,20,e5,cd,15,a1,18,e0,fb,c
9,fe,07,cc,f5,a0,2504
420 DATA fe,06,c0,dd,e5,dd,6e,00,dd,66,0
2,e5,47,dd,23,10,2130
430 DATA fc,dd,e5,3e,03,16,00,cd,46,a0,d
d,e1,e1,dd,75,00,2233
440 DATA dd,74,02,3e,03,cd,3d,a0,dd,e1,3
e,03,cd,44,a0,c9,1975
450 DATA 45,cb,85,4c,26,00,54,5d,29,29,1
9,29,29,29,59,19,1040
460 DATA ed,5b,c9,b1,19,7c,e6,07,67,3a,c
b,b1,84,cb,18,30,2040
470 DATA 02,c6,20,67,c9,5f,dd,7e,00,dd,2
3,dd,23,fe,05,d0,1957
480 DATA 21,10,a1,85,6f,30,01,24,7e,57,3
2,45,a0,7b,3d,c9,1416
490 DATA 00,ae,a6,b6,2f,7c,d6,40,67,7d,c
6,50,6f,d0,24,7c,1956
500 DATA e6,07,c0,7c,d6,08,67,c9,3d,c0,d
d,66,01,dd,6e,00,1987
510 DATA cd,60,bb,77,23,36,00,c9,3d,c0,d
d,7e,00,cd,5d,bb,1982
520 DATA c9,d6,02,c0,cd,2a,a1,dd,7e,02,c
d,5d,bb,c9,00,00,2052
530 REM hier werden die shapes erzeugt
540 MODE 1:RESTORE 1000:READ x,y:x=x\4
550 FOR i=0 TO 6
560 shape$(i)=CHR$(x)+STRING$(x*y,CHR$(0
)):NEXT
570 FOR sh=1 TO 6
580     FOR i=1 TO y
590         READ daten$:IF LEN(daten$)<>x*4
THEN ERROR 5
600         FOR j=1 TO x
610             hi=0:l0=0
620             FOR k=1 TO 4
630                 pixel=VAL(MID$(daten$,
(j-1)*
4+k,1)) AND 3
640                 hi=hi*2+(pixel AND 1):l0=l0*
2+pixel\2

```

```

650     NEXT k
660     MID$(shape$(sh), (i-1)*x+j+1, 1)
=CHR$(lo+16*hi)
670     !PSHAPE, @shape$(sh), 60, 30
680     NEXT j
690     NEXT i
700 NEXT sh
710 MODE 1
720 READ x, y: x=x\4
730 hand$=CHR$(x)+STRING$(x*y, CHR$(0))
740 FOR i=1 TO y
750   READ daten$: IF LEN(daten$)<>x*4 TH
EN ERROR 5
760   FOR j=1 TO x
770     hi=0: lo=0
780     FOR k=1 TO 4
790       pixel=VAL(MID$(daten$, (j-1)*4+
k, 1)) AND 3
800       hi=hi*2+(pixel AND 1): lo=lo*2+
pixel\2
810     NEXT k
820     MID$(hand$, (i-1)*x+j+1, 1)=CHR$(1
o+16*hi)
830     !PSHAPE, @hand$, 60, 30
840     NEXT j
850 NEXT i

```

```

860 PRINT "OK"
870 REM titelbild beim laden von boulder
dash 2
880 MODE 1
890 FOR a=1 TO 20 STEP 2: !PSHAPE, @shape$
(1), a*4-4, 2: !PSHAPE, @shape$(2), (a+1)*4-4
, 2: NEXT a
900 FOR a=1 TO 20 STEP 2: !PSHAPE, @shape$
(2), a*4-4, 46: !PSHAPE, @shape$(1), (a+1)*4-
4, 46: NEXT a
910 FOR a=2 TO 11 STEP 2: !PSHAPE, @shape$
(2), 0, a*4-2: !PSHAPE, @shape$(1), 0, (a+1)*4
-2: !PSHAPE, @shape$(1), 76, a*4-2: !PSHAPE, @
shape$(2), 76, (a+1)*4-2: NEXT a
920 INK 3, 20, 10
930 LOCATE 15, 20: PEN 3: PRINT "Loading..."
: PEN 1
940 LOCATE 19, 9: PEN 3: PRINT "by": PEN 1
950 POKE &B1CB, &0: POKE &B1CF, &CC: POKE &B
1D0, &33
960 LOCATE 5, 6: PRINT "BOULDER DASH"
970 POKE &B1CF, &88: POKE &B1D0, &22: PEN 15
: LOCATE 7, 14: PRINT "RamaSoft"
980 CHAIN MERGE"!BOULDER DASH 2", 10, DELE
TE

```

```

990 REM datas fuer shapes
1000 DATA 16, 16
1010 REM Erde
1020 DATA "1011111222113331"
1030 DATA "131111121111101"
1040 DATA "1113111113331111"
1050 DATA "1311011131111131"
1060 DATA "1333111333112221"
1070 DATA "1110222211333213"
1080 DATA "1122202222231101"
1090 DATA "3322233311221111"
1100 DATA "1221213110122333"
1110 DATA "2211133311112211"
1120 DATA "1113111122213331"
1130 DATA "1333122112111103"
1140 DATA "1110111113331111"
1150 DATA "111131111111122"
1160 DATA "1333122210113332"
1170 DATA "1111112133311111"
1180 REM Steine
1190 DATA "0000033311100000"
1200 DATA "0003313031111000"
1210 DATA "0033313331111100"
1220 DATA "0333033033011110"
1230 DATA "0333311333113110"
1240 DATA "3313331310333311"
1250 DATA "3331133313333331"
1260 DATA "3333130331130331"
1270 DATA "3303333333333313"
1280 DATA "3331331133103313"
1290 DATA "3331330331331303"
1300 DATA "0331313331133330"
1310 DATA "0333333313033330"
1320 DATA "00333110331133300"
1330 DATA "0003333331333000"

```

```

1340 DATA "0000033313300000"
1350 REM Mauer
1360 DATA "0000000000000000"
1370 DATA "0131313001313130"
1380 DATA "0313131003131310"
1390 DATA "0131313001313130"
1400 DATA "0313131003131310"
1410 DATA "022222002222220"
1420 DATA "022222002222220"
1430 DATA "0000000000000000"
1440 DATA "0000000000000000"
1450 DATA "3130013131300131"
1460 DATA "1310031313100313"
1470 DATA "3130013131300131"
1480 DATA "1310031313100313"
1490 DATA "2220022222200222"
1500 DATA "2220022222200222"
1510 DATA "0000000000000000"
1520 REM Diamant
1530 DATA "0000000110000000"
1540 DATA "0000011111100000"
1550 DATA "0000011111100000"
1560 DATA "0002222222222000"
1570 DATA "0002222222222000"
1580 DATA "0333333333333330"
1590 DATA "0333333333333330"
1600 DATA "0000000000000000"
1610 DATA "0000000000000000"
1620 DATA "0333333333333330"
1630 DATA "0333333333333330"
1640 DATA "0002222222222000"
1650 DATA "0002222222222000"
1660 DATA "0000011111100000"
1670 DATA "0000011111100000"
1680 DATA "0000000110000000"

```

```

1690 REM Spielfigur
1700 DATA "0000000330000000"
1710 DATA "0003333333333000"
1720 DATA "0000020220200000"
1730 DATA "0000022222000000"
1740 DATA "0000022002200000"
1750 DATA "0000000220000000"
1760 DATA "0001113113111000"
1770 DATA "0001101111011000"
1780 DATA "0001101111011000"
1790 DATA "0003301111033000"
1800 DATA "0002203333022000"
1810 DATA "0000001111000000"
1820 DATA "0000111001110000"
1830 DATA "0011100000011100"
1840 DATA "0330000000000330"
1850 DATA "2220000000000222"
1860 REM Gegner
1870 DATA "3000000000000003"
1880 DATA "1300000000000031"
1890 DATA "2130000000000312"
1900 DATA "2213000000003122"
1910 DATA "2233300000033322"
1920 DATA "2222130000312222"
1930 DATA "222213003122222"
1940 DATA "3333333113333333"
1950 DATA "3333333113333333"
1960 DATA "2222213003122222"
1970 DATA "2222130000312222"
1980 DATA "2233300000033322"
1990 DATA "2213000000003122"
2000 DATA "2130000000000312"
2010 DATA "1300000000000031"
2020 DATA "3000000000000003"
2030 REM hand

```



```

430 DATA 3,2,1,2,3,0,0,0,4,4,2,4,4,4,0,0
,3,4,4,3
440 DATA 3,2,4,4,3,0,0,4,2,4,2,2,4,4,2,0
,3,4,4,3
450 DATA 3,2,4,4,3,0,2,4,4,2,2,2,2,2,4,4
,0,4,2,3
460 DATA 3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
,3,3,3,3
470 REM Level 4
480 DATA 3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
,3,3,3,3
490 DATA 3,0,0,0,0,0,2,0,0,0,3,2,1,4,1,2
,4,1,2,3
500 DATA 3,0,0,0,0,4,4,0,0,0,3,4,1,1,1,1
,2,1,2,3
510 DATA 3,0,0,0,4,2,4,2,0,0,3,2,1,1,4,1
,2,4,2,3
520 DATA 3,0,2,4,4,2,4,4,2,2,3,1,1,2,1,4
,1,1,1,3
530 DATA 3,0,4,4,4,4,4,4,1,1,1,1,1,1,1,1
,2,4,4,3
540 DATA 3,3,3,3,3,3,3,3,3,4,3,3,3,3,3,3
,3,3,3,3
550 DATA 3,2,4,2,4,1,1,2,1,1,1,1,4,4,4,4
,4,4,0,3
560 DATA 3,2,1,2,1,2,4,1,1,2,3,0,2,2,2,2
,2,0,0,3
570 DATA 3,4,4,2,5,2,2,4,1,2,3,0,0,4,4,4
,4,4,2,3
580 DATA 3,2,1,4,1,4,1,1,1,4,3,0,0,0,2,0
,0,0,0,3
590 DATA 3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
,3,3,3,3
600 REM Level 5
610 DATA 3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
,3,3,3,3
620 DATA 3,2,1,1,2,1,1,1,2,0,2,4,1,1,1,1
,2,1,2,3
630 DATA 3,1,4,1,1,1,1,4,2,0,4,2,1,1,4,6
,2,1,1,3
640 DATA 3,1,2,1,1,2,1,1,1,0,1,4,1,2,1,1
,2,1,4,3
650 DATA 3,1,1,1,1,1,1,1,1,0,2,1,1,2,1,1
,2,1,1,3
660 DATA 3,1,2,1,6,4,1,1,1,0,2,1,1,4,1,1
,2,4,1,3
670 DATA 3,1,1,1,1,1,1,4,1,0,2,1,1,2,1,2
,2,1,2,3
680 DATA 3,2,4,1,1,1,1,1,5,0,1,1,4,1,1,1
,1,2,1,3
690 DATA 3,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0
,1,4,1,3
700 DATA 3,2,0,0,1,1,1,1,1,4,1,1,1,1,1,0
,0,1,1,3
710 DATA 3,0,0,1,1,1,4,1,1,6,1,1,4,1,1,1
,0,0,0,3
720 DATA 3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
,3,3,3,3
730 REM anfang spiel
740 level=1
750 live=3
760 hoelevel=5

```

```

770 ON level GOSUB 1610,1620,1630,1640,1
650
780 MODE 1
790 SOUND 1,400,500,4,,2
800 FOR y2=1 TO 12
810   FOR x2=1 TO 20
820     READ spfel%(x2,y2)
830   NEXT x2
840 NEXT y2
850 yi=0:xi=0
860 FOR y=2 TO 49 STEP 4
870   yi=yi+1
880   FOR x=0 TO 79 STEP 4
890     xi=xi+1:IF xi>20 THEN xi=1
900     I$SHAPE,@shape$(spfel%(xi,yi)),x
,y,0
910   NEXT x
920 NEXT y
930 LOCATE 10,1:PRINT"SPIELSZENE:";level
940 ON level GOTO 950,980,1010,1040,1080
950 x=7:y=2:diamant=18
960 stein%=0:dauer=4
970 GOTO 1130
980 x=3:y=10:diamant=57
990 stein%=0:dauer=4
1000 GOTO 1130
1010 x=15:y=4:diamant=47
1020 stein%=0:dauer=4
1030 GOTO 1130
1040 x=5:y=10:diamant=44
1050 stein%=0:dauer=4
1060 GOTO 1130
1070 REM hauptsteuerprogramm
1080 EVERY 100,3 GOSUB 1670
1090 schmeanz=3:xx(1)=5:yy(1)=6:xx(2)=10
:yy(2)=11:xx(3)=16:yy(3)=3
1100 x=9:y=8:diamant=17
1110 stein%=0:dauer=4
1120 GOTO 1130
1130 IF schmeanz>0 THEN EVERY 100,3 GOSU
B 1670
1140 tot=0
1150 ax=x:ay=y
1160 j=JOY(0)
1170 IF j=1 THEN y=y-1
1180 IF j=2 THEN y=y+1
1190 IF j=4 THEN x=x-1
1200 IF j=8 THEN x=x+1
1210 IF x>19 THEN x=19
1220 IF x<2 THEN x=2
1230 IF y<2 THEN y=2
1240 IF y>11 THEN y=11
1250 IF spfel%(x,y)=4 THEN GOTO 1390
1260 IF x<>ax OR y<>ay THEN IF spfel%(x
,y-1)=2 THEN stein%=stein%+1:xste(stein%)
=x:yste(stein%)=y-1:stfall%(stein%)=dau
r
1270 IF stein%>0 THEN FOR zahl=1 TO stei
n%:stfall%(zahl)=stfall%(zahl)-1:IF stfa
ll%(zahl)=0 THEN GOSUB 1440 ELSE NEXT
1280 FOR verzoe=1 TO 50:NEXT verzoe
1290 IF tot=1 THEN exzs=REMAIN(3):RETURN

```

# programme

```
1300 IF spfel%(x,y)<>1 AND spfel%(x,y)<>
0 AND spfel%(x,y)<>5 THEN x=ax:y=ay:GOTO
1330
1310 spfel%(ax,ay)=0:spfel%(x,y)=5
1320 IF x<>ax OR y<>ay THEN SOUND 1,248,
10,6,,5
1330 h$=shape$(0)
1340 I$SHAPE,@h$,ax*4-4,ay*4-2,@shape$(5
),x*4-4,y*4-2,0
1350 I$SHAPE,@shape$(0),ax*4-4,ay*4-2
1360 I$SHAPE,@shape$(5),x*4-4,y*4-2
1370 GOTO 1150
1380 REM diamant abfrage
1390 spfel%(x,y)=0
1400 SOUND 2,100,10,6
1410 diamant=diamant-1:IF diamant=0 THEN
level=level+1:IF level>hoelevel THEN sx
sz=REMAIN(3):SOUND 1,284,45,5,,1:SOUND 2
,286,45,4,,2:SOUND 4,288,45,3:RETURN ELS
E GOTO 770
1420 GOTO 1260
1430 REM felsbewegung
1440 xs=xste(zahl):ys=ycte(zahl)
1450 IF spfel%(xs,ys+1)<>0 THEN 1560
1460 I$SHAPE,@shape$(0),xs*4-4,ys*4-2,0
1470 I$SHAPE,@shape$(2),xs*4-4,(ys+1)*4-
2,0
1480 SOUND 4,480,5,4,,30
1490 FOR axa=1 TO 10:NEXT axa
1500 spfel%(xs,ys)=0
1510 spfel%(xs,ys+1)=2
1520 IF spfel%(xs,ys-1)=2 THEN ys=ys-1:G
OTO 1460
1530 ycte(zahl)=ycte(zahl)+1:GOTO 1440
1540 stein%=stein%-1
1550 RETURN
1560 IF spfel%(xs,ys+1)=5 THEN 1600
1570 IF spfel%(xs,ys+1)<>6 THEN RETURN
1580 pu=pu+200
1590 GOTO 1460
1600 ENV B,=11,5000:SOUND 1,0,60,14,8,0,
15:sxsz=REMAIN(3):LOCATE 13,15:PRINT"GAM
E OVER":SOUND 1,0,500,5,,1:FOR a=1 TO 50
00:NEXT a:SOUND 1,0,2,1:tot=1:RETURN
1610 RESTORE 90:RETURN
1620 RESTORE 220:RETURN
1630 RESTORE 350:RETURN
1640 RESTORE 480:RETURN
1650 RESTORE 610:RETURN
1660 REM schmetterlingbewegung
1670 FOR a=1 TO schmeanz
1680 axx(a)=xx(a):ayy(a)=yy(a)
1690 IF x>xx(a) THEN xx(a)=xx(a)+1:GOTO
1730
1700 IF x<xx(a) THEN xx(a)=xx(a)-1:GOTO
1730
1710 IF y<yy(a) THEN yy(a)=yy(a)-1:GOTO
1730
1720 IF y>yy(a) THEN yy(a)=yy(a)+1
1730 IF spfel%(xx(a),yy(a))=5 THEN tot=1
:RETURN
1740 IF spfel%(xx(a),yy(a))<>0 THEN xx(
```

```
a)=axx(a):yy(a)=ayy(a)
1750 I$SHAPE,@shape$(0),axx(a)*4-4,ayy(a
)*4-2,0
1760 I$SHAPE,@shape$(6),xx(a)*4-4,yy(a)*
4-2,0
1770 spfel%(axx(a),ayy(a))=0
1780 spfel%(xx(a),yy(a))=6
1790 NEXT
1800 RETURN
1810 REM gamedesigner
1820 x=1:y=1
1830 FOR y=1 TO 12:FOR x=1 TO 20:spfel%(
x,y)=0:NEXT x:NEXT y
1840 x=1:y=1
1850 MODE 1
1860 FOR x=1 TO 20:spfel%(x,y)=3:espfel%(
x,y)=3:I$SHAPE,@shape$(3),x*4-4,y*4-2,0
:NEXT x
1870 y=12:FOR x=1 TO 20:spfel%(x,y)=3:es
pfel%(x,y)=3:I$SHAPE,@shape$(3),x*4-4,y*
4-2,0:NEXT x
1880 FOR y=2 TO 11:spfel%(1,y)=3:espfel%(
1,y)=3:I$SHAPE,@shape$(3),1*4-4,y*4-2:s
pfel%(20,y)=3:espfel%(20,y)=3:I$SHAPE,@s
hape$(3),20*4-4,y*4-2:NEXT y
1890 MODE 1
1900 LOCATE 9,8:PRINT CHR$(24);"1";CHR$(
24);" EDITOR"
1910 LOCATE 9,2:PRINT CHR$(24);"Game-Des
igner";CHR$(24)
1920 LOCATE 9,11:PRINT CHR$(24);"2";CHR$(
24);" LADEN"
1930 LOCATE 9,14:PRINT CHR$(24);"3";CHR$(
24);" SPEICHERN"
1940 LOCATE 9,17:PRINT CHR$(24);"4";CHR$(
24);" SPIELEN"
1950 LOCATE 9,20:PRINT CHR$(24);"5";CHR$(
24);" ENDE"
1960 a$=INKEY$:IF a$="" THEN 1960
1970 IF a$="1" THEN 2030
1980 IF a$="2" THEN 2240
1990 IF a$="3" THEN 2460
2000 IF a$="4" THEN 2660
2010 IF a$="5" THEN 3220 ELSE 1960
2020 REM editor
2030 MODE 1
2040 durchlauf=durchlauf+1
2050 LOCATE 17,3:PRINT CHR$(24);"EDITOR"
;CHR$(24)
2060 LOCATE 3,9:PRINT CHR$(24);"0";CHR$(
24):I$SHAPE,@shape$(0),10,18,0
2070 LOCATE 12,6:PRINT CHR$(24);"1";CHR$(
24):I$SHAPE,@shape$(1),28,12,0
2080 LOCATE 12,9:PRINT CHR$(24);"2";CHR$(
24):I$SHAPE,@shape$(2),28,18,0
2090 LOCATE 12,12:PRINT CHR$(24);"3";CHR
$(24):I$SHAPE,@shape$(3),28,24,0
2100 LOCATE 21,6:PRINT CHR$(24);"4";CHR$(
24):I$SHAPE,@shape$(4),46,12,0
2110 LOCATE 21,9:PRINT CHR$(24);"5";CHR$(
24):I$SHAPE,@shape$(5),46,18,0
2120 LOCATE 21,12:PRINT CHR$(24);"6";CHR
```

weiter Seite 57



## Biete Software

**Sensation!** Unser brandneuer Katalog ist da! Mit viel Hard- & Software für Ihren C 64 / 128 / VC 20! Supersonderangebote: Spiele, Adventures, Utilities, User, Interfaces, Accessoires u.v.m.! Z.B.: DFÜ-Interface, Disc Aid, Thermocontrol, Superaction mit White Shark etc.! Gleich anfordern gegen 1,60 DM in Marken bei: Comp Versand U. Tiedau, a. d. Ölmühle 24, 4270 Dorsten 1.

GESTRICHEN... haben wir fast alle Softwarepreise (z.B. CYRUS II 3-D Schach DM 41,90) u. das Angebot auf über 140 Artikel erweitert! Die Mitgliedschaft ist immer noch für 20,-- / Halbjahr zu haben!! COMPI-CLUB, J. Heise, A.d. Linde 8, 5226 Reichshof.

- ★★★★★★★★★★★★★★★★★★★★★★★
- ★ **SCHNEIDER CPC 464/664** ★
  - ★ Univers. Ein/Ausgabe-Interface 32 Bit ★
  - ★ TTL. Zum Anschl. vielseit. Hardware a. ★
  - ★ CPC. z. B. A/D Wandler, Relais, Roboter, ★
  - ★ Schul-Eisenbahn-Discoelectr. od. ★
  - ★ M&L Zubehör. Ausf. Bedien.-Anl. Eig- ★
  - ★ nes Netzteil. Programmierb. in Basic ★
  - ★ od. Asemml. Interrupttaugl.\* DM 198 ★
  - ★ EPROMPROGRAMMIERGERÄT an E/ ★
  - ★ A-Interface anschließb. f. 2716-27128, ★
  - ★ 2516-2564 m. komfort. Bedienersoftw. ★
  - ★ 28pol. Textoolsock. vorber. f. 27256. ★
  - ★ Maschinenspr.-Monitor. Lesen, Leert. ★
  - ★ Programm. Vergleich. v. Eprom's \* DM ★
  - ★ 178 M&L-Maximini&Lichter, Hauptstr. ★
  - ★ 1a, 5511 Mannebach, T. 06581/3535 ★
- ★★★★★★★★★★★★★★★★★★★★★★★

**BACKUP:** Disketten-Kopierprogramm für CPC 464 mit Vortex 512 K Erweiterung. Wie Discopy, jedoch ohne CP/M und in nur einem Durchgang. Für 20 DM (Brief), inkl. Anleitung, Porto und Verpackung von W. Spütz, Eyller-Str. 225, 4132 Kamp-Lintfort.

**COMMODORE 16/116:** Wir haben die Programme, die Sie suchen! Infos kostenlos bei: Andreas Bachler, Blücherstr. 24, 4290 Bocholt. Außerdem: Die C 16/116 Insider-Mappe für 6,-- DM oder das Überraschungspaket mit 5 Top-Spielen für nur 15 DM (Bitte nur als Schein oder Scheck beilegen!)

**C-16/116 20 Programme, C-16/116 20 Spiel- und Anwenderprogramme** für nur ★ 20 DM ★. Je Programm nur 1 Mark. Scheck oder Schein an: Daniel Beitlich, Kleine Höllbergstraße 6, 6000 Frankfurt 50. Oder Info anfordern gegen Porto.

**Funkfernsehprogramm für CPC 464 (RTTY).** Soft- und Hardware incl. Konverter, AFSK, Festtext-Drucker und Empfangsspeicher. Baudot und ASCII bis 200 Bard. Machen Sie Ihren CPC zum prof. Fernschreiber. Anwendung z. B. beim Amateurfunk. Info kostenlos bei: Stefan Peim, Soft- und Hardware, Friedrichruher Str. 32, 1000 Berlin 33, Tel. 030/8242943.

**Commodore 64/20 + Schneider CPC 464:** Supergünstige Programme zu Taschengeldpreisen! Kaufen Sie bei den Profis, denn wir sind exklusiv und günstig! Infos kostenlos: Andreas Bachler, Blücherstr. 24, 4290 Bocholt. Programme für: Schneider CPC 464 Commodore 64/20/16/116.

**Commodore 64, VC 20 C 16.** Fibu, EK-VK, Foto-, Film-, Video-, Musikarchiv, Lotto, Adress, 'YSI', Tabbello, Digitalo, Lernprogramme, Denktraining, Spiele, Karteikasten, Biblio, KFZ-Programme u.v.m. schon ab 19,90 DM! Katalog gegen 80 Pf. in Briefmarken, Tino Hofstedes Computerservice. A.d. Windmühle 8, 5010 Bergheim 5.

●●●● C 16/C 116 ●●●● C 16 / C 116 ●●●●  
 Biete Vokabel Drill Sergeant (Vokabellernprogramm) keine Raubkopie!! ●●●●●●●●  
 Tobias Brandt, Alte Trift 38, 2150 Buxtehude. Nur auf Kassette lieferbar!! ●●●●●●●●  
 Preis: 10 DM (Schein)!!! PS: Info liegt bei.

**C 16, C 116, Plus 4.** An alle Floppybesitzer: Habe das Adressenverarbeitungspr. Bis zu 2000 Adressen abspeichern möglich. 26 Dateien auf Diskette. Adressensuchroutine + Verändern + Löschen. ★★★ 10 DM ★★★ mit Disketten ★★★ bei: Alexander Füngling, Vor den Knäppen 30, 4410 Warendorf 1, Tel.: 0251/1022.

**Für CPC 464:** ca. 20 Original-Spiele, Topcalc und andere Textprogramme, dt. Anleitungen, VB 180,-. Peter Stange, 5800 Hagen, In der Senke 20.

Programme für die Arztpraxis auf dem Schneider Computer  
 Privatliquidation - Leistung-Statistik  
 Abrechnung-Unterstützung  
 Info bei: Fa. EFTEKTA  
 Am Wiggert 9c, 4500 Osnabrück  
 Telefon (05 41) 4 59 16

**Einmalige Gelegenheit f. VC-20.** Ich verk. 11 org. Spielecassett. z. B. Kegeln, Luftkampf, Ufo etc... (Preis 50,- DM) + 2 Spielhallen Module (Hau den Lukas, Pole Posit.) NP: pro Stck. 98,-/VP: pro Stck. 40,- + Vbk. z. Cassettenrecorder und VC-20 (Preis: 30,- DM). Schreibt an: Bernd Failner, Birkensteingasse 2, 8563 Schnaittach.

**Beachten Sie bitte auch die letzten Restangebote von unseren**

**Top-Hit-Programm-paketen**

**auf Seite 31 dieser Ausgabe.**

**VC-20 Programme,** z. B. Boeing 727-Simulator. Angebote auch Tel. 02324/61862, sonst. geg. frankierten Umschl., Horst Querfurt, Postf. 843042, 4320 Hattingen 14.

- ★★★★★★★★★★★★★★★★★★★★★★★
- ★ **Achtung!! VC-20-Besitzer.** Das ein- ★
  - ★ malige Angebot! Der VC-20 ist noch ★
  - ★ lange nicht out!! Info DM 5,- in Brief- ★
  - ★ marken! 40/80-Zeichenkarte 120 ★
  - ★ DM, 64 Kb + Steuerp, ohne geh. ★
  - ★ nur 120 DM, 32Kb + geh. nur 130 ★
  - ★ DM! DFÜ mit dem VC -20 jetzt ★
  - ★ zugreifen. HPT Electronic, Am Rain ★
  - ★ 23, 8024 Deisenhofen. ★
- ★★★★★★★★★★★★★★★★★★★★★★★



## Fehlerhafte Listings

Ich finde es eine bodenlose Frechheit wie Sie mit Ihren Lesern umspringen. Für den C-16 habe ich jetzt schon 6 Spiele abgetippt und keines funktionierte. Aber eines muß man Ihnen lassen, Sie haben immer schöne Absturzbilder. Mein Freund besitzt den VC-20 und bei ihm klappt (Oh Wunder) sogar der Vorspann. Eines ist uns klargeworden, Sie wollen nur Geld schneffeln? Sie haben von der Materie keine Ahnung! Sonst wüßten Sie, daß man Spiele für den C16/116 auch auf dem Plus 4 abspielen kann. Denn beide haben BASIC 3.5, nur der Speicher ist größer beim Plus 4. Ihre Programme habe ich fehlerfrei abgetippt. Mein Freund und ich kaufen Ihre Zeitschrift nicht mehr und warnen auch alle anderen Leser.

**Manuel Japes, Lüdenscheid**

PS.: Diesen Brief werden Sie sowieso nicht abdrucken!

**Redaktion:** Zunächst einmal haben wir Ihren Brief doch abgedruckt, obwohl der Brief natürlich eine ganze Reihe von Vorwürfen und Inkompetenz enthält. Aber wir möchten auch zu kritischen Leserbriefen Stellung nehmen und alle Leser ermuntern, uns Ihre Probleme mitzuteilen.

Die von Ihnen abgetippten C-16 Programme müssen mit Sicherheit noch einige Eingabefehler enthalten, da diese fehlerfrei veröffentlicht wurden. Ebenfalls trifft dies auch für die VC-20 Programme Ihres Freundes zu, da sonst eine wahre Flut von Leserbriefen über nicht funktionierende Programme bei uns eingegangen wäre.

Die in Basic geschriebenen Programme des C16/116 laufen tatsächlich auf dem Plus 4, da das Basic identisch ist. Werden in den Programmen jedoch PEEK- und POKE-Befehle verwandt, läuft nichts mehr. Sie sehen also, ein wenig technisches Know-how und Erfahrung im Bereich der Homecomputer-Szene haben wir schon gesammelt. Im Übrigen ist eine Warnung der Leser immer nur erfolgreich, wenn die

aufgestellten Behauptungen auch hieb- und stichfest belegt werden können. Natürlich möchte sich aber auch unsere Redaktion nicht ganz von Fehlern freisprechen. Sollten noch mehr Leser Schwierigkeiten mit den Programmen haben, lassen Sie es uns wissen, wir helfen gern weiter.

## Probleme mit dem CPC-Checksummer

Heute habe sich meine ersten Versuche mit dem CPS-Checksummer gefahren! Ich finde die SUPER!!!

Leider habe ich Schwierigkeiten mit DELETE 56200. Wie kann ich das angehängte Programm löschen?

**Thomas Stöferle jun., Erbach-Ringingen**

**Redaktion:**

Bei genauem hinsehen erkennen Sie, daß Ihnen ein Zahlendreher unterlaufen ist. Um den Checksummer wieder zu löschen, gegen Sie folgenden Befehl ein: DELETE 65200 -. Der Bindestrich darf nicht vergessen werden.

## Maschinenspracheprogramme für VC-20

Ich bin begeisterter Leser Ihrer Zeitschrift und habe eine Bitte. In Ihrer Zeitschrift sind pro Ausgabe mindestens 2 VC-20 Spiele abgedruckt. Ich würde es begrüßen, alle Spiele mit einer geschwindigkeitsbestimmenden Maschinensprachroutine abzutippen. Ansonsten ist Ihre Zeitschrift spitze.

**Jürgen Gmeiner, Würselen**

**Redaktion:** Vielen Dank für das dicke Lob unserer Zeitschrift. Unsere Redaktion plant in den künftigen Ausgaben sogar noch eine Erweiterung der Programmangebote für den VC-20. Zu Ihrer Bitte mit den Maschinensprachroutinen müssen wir Ihnen sagen, daß dies natürlich von den einzelnen Programmautoren bei der Entwicklung Ihrer Software berücksichtigt werden muß. Bei unserer Auswahl der

Programme wird diese Tatsache jedoch meistens berücksichtigt.

## Programmfehler in „SUPER-BREAK-OUT“ aus Heft 11/85

Nachdem ich das Programm mit „RUN“ starte erscheint „Maschinen-Data-Error“. Gebe ich nun „CONT“ ein, erscheint das Spielbild. Nach Abfrage der Spielstärke beginnt das Spiel, jedoch werden weder Punkte gezählt, noch Steine zerstört. Worum liegt das?

**Thomas Martin, Saarbrücken-Dudweiler**

**Redaktion:** Das von Ihnen erwähnte Programm wurde vollkommen fehlerfrei veröffentlicht. Die bei Ihnen aufgetretene Fehlermeldung ist ein deutlicher Hinweis auf einen Tippfehler bei der Eingabe. In solchen Fällen darf auf keinen Fall der Befehl CONT verwandt werden, da der Fehler auf diese Weise nur zeitlich verschoben wird. Überprüfen Sie also nochmals sorgfältig Ihr Programm auf Eingabefehler.

## Reverse Zeichen auf dem C-16/116

Anfrage: Wie bekomme ich die, bei dem Programm „City-Runner“, beschriebenen reversen Zeichen auf den Bildschirm? Oder besser gesagt, wie bekomme ich überhaupt reverse Grafikzeichen in Verbindung mit einem Print auf den Bildschirm?

**Thorsten Hup, Herne**

**Redaktion:** Reverse Steuerzeichen innerhalb von Listings sind Steuerzeichen, welche eine bestimmte Funktion oder Aufgabe erfüllen. Als Beispiel sind hier die Cursor-Tasten oder die CTRL-Taste in Verbindung mit der Farbumschaltung zu nennen. Diese Steuerzeichen können nur innerhalb von Anführungszeichen erzeugt werden und erscheinen dann auch auf dem Bildschirm oder Drucker. Sie müssen also darauf achten, daß vor dem Erzeugen dieser reversen Grafikzeichen, die Anführungszeichen stehen müssen.

```

$(24):IPSHAPE,@shape$(6),46,24,0
2130 a$=INKEY$:IF a$="" THEN 2130
2140 IF a$="0" THEN symb=0:GOTO 2900
2150 IF a$="1" THEN symb=1:GOTO 2900
2160 IF a$="2" THEN symb=2:GOTO 2900
2170 IF a$="3" THEN symb=3:GOTO 2900
2180 IF a$="4" THEN symb=4:GOTO 2900
2190 IF a$="5" THEN symb=5:GOTO 2900
2200 IF a$="6" THEN symb=6:GOTO 2900
2210 IF INKEY(18)=0 THEN 1890
2220 GOTO 2130
2230 REM laden
2240 MODE 1
2250 CLOSEIN:CLOSEOUT
2260 LOCATE 17,3:PRINT CHR$(24);"LADEN";
CHR$(24)
2270 LOCATE 1,10:INPUT"Name des zu laden
den Programms ";name$
2280 OPENIN name$
2290 FOR sas=1 TO 12
2300   FOR sbs=1 TO 20
2310     INPUT #9,espfel%(sbs,sas)
2320   NEXT sbs
2330 NEXT sas
2340 INPUT #9,ediamant
2350 INPUT #9,xxpo
2360 INPUT #9,yypo
2370 INPUT #9,eschmeanz
2380 IF eschmeanz<=0 THEN 2440
2390 FOR schme=1 TO eschmeanz
2400   INPUT #9,xx2(schme)
2410   INPUT #9,yy2(schme)
2420 NEXT schme
2430 durchlauf=1
2440 CLOSEIN:GOTO 1890
2450 REM speichern
2460 MODE 1
2470 LOCATE 17,3:PRINT CHR$(24);"SPEICHE
RN";CHR$(24)
2480 LOCATE 1,10:INPUT"Name des zu speic
hernden Programms ";name$
2490 OPENOUT name$
2500 FOR sas=1 TO 12
2510   FOR sbs=1 TO 20
2520     PRINT#9,espfel%(sbs,sas)
2530   NEXT sbs
2540 NEXT sas
2550 PRINT#9,ediamant
2560 PRINT#9,xxpo
2570 PRINT#9,yypo
2580 PRINT#9,eschmeanz
2590 IF eschmeanz<=0 THEN 2640
2600 FOR schme=1 TO eschmeanz
2610   PRINT#9,xx2(schme)
2620   PRINT#9,yy2(schme)
2630 NEXT schme
2640 CLOSEOUT:GOTO 1890
2650 REM spiel ueber gamedesigner
2660 MODE 1
2670 FOR sas=1 TO 12
2680   FOR sbs=1 TO 20
2690     spfel%(sbs,sas)=espfel%(sbs,sas
)
2700   NEXT sbs
2710 NEXT sas
2720 tot=0
2730 hoelevel=1:level=1
2740 x=xxpo:y=yypo
2750 schmeanz=eschmeanz
2760 IF schmeanz>0 THEN FOR schme=1 TO s
chmeanz:xx(schme)=xx2(schme):yy(schme)=y
y2(schme):NEXT schme
2770 dauer=4
2780 stein%=0
2790 LOCATE 12,1:PRINT"SPIELSZENE: ";nam
e$
2800 FOR sas=1 TO 12
2810   FOR sbs=1 TO 20
2820     IPSHAPE,@shape$(spfel%(sbs,sas)
),sbs*4-4,sas*4-2,0
2830   NEXT sbs
2840 NEXT sas
2850 diamant=ediamant
2860 GOSUB 1130
2870 GOTO 1890
2880 GOTO 2880
2890 REM editor fuer bildaufbau
2900 h$=""
2910 MODE 1
2920 FOR sas=1 TO 12:FOR sbs=1 TO 20:IPS
HAPE,@shape$(espfel%(sbs,sas)),sbs*4-4,s
as*4-2,0
2930 NEXT sbs,sas
2940 IF durchlauf=1 THEN FOR yay=2 TO 11
:FOR xax=2 TO 19:espfel%(xax,yay)=symb:N
EXT xax,yay:GOTO 2030
2950 x=2:y=2
2960 IPSHAPE,@hand$,x*4-4,y*4-2,3
2970 h$=shape$(espfel%(x,y))
2980 axx=x:ayy=y
2990 j=JOY(0)
3000 IF INKEY(18)=0 THEN a$="":GOTO 2030
3010 FOR verzoe=1 TO 50:NEXT verzoe
3020 IF j=1 THEN y=y-1:GOTO 3150
3030 IF j=2 THEN y=y+1:GOTO 3150
3040 IF j=4 THEN x=x-1:GOTO 3150
3050 IF j=8 THEN x=x+1:GOTO 3150
3060 IF j<>16 THEN GOTO 2990
3070 IF espfel%(x,y)=6 THEN GOTO 2980
3080 IF espfel%(x,y)=4 THEN ediamant=edi
amant-1
3090 IF espfel%(x,y)=5 THEN GOTO 2980
3100 IF symb=6 THEN eschmeanz=eschmeanz+
1
3110 IF symb=6 THEN xx2(eschmeanz)=x:yy2
(eschmeanz)=y
3120 IF symb=4 THEN ediamant=ediamant+1
3130 IF symb=5 THEN xxpo=x:yypo=y
3140 espfel%(x,y)=symb
3150 IF x>19 THEN x=19
3160 IF x<2 THEN x=2
3170 IF y>11 THEN y=11
3180 IF y<2 THEN y=2
3190 !XSHAPE,@h$,axx*4-4,ayy*4-2,@hand$,

```



# BASIC-Befehle auf den Funktions-Tasten des CPC 464

Der CPC 464 gibt uns die angenehme Möglichkeit, maximal 32 Tasten mit beliebigen Ausdrücken zu belegen. 13 dieser Tasten liegen fest auf dem Zahlen-Block (Bereich 128 bis 140), die anderen können frei auf die Schreibmaschinen-Tastatur gelegt werden (Bereich 141 bis 159). Für die Schreibmaschinen-Tastatur begibt man sich vorteilhafterweise auf die „CTRL“-Ebene; die Groß- und Kleinschreibung, wie sie normalerweise üblich ist, wird dann nicht beeinträchtigt. Es können – wie in folgendem Programm – durchaus insgesamt 119 Zeichen auf die Tasten gelegt werden; das Handbuch sagt: maximal 100 Zeichen.

## Die Vorteile von BASIC-Befehlen auf den Funktionstasten.

Sobald Sie sich an die Lage der BASIC-Befehle auf den Tasten gewöhnt haben, ersparen Sie sich beim Neu-Eintippen von Programmen ca. 50 % der Zeit. Außerdem kommen weniger Schreibfehler (SYNTAX ERROR) vor, was Ihnen ebenfalls beim Korrigieren (DEBUGGING) Zeit erspart. Insgesamt – so zeigt meine Erfahrung –

kommt man mit BASIC-Befehlen auf den Funktions-Tasten ungefähr zweimal schneller zu seinem kompletten und fehlerfreien Programm.

## Vorüberlegungen zur Belegung der Tasten.

Um maximale Zeit beim Eintippen und Korrigieren zu sparen, suchte ich erst einmal die von mir am häufigsten benutzten Kommandos heraus. Diese Häufigkeit ist wohl je nach User unterschiedlich. Bei meiner Belegung können Sie erkennen, daß ich kaum Graphiken benutze und wohl noch weniger Töne.

Aus der Tatsache, daß ich Schreibmaschinenschreiben kann, ergab sich dann, daß ich den Zahlenblock voll belegte. (Man benötigt hier für einen Befehl nur einen Finger.) Da sich die „CTRL“-Taste beim CPC 464 rechts unten in der Schreibmaschinen-Tastatur befindet, kam für mich nur eine Belegung von Tasten in der linken Hälfte der Schreibmaschinen-Tastatur in Frage. (Mit der rechten Hand drückt man „CTRL“, mit der linken eine Schreibmaschinen-Taste; dies geht recht schnell.) Schon daraus ergibt es sich, daß ich nicht

auf die sonst übliche Praxis zurückgreifen konnte und auch nicht wollte, jeweils den Anfangsbuchstaben eines Befehls als die Befehls-Taste (auf der „CTRL“-Ebene) zu nehmen. Also nicht: „CTRL“ und „L“ = „LIST“!

Stattdessen zog ich es vor zusammenhängende Befehle nahe zusammen, am besten nebeneinander, auf die Tastatur zu legen. Und hierbei ging ich dann auch noch so vor: die am meisten benutzten Funktionstasten liegen einmal unten und zum anderen mal auf dem Zahlenblock. Also: von allen Befehlen, die auf den Tasten liegen, liegen die, welche ich am wenigsten benutze, links oben auf der Schreibmaschinen-Tastatur („auto“, „move“, „draw“). Diejenigen Befehle, die ich am häufigsten beim Eingeben eines neuen Programmes benutze, liegen unten auf dem Zahlen-Block; es sind: „cls“, „list“, „save“ & „run“. (Das bisherige „RUN“, welches ein Einlesen und Starten eines neuen Programmes hervorruft, ist hier ohne „!“ Das fehlende Anführungszeichen (!) kann ja leicht durch „SHIFT“ und „2“ hinzugefügt werden.)

## Schreibmaschinen-Tastatur-Befehle

```

auto  <2>      move  draw
mode  border   ink    paper  pen
if    and      or     then   else
<z>   for      to     step   next

```

## Zahlenblock-Befehle

```

read  data  restore
on    goto  gosub
return locate input
cls   list  save
<CTRL> run

```

## Beschreibung der belegten Schreibmaschinen-Tasten.

Die oberste Reihe der Schreibmaschinen-Tastatur erhält drei BASIC-Befehle: „auto“, „move“ und „draw“. Hierbei ist jedoch die Taste „2“ ausgespart; hier liegt mit „CTRL“ für meinen Drucker das „B“. „Move“ und „draw“ können durch ein angehängtes „r“ in „mover“ und „drawr“ umgewandelt bzw. erweitert werden.

In der darunterliegenden Reihe von „Q“

bis „T“ liegen BASIC-Befehle, die einfache Bildschirm-Veränderungen bringen: „mode“, „border“, „ink“, „paper“ und „pen“. Durch ein Anhängen von „ey“ bzw. „ey\$“ kann man „ink“ zu „inkey“ bzw. „inkey\$“ erweitern. Die Reihe „A“ bis „G“ haben die Bedingungs-Befehle „if“, „and“, „or“, „then“ und „else“ erhalten. Tippt man zuerst ein „x“ und dann „CTRL“ und „D“, erhält man zusätzlich „xor“.

Auf „X“ bis „B“ liegen die Zähl-Befehle

„for“, „to“, „step“ und „next“, sie bilden eine zählende Schleife. In dieser untersten Reihe der Schreibmaschinentasten habe ich bewußt das „Y“ ausgelassen, um eine gewünschte mögliche Anpassung der vorhandenen amerikanischen Tastatur an die deutsche DIN-Tastatur nicht unnötig zu erschweren; ebenso ist das „Z“ in der obersten Buchstabenreihe bewußt ausgespart worden.

## Beschreibung der belegten Zahlenblock-Tasten.

Auf den Tasten 7-9 liegen die DATA-Befehle „read“, „data“ und „restore“. Auf 4-6 sind „on“, „goto“ und „gosub“; links darunter auf 1 das dazugehörige „return“. Hier sind also die Sprung-Befehle. Jetzt folgen dann noch die von mir häufig benutzten Einzelbefehle. Auf 2 und 3: „locate“ und „input“. Auf 0: „ENTER: „cls“, „list“ und „save“. Und zum Schluß auf „CTRL“

und „ENTER“ das „run“ ohne Anführungszeichen „“.

Noch kurz eine Bemerkung:

Ich habe alle BASIC-Befehle klein geschrieben, denn der CPC 464 schreibt sie in seinem Listing von selber groß, wenn ... er sie angenommen hat. Wenn Sie sich also eine Programmzeile auf dem Bildschirm ausgeben lassen, und ein BASIC-Befehl ist klein geschrieben, dann ist

der Befehl eben vom CPC 464 nicht als Befehl erkannt. Dann ist möglicherweise der Befehl selber falsch geschrieben, wie beispielsweise „prtt“ statt „print“, oder es ist ein Zwischenraum, eine Leerstelle, vergessen worden wie etwa bei „inputa“, was „input a“ geschrieben werden muß. Legt man also Befehle auf die Tasten des CPC 464, so empfiehlt es sich immer - für ein schnelles DEBUGGING -, die Befehle in Kleinschrift auf die Tasten zu legen.

```

10 'C.Hass:Basic-Befehle
20
30 FOR i=128 TO 140
40 READ a$:KEY i,a$:IF i=139 THEN KEY i,
a$ + CHR$(34)
50 NEXT
60 DATA cls,return,locate,input,on,goto,
gosub,read,data,restore,list,save,run
70
80 FOR i=141 TO 143
90 READ a$,a,b$:KEY i,a$:b=ASC(b$):c=b-1
6:KEY DEF a,i,b,c,i
100 NEXT
110 DATA auto,64,1,move,57,3,draw,56,4
120
130 FOR i=144 TO 157
140 READ a$,a,b$:KEY i,a$:b=ASC(b$):c=b-
32:KEY DEF a,i,b,c,i
150 NEXT
160 DATA mode,67,q,border,59,w,ink,58,e,
paper,50,r,pen,51,t,if,69,a,and,60,s,or,
61,d,then,53,f,else,52,g,for,63,x,to,62,
c,step,55,v,next,54,b
170
180 MODE 2
190 PRINT#g,"Schreibmaschinen-Tastatur-B
efehle                Zahlenblock-Befehle"

```

```

200 PRINT#g,"-----"
210 PRINT#g,"auto <2>      move draw
read data restore"

220 PRINT#g,"mode border ink paper
pen on goto gosub"
230 PRINT#g,"if and or then
else return locate input"
240 PRINT#g,"<z> for to step
next cls list save'"
250 PRINT#g,"
<CTRL> run"
260 IF g=8 THEN END ELSE IF g=0 THEN g=8
:GOTO 190

```

CHECK V1:PRUEFSUMMEN:		130-	140:	26232
		150-	160:	27342
		170-	180:	606
ZEILENNR.: SUMMEN		190-	200:	43799
10-	20:	210-	220:	11561
30-	40:	230-	240:	95916
50-	60:	250-	260:	115351
70-	80:			
90-	100:			
110-	120:			
				GESAMTCHECKSUMME: 37785

## Programm-Beschreibung

Dieses Programm kann leicht auf die Wünsche des jeweiligen Programmierers umgeschrieben werden; es besteht aus vier Teilen, wobei der letzte Teil (Zeile 190 bis 260) auf dem Bildschirm die Tasten-Belegung darstellt. Ist ein Drucker vorhanden, so kann dieser dann die Tastenbelegung auf einen schmalen Streifen Papier ausdrucken, den man mit Klebeband leicht auf dem Keyboard oberhalb der Schreibmaschinen-Tasten befestigen

kann. (Sonst Belegung kopieren oder auf kariertem Papier abschreiben.)

Die Belegung der Tasten erfolgt durch drei verschiedene Schleifen. Die Codierung der Tasten des Zahlenblocks sind aus dem Handbuch ersichtlich; hier genügt das einfache „key“-Kommando. Für den Befehl „save“ wurde eine kurze „if“-Bedingung in Zeile 40 hinzugefügt, welche „an „save“ anhängt. (Zahlenblock-Tasten = Zeilen 30 bis 60.)

Für die oberste Reihe der Schreibmaschinen-Tasten gilt, daß die ASCII-Werte eine

Taste mit „SHIFT“ um 16 geringer sind als die der Taste ohne „SHIFT“. In den anderen Zeilen der Schreibmaschinentastatur ist der Differenzwert dagegen 32. Die DATA-Zeilen (Zeile 110 und 160) sind aber gleich aufgebaut: z. B.: auto,64,1. Und dies bedeutet: das Kommando „auto“ soll auf die Taste 64 kommen, welche unge-„shift“-et das Zeichen „1“ ergibt, usw. Eine Modifizierung dieser Tastatur-Belegung oder eine völlige Umänderung ist also sehr leicht durch das Umändern der DATA-Zeilen möglich.

# Tips & Tricks für den Schneider

## Tapefile/Discfile (für Kassette und Diskette zu verwenden)

Das Kopieren von Binären-Files bereitet beim Schneider CPC leider immer ein paar Probleme. Zwar hat der Schneider-Computer vorzügliche Befehle zum Laden und Abspeichern von beliebigen Speicherbereichen, jedoch existiert kein Befehl zum Feststellen von Startadresse und Länge eines schon abgespeicherten Programmes.

Das hier vorgestellte Programm macht nun diesem Manko ein Ende.

Sie starten das Programm TABEFILE mit RUN und werden dadurch aufgefordert, eine Kassette einzulegen. Dieses bestätigen Sie durch das Betätigen der Taste PLAY und einer weiteren beliebigen Taste. Nach kurzer Zeit erscheint die Meldung „FOUND...“ auf dem Bildschirm. Ebenfalls einige Augenblicke später wird die Meldung "BLOCK OK" ausgegeben. In diesem Augenblick betätigen Sie die Taste ESC. Dadurch werden die Startadresse, die Länge und die Art des Programmes angezeigt. Diese Daten können Sie nun zum Kopieren von dem entsprechenden File verwenden.

```

10 ' *** tapefile ***
20 ' copyright frank brall
30 '
40 ON BREAK GOSUB 290
50 CLS:PRINT"**** TAPEFILE ****" F.Bra
11"
60 PRINT:PRINT
70 PRINT"Kassette einlegen und nach der"
80 PRINT"Meldung 'OK' ESC ESC druecken !"
":PRINT
85 :TAPE.IN
90 CAT:STOP
290 ' *** auswertung ***
300 PRINT:PRINT:PEN 3:PRINT"Auswertung:"
:PRINT
320 name$="":FOR i=47244 TO 47259
330 name$=name$+CHR$(PEEK(i)):NEXT i
340 laenge= PEEK(47269)*256+PEEK(47268)
350 start = PEEK(47266)*256+PEEK(47265)
360 art =PEEK(47262)
370 PRINT"Filename: "name$
380 PRINT"Startadr: "start
390 PRINT"Laenge : "laenge;" Bytes"
400 PRINT:PRINT"Art : "art;" (0/1 Basic
2 Binaer 22 Text)"

```

CHECK V1:PRUEFSUMMEN:

ZEILENNR.: SUMMEN  
10- 20: 43447

30- 40: 1402  
50- 60: 33722  
70- 80: 96139  
85- 90: 4962

290- 300: 44718  
320- 330: 93430  
340- 350: 109771  
360- 370: 31047

380- 390: 57891  
400- : 16090

GESAMTCHECKSUMME: 57777

## 5.25" Floppy Disk Stationen F1-S und F1-D für den Schneider CPC 464

Die Floppy-Disk-Stationen F1-S (ingle = 1 Laufwerk) und F1-D (ouble = 2 Laufwerke) sind hochwertige und leistungsstarke Diskettenspeicher mit großer Speicherkapazität und hoher Datenübertragungsgeschwindigkeit. Verwendet wird die Standard Diskettengröße 5.25". Modernste Controller Elektronik, sowie Laufwerke (BASF 6138) mit sehr guter Einsatzbewährtheit, garantieren ein dauerhaftes und fehlerfreies Arbeiten der Station. Professionelle Betriebssysteme (CP/M 2.2 und erweitertes BASIC Disketten Betriebssystem VDOS) bieten dem Be-

nutzer die Möglichkeiten zum Einsatz hochwertiger Anwender- und Systemsoftware, sowie eigener Programme.

Die Floppy-Disk-Station F1-S kann mit dem Aufrüst-Kit A1-S zur Floppy-Disk-Station F1-D ausgebaut werden.

### Hardwarekonzept:

- 5.25" Laufwerke 6138 der Firma BASF; 708 KB formatierte Speicherkapazität pro Laufwerk (d. h. 1.4 MB bei der Station F1-D); 80 Spuren pro Seite; Zweiseitig (DS = Doppelkopf Laufwerk); Doppelte Schreibdichte (DD);

4 msec Spurwechselzeit; IBM 34 Format; es werden Disketten der Bezeichnung DS/DD 96 tpi benötigt.

- Der Controller wird nicht direkt an den Expansions Bus angesteckt, sondern mit einem Flachbandkabel mit entsprechendem Stecker mit diesem verbunden. Dadurch werden die Kontakte nicht mechanisch belastet, wodurch sich eine optimale Kontaktsicherheit ergibt.
- Mit dem Adapterkabel FDA-1, kann ein 3" Schneider Laufwerk angeschlossen werden. Hierzu sind keinerlei Soft-

oder Hardwareänderungen nötig. Das 3" Laufwerk wird voll integriert.

- Ausführliches deutsches Handbuch mit CP/M Befehlseinführung.
- Solide ergonomische, farblich an den CPC 464 angepaßte Gehäuse für Laufwerke und Controller.
- Abmessungen:  
305 x 150 x 90 (L x B x H in mm)
- Gewichte: FS-S 3,1 kg, F1-D 4,6 kg.

#### Softwarekonzept:

- 2 Betriebssysteme:

\* CP/M 2.2 voll angepaßt mit Utilities  
\* erweitertes BASIC Diskettenbetriebssystem VDOS (Vortex Disk Operating System). Zum Arbeiten unter BASIC wird keine Systemdiskette benötigt, da z. B. selbst zum Formatieren einer Diskette ein eigener, im Controller fest eingebauter Befehl vorhanden ist.

- Alle Meldungen in Deutsch.
- Alle Utilities sind menügesteuert.

#### Preise:

Floppy Disk Station F1-S = 1.198,- DM

Floppy Disk Station F1-D = 1.698,- DM

#### Lieferumfang:

- 1.1 Diskettenstation (ein oder zwei Laufwerke) mit Controller und allen Kabeln.
- 2.1 CP/M 2.2 Systemdiskette mit dem CP/M 2.2 Betriebssystem und Utilities 3.1 Deutsches Handbuch mit ausführlicher CP/M 2.2 Befehlseinführung.

Diese Floppy Disk Stationen sind bei der

Vortex Vertriebs GmbH

Klingenberg 13

7106 Neustadt 5

erhältlich.

## Einführung in die höheren Programmiersprachen

### FORTH

Forth gehört zu den höheren Programmiersprachen, die auf vielen Personal-Computern Anwendung finden und auch im Homecomputer-Bereich immer mehr berücksichtigt wird. Die Vorteile dieser Sprache liegen in ihrer Flexibilität und in der Schnelligkeit, mit der in Forth geschriebene Programme laufen können. Dieses Buch gibt dem Anfänger eine erste Einführung in die Strukturen und die Besonderheiten dieser modernen Programmiersprache. Jeder der dieses Buch liest braucht keine Vorkenntnisse vorzuweisen, da es die Grundlagen von Forth und



der Datentechnik weitergibt. So werden unter anderem

**Basiskonzepte,  
die Handhabung von Zahlen,  
die Handhabung von Variablen,  
Verzweigung  
und Übungsbeispiele**

aufgezeigt. Erschienen ist dieses Buch im

Goldmann-Verlag,  
München,

ISBN 3-442-13128-6 für 12,80 DM

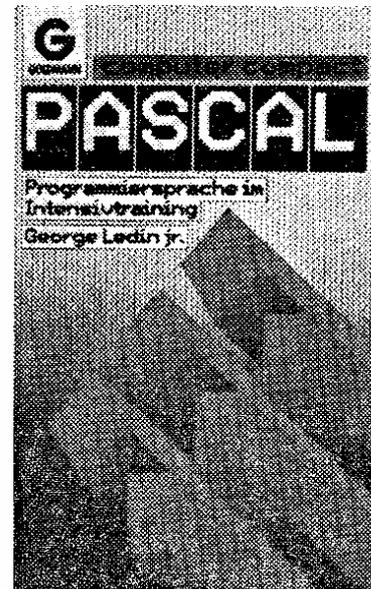


Als weitere Neuerscheinung im Bereich der höheren Programmiersprachen stellt der Goldmann-Verlag

### PASCAL

vor. Diese Programmiersprache ist fast ausschließlich auf größeren Rechner-Anlagen installiert. Gelegentlich findet sie aber auch auf Personalcomputern Anwendung. Der Profi weiß es ohnehin, Turbo-Pascal ist zur Zeit der absolute Renner in der Homecomputerszene. Das kommt natürlich nicht von ungefähr, denn PASCAL zeichnet sich durch besondere Übersichtlichkeit in den verfaßten Programmen aus. Das Buch gibt einen Überblick über

**die Struktur von Pascal-  
Programmen,  
Pascal-Unterprogramme,  
IF- und FOR-Anweisungen,  
Schleifen  
und praktische Beispiele**



Der Anfänger erhält hier auf leicht verständliche, gegliederte Art den Einstieg in die Programmiersprache Pascal und macht ihn in effizienter Weise mit den Besonderheiten dieser modernen Programmiersprache vertraut.

- ISBN 3-442-13127-8,

Preis: 12,80 DM

# Computer Jahrbuch '86

von Dietmar Eirich/Peter Herzberg

Dieses Buch gibt jedem Computer-Fan einen Überblick über die ganze Welt der Microcomputer in einem Band. Jeder der einen Microcomputer erwerben möchte, sollte unbedingt dieses Buch lesen. Er findet hier

den Stand der Technik in der Mikro-Elektronik die Unterschiede der Computer-Systeme einen Einkaufs-Check

Entscheidungshilfen mit Preisvergleichstabelle Adressen von Herstellern und Computerclubs u.v.m.

In diesem Jahrbuch ist das zusammengefaßt, was jeder Computer-Anwender, -Einsteiger, Freak und Profi wissen muß, um sich im verwirrenden Angebot der Systeme und Programme zurechtzufinden. Hier finden Sie Kriterien und Auskünfte.

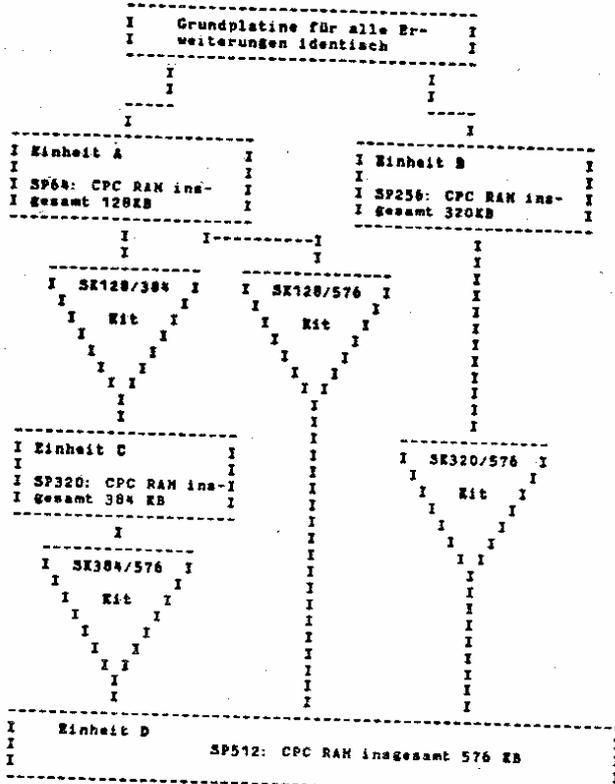
Heyne Computerbücher München,

ISBN 3-453-47056-7,

Preis: 16,80 DM

## RAM Karten SP64, SP256, SP320, SP512 für den Schneider CPC 464

Ausbauschema RAM - Erweiterungen:



Bemerkungen:

- Die Bezeichnung SKxxx/yyy bedeutet, daß der gesamte RAM Speicher des CPC mit diesem Kit von xxx KB auf yyy KB erweitert wird: z. B. SK128/384 von 128 KB auf 384 KB.
- Jedes Kit besteht aus einem Satz Speicher IC's und einer Firmware - ROM.

Die RAM Karten der SP-Serie (64 KB, 256 KB, 320 KB und 512 KB) vergrößern den CPC Arbeitsspeicher auf volle 64 KB und stellen einen 32 KB Druckerpuffer (Spooler) zur Verfügung. Vom Speicher her gibt es damit keinerlei Einschränkungen mehr bezüglich Lauffähigkeit von Standard CP/M Software. Programme wie dBase oder WordStar arbeiten nun uneingeschränkt.

Die Karten SP256, SP320 und SP512 bieten zudem die Möglichkeit einer RAM-Floppy, also eines superschnellen Massenspeichers. Alle Karten basieren auf derselben Platine. Jede Karte kann durch die entsprechenden Ausbausätze (Kit's) bis zur SP512 ausgebaut werden: kein Löten, nur Einstecken von integrierten Schaltungen. Die Speicherkarte wird direkt im CPC untergebracht.

Hardwarekonzept:

- Erweiterung des Arbeitsspeichers von 48 KB RAM auf 64 KB RAM
- Bereitstellung eines 32 KB Druckerpuffers
- RAM - Floppy bei den Karten SP256, SP320 und SP512
- Durch Ausbausätze jede Karte bis zur SP512 ausbaubar, z. B. SP64 zur SP512
- Einbau der Platine direkt in den CPC
- völlig unkomplizierter Einbau: kein Löten

Softwarekonzept:

- Durch die mitgelieferte Software (für BASIC im ROM, für CP/M auf Diskette) wird der Speicher sowohl unter BASIC als auch unter CP/M voll in die CPC Soft- und Hardwareumgebung integriert und ist damit vielfältig einsetzbar.

Lieferumfang:

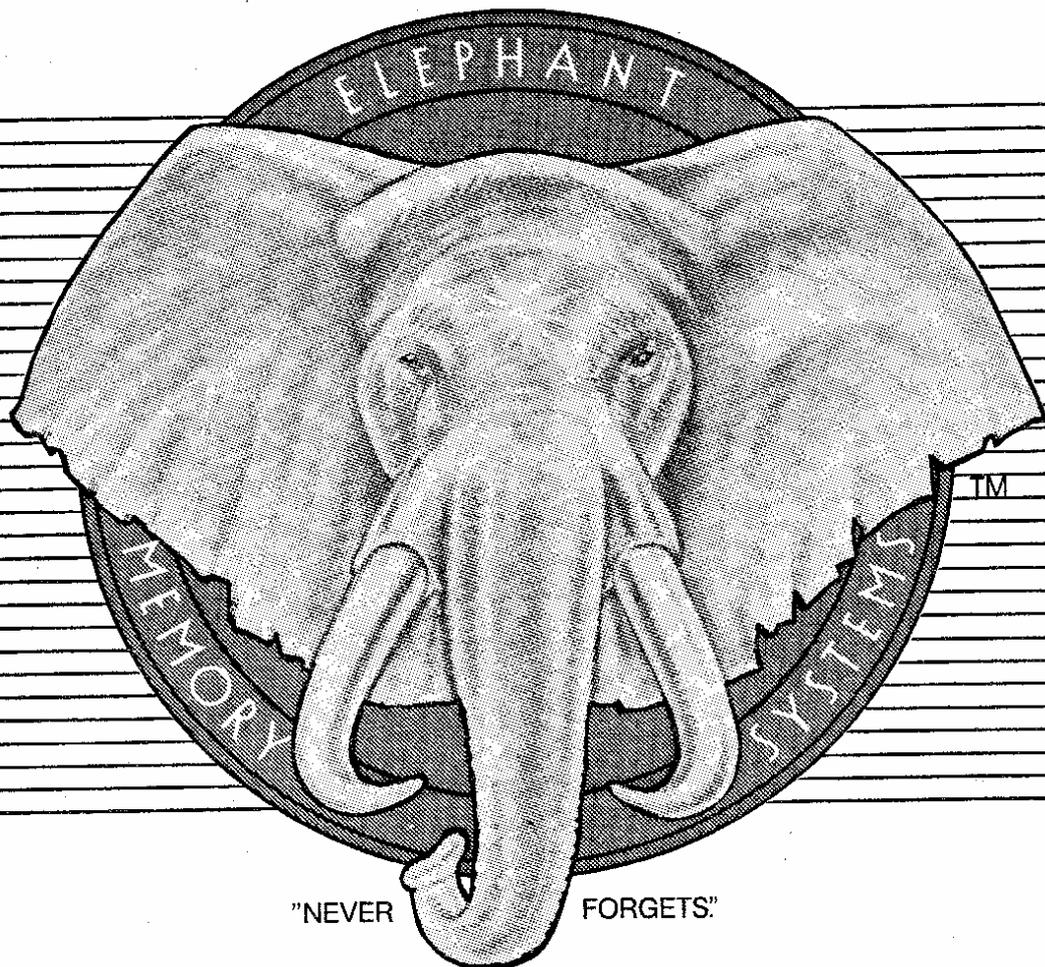
- SP64, SP256, SP320, SP512
- 1 Speicherplatine
  - 1 Systemdiskette
  - 1 ausführliche Einbauanleitung
- Kit SK128/384, SK128/576, SK320/576, 384/576
- 1 Satz ICs
  - 1 Systemdiskette
  - 1 ausführliche Einbauanleitung

Preise:

SP64 = 225,-- DM  
 SP256 = 533,-- DM  
 SP320 = 573,-- DM  
 SP512 = 881,-- DM

Diese RAM-Karten sind bei der Vortex Vertriebs GmbH, Klingenberg 13, 7106 Neuenstadt 5 erhältlich.

# ELEPHANT™ KOMPROMISSLOS ERSTE WAHL.



"NEVER FORGETS."



Speichern und Drucken. Fehlerlos. Mit 100% Sicherheit:  
das ist Elephant Memory Systems™.

Unter diesem Markennamen erhalten Sie hochwertiges Zubehör  
für perfekt arbeitende Kommunikations-Systeme der Text-  
und Datenverarbeitung – wie Disketten, Farbbänder, Farbband-  
kassetten und Spezialreinigungsmittel.

Kompromiß ist Risiko. Zweite Wahl ist Unsicherheit – deshalb:  
verlangen Sie Elephant Memory Systems™.

Dennison

## ELEPHANT™ NEVER FORGETS.

MARCOM Computerzubehör GmbH

Podbielskistraße 321, 3000 Hannover 51, Tel. (05 11) 64 74 20

Frankreich: Soroclass, 8, Rue Montgolfier - 93115, Rosny-Sous-Bois, Tel.: 16 (1) 855-73-70

Großbritannien: Dennison Mfg. Co. Ltd., Colonial Way, Watford WD2 4JY, Tel.: 0923 41244, Telex: 923321

Italien: King mec s.p.a. - Via Regio Parco 108 - 10036 Settimo Torinese

Weiteres Ausland: Dennison International Company, 4006 Erkrath 1, Matthias-Claudius-Straße 9, Telex: 858 6600

Compute mit