

SA 1/86

# Compute mit

öS 55 sfr 6,50 DM 6,50

## CE16

Heimcomputer

CE16 / 116

# SONDER- Ausgabe

- *Hardware-Berichte*
- *Buchbesprechungen*

über 15 Programme

- *Werkstatt*
- *Tips & Tricks*
- *Software-Reviews*

Pac-Man – Jump-Man – Poker – Cowboy-Duell – Mover – Rocketman  
Tank-Duell – Turmspringen – Ligatabelle – Super-Dat – Window-Manager  
Softscroll – POKE-Trickkiste – Multigraphic – und vieles mehr

# inhalt

## GAMES

Cowboy-Duell	3
Mover	6
Freß-Man	8
Jump Man	10
Poker	12
Turmspringen	16
Tank-Wars	20
Rocketman	23

<b>Software-Reviews</b>	28
-------------------------	----

### Bücher:

C-16 für Einsteiger	34
Der Commodore 16 in der Praxis	35

### Hardware:

Schlafender Riese im Zwergenlook	36
-------------------------------------	----

## TIPS & TRICKS

Comfort-List	37
Berechnung einer Prüfsumme für MC-Programme	38
Old- und Merge-Routine	38
POKE-Trickkiste	39

## WERKSTATT

Window-Manager	42
Grafik-Beispiele	44
Softscroll	44

## ANWENDER-PROGRAMME

Ligatabelle	47
Superdat	50
Multigrafik	54

### Impressum

„Compute mit – C-16/116 Sonderheft“  
erscheint monatlich im Tronic-Verlag, 3440 Eschwege

#### Redaktion:

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

#### Gesamtherstellung:

Druckhaus Dierichs Kassel, Frankfurter Str. 168, 3500 Kassel

#### Vertrieb:

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

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

#### Erscheinungsweise:

Erstverkaufstag vom „Compute mit – Sonderheft“ 1. April 1986

#### Urheberrecht:

Alle im „Compute mit – Sonderheft“ veröffentlichten Beiträge  
sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzun-  
gen, 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 5,80 DM  
ohne Kassetten!

#### Autoren, Manuskripte:

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

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

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

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

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

#### Anzeigenpreise:

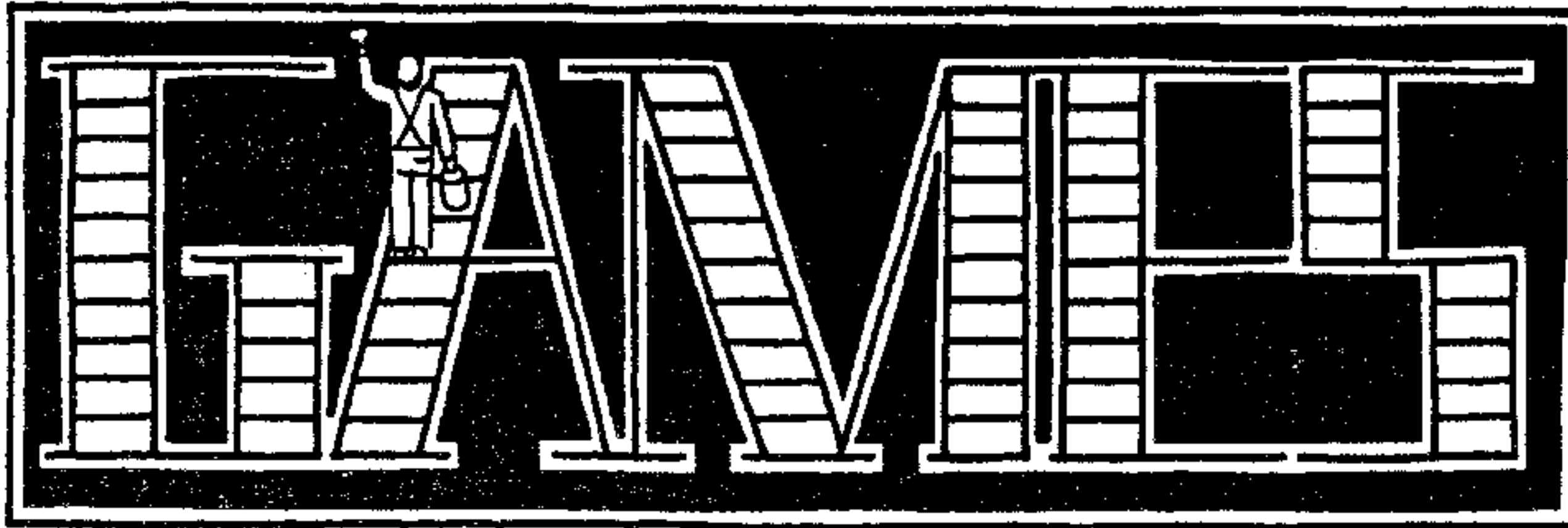
Bitte Mediaunterlagen anfordern.

#### Anzeigenverwaltung:

Tronic-Verlag GmbH,  
Am Stad 35, 3440 Eschwege  
Telefon 0 56 51 / 3 00 11  
Telefax 0 56 51 / 3 00 14

#### Anzeigenleitung:

Inland: H. Wendt  
Ausland: M. Kleimann



## „Cowboy-Duell“

## unter gleißender Sonne in der Wüste von Nevada

Für alle Western-Fans bietet der C-16-Computer diesen klassischen Revolver-Zweikampf nach alter „High Noon-Manier“. Also auf geht's! – Laßt die „blauen Bohnen“ fliegen.

Zu Beginn des Programms ist die Entscheidung zu treffen, ob gegen den Computer oder einen 2. Spieler angetreten werden soll (Joystick nach links oder rechts bewegen). Anschließend bitte

den Feuerknopf betätigen.

Und schon befinden Sie sich mitten in der glühend heißen Wüste Nevadas. Zwischen vereinzelt Kakteen taucht sofort der Gegner vor Ihnen auf und ehe Sie sich versehen, eröffnet dieser das Feuer. Also volle Deckung und keine Kugel schuldig bleiben (Joystick-Feuertaste betätigen). Durch Hochspringen oder sich Niederwerfen kann man den feindlichen Kugeln ausweichen (Joystick nach oben bzw. nach unten bewegen).

Derjenige, der seinen Gegenüber zuerst 3-mal „erwischt“ hat, kann eine weitere Kerbe in seinen „Peacemaker“ einritzen.

```

1 REM *****
2 REM *
3 REM *           (C) 1985
4 REM *           BY
5 REM *           JAN-PETER GUZIK
6 REM *
7 REM *
8 REM *****
10 POKE56,47:CLR:FORT=832TO849:READA:POKET,A:NEXT:SYS832
20 FORT=1280TO13167:READA:POKET,A:NEXT
30 V=65280
40 POKEV+18,PEEK(V+18)AND251
50 POKEV+19,PEEK(V+19)AND30R48
60 VOL8:COLOR0,1:COLOR4,1:C1=3:C2=3:T1=3677:T2=3703:GOSUB1790:GOSUB1470
70 REM ***** HAUPTPROGRAMM *****
80 ONJ1GOTO190,90,90,90,400
90 J1=JOY(1):ONJ1GOTO190,100,100,100,400
100 IFS1=2THEN340
110 IFJ1>127THENS1=2:SU=B+2:SOUND3,740,3:POKEB+1,82:POKEB+41,83:GOTO340
120 FORT=1TO30:NEXT
130 ONJ2GOTO510,140,140,140,730
140 GOSUB1720:ONJ2GOTO510,150,150,150,730
150 IFS2=2THEN670
160 IFJ2>127THENS2=2:KU=V-1:SOUND3,440,3:POKEV,85:POKEV+40,86:GOTO670
170 FORT=1TO30:NEXT:GOTO80
180 REM ***** SPRUNG SPIELER 1 *****
190 H1=H1+1:ONH1GOTO200,210,220,230,240,250
200 LO=B+80:B=B-40:GOTO260
210 LO=B+80:B=B-40:GOTO260
220 LO=B+80:B=B-40:GOTO260
230 LO=B:B=B+40:GOTO270
240 LO=B:B=B+40:GOTO270
250 LO=B:B=B+40:J1=JOY(1):H1=0:GOTO270
260 PP=PEEK(B+1):GOTO280
270 PP=PEEK(B+81)

```

# programme

```
280 POKEB,64:POKEB+1,65:POKEB+40,66:POKEB+41,67:POKEB+80,68:POKEB+81,69
290 POKELO,32:POKELO+1,32:IFPP=76THEN850
300 IFS1=2THEN340
310 IFJOY(1)>127THENS1=2:SU=B+2:SOUND3,740,3:POKEB+1,82:POKEB+41,83:GOTO340
320 GOTO130
330 REM ***** SCHUSS SPIELER 1 *****
340 PP=PEEK(SU):POKESU-1024,112:POKESU,76:POKESU-6,32:POKESU-1030,54
350 IFPP<>32THEN890
360 IFL1=5THENS1=0:L1=0:POKESU,32:GOTO130
370 SU=SU+6:L1=L1+1
380 GOTO130
390 REM ***** DUCKEN SPIELR 1 *****
400 D1=D1+1:OND1GOTO410,430,440
410 POKEB+40,64:POKEB+41,65:POKEB+80,66:POKEB+81,78:POKEB+82,79:POKEB,32
420 POKEB+1,32:GOTO470
430 FORT=1TO20:NEXT:GOTO470
440 PP=PEEK(B+1):POKEB,64:POKEB+1,65:POKEB+40,66:POKEB+41,67:POKEB+80,68:POKEB+8
2,32
450 POKEB+81,69:D1=0:IFPP=76THEN850
460 J1=JOY(1):GOTO100
470 IFS1=2THEN340
480 IFJOY(1)>127THENS1=2:SU=B+42:SOUND3,740,3:POKEB+41,82:POKEB+81,84:GOTO340
490 GOTO130
500 REM ***** SPRUNG SPIELER 2 *****
510 H2=H2+1:ONH2GOTO520,530,540,550,560,570
520 LO=V+80:V=V-40:GOTO580
530 LO=V+80:V=V-40:GOTO580
540 LO=V+80:V=V-40:GOTO580
550 LO=V:V=V+40:GOTO590
560 LO=V:V=V+40:GOTO590
570 LO=V:V=V+40:GOSUB1720:H2=0:GOTO590
580 PP=PEEK(V):GOTO600
590 PP=PEEK(V+80)

600 POKEV,70:POKEV+1,71:POKEV+40,72:POKEV+41,73:POKEV+80,74:POKEV+81,75
610 POKELO,32:POKELO+1,32:IFPP=76THEN890
620 IFS2=2THEN670
630 GOSUB1690
640 IFJJ>127THENS2=2:KU=V-1:SOUND3,440,3:POKEV,85:POKEV+40,86:GOTO670
650 GOTO80
660 REM ***** SCHUSS SPIELER 2 *****
670 PP=PEEK(KU):POKEKU-1024,54:POKEKU,76:POKEKU+6,32:POKEKU-1018,112
680 IFPP<>32THEN850
690 IFL2=5THENS2=0:L2=0:POKEKU,32:GOTO80
700 KU=KU-6:L2=L2+1
710 GOTO80
720 REM ***** DUCKEN SPIELR 2 *****
730 D2=D2+1:OND2GOTO740,760,770
740 POKEV+40,70:POKEV+41,71:POKEV+80,80:POKEV+81,73:POKEV+79,81:POKEV,32
750 POKEV+1,32:GOTO800
760 FORT=1TO20:NEXT:GOTO800
770 PP=PEEK(V):POKEV,70:POKEV+1,71:POKEV+40,72:POKEV+41,73:POKEV+80,74:POKEV+79,
32
780 POKEV+81,75:D2=0:IFPP=76THEN890
790 GOSUB1720:GOTO150
800 IFS2=2THEN670
810 GOSUB1690
820 IFJJ>127THENS2=2:KU=V+39:SOUND3,440,3:POKEV+80,87:POKEV+40,85:GOTO670
830 GOTO80
840 REM ***** GETROFFEN *****
850 C1=C1-1:POKEB,32:POKEB+1,32:POKEB+40,32:POKEB+41,32:POKEB+80,32:POKEB+81,32
860 POKE2654,112:POKE2655,112
870 POKET1,98:POKET1+40,99:POKET1+1,100:POKET1+41,101:POKET1+2,102:POKET1+42,103
880 GOTO920
890 C2=C2-1:POKEV,32:POKEV+1,32:POKEV+40,32:POKEV+41,32:POKEV+80,32:POKEV+81,32
```

```

900 POKE2678,54:POKE2679,54
910 POKET2,104:POKET2+40,105:POKET2+1,106:POKET2+41,107:POKET2-1,108:POKET2+39,1
09
920 L=8:FORT=800TO600STEP-20:L=L-.6:VOLL:SOUND1,T,5:SOUND2,T-20,3:NEXT
930 IFC1=0ORC2=0THEN960
940 S1=0:S2=0:H1=0:H2=0:D1=0:D2=0:PF=0:J1=0:J2=0:L1=0:L2=0:GOSUB1470:VOL8:GOTO80
950 REM ***** GAME OVER *****
960 IFC1>C2THENCHAR1,1,23,"*||SIEGER*":ELSECOLOR1,7,3:CHAR1,29,23,"*||SIEGER*"
970 FORT=1TO7:CHAR1,15,T,"GAME OVER":COLOR1,15,T:CHAR1,15,T-1," "
980 VOL8:FORTT=1TO10:SOUND1,14*TT*T,2:NEXT:NEXT:RUN60
990 REM *** DATEN FUER ZEICHENSATZ ***
1000 DATA162,0,189,0,208,157,0,48,189,0,209,157,0,49,202,208,241,96
1010 DATA0,7,7,31,7,7,3,7
1020 DATA0,192,192,240,64,64,128,128
1030 DATA15,28,28,27,27,28,14,3
1040 DATA192,224,240,240,240,240,48,224
1050 DATA15,14,28,28,28,56,62,62
1060 DATA224,112,56,56,56,56,31,31
1070 DATA0,3,3,15,2,2,1,1
1080 DATA0,224,224,248,224,224,192,224
1090 DATA3,7,15,15,15,15,12,7
1100 DATA240,56,56,152,152,56,112,192
1110 DATA14,28,28,28,28,28,248,248
1120 DATA240,112,56,56,56,28,124,124
1130 DATA0,0,0,24,24,0,0,0
1140 DATA255,255,255,255,255,255,255,255
1150 DATA192,224,240,240,240,255,63,255
1160 DATA0,0,0,12,12,252,252,252
1170 DATA3,7,15,15,15,255,252,255
1180 DATA0,0,0,48,48,63,63,63
1190 DATA0,192,192,240,64,64,128,159
1200 DATA248,248,240,240,240,240,48,224
1210 DATA248,248,240,240,240,255,63,255
1220 DATA0,3,3,15,2,2,1,249
1230 DATA31,31,15,15,15,15,12,7
1240 DATA31,31,15,15,15,255,252,255
1250 DATA15,31,31,15,0,0,0,0
1260 DATA28,62,62,62,62,62,62,62
1270 DATA255,255,255,255,62,62,62,62
1280 DATA0,0,0,24,60,60,60,60
1290 DATA252,252,248,240,0,0,0,0
1300 DATA64,224,240,252,254,255,255,255
1310 DATA0,28,62,126,255,255,255,255
1320 DATA0,0,0,24,60,255,255,255
1330 DATA60,66,153,161,161,153,66,60
1340 DATA64,96,80,72,38,22,14,6
1350 DATA0,0,0,0,16,16,124,115
1360 DATA127,127,126,16,16,0,0,0
1370 DATA0,0,0,0,62,127,253,253
1380 DATA153,131,230,254,124,0,0,0
1390 DATA0,0,0,0,3,3,3,255
1400 DATA255,255,0,0,0,0,0,0
1410 DATA0,0,0,0,124,254,191,191
1420 DATA153,193,103,127,62,0,0,0
1430 DATA0,0,0,0,8,8,62,206
1440 DATA254,254,190,8,8,0,0,0
1450 DATA0,0,0,0,192,192,192,255
1460 DATA255,255,0,0,0,0,0,0
1470 REM ***** SPIEFELD *****
1480 SCNCLR:FORT=1TO7:COLOR1,15,T:PRINT"
/:NEXT
1490 COLOR0,8,6:COLOR4,3,3:COLOR1,10,1
1500 PRINT"π▼ππ|||π▼||π▼π▼|||π▼π|||π▼ππ|||π▼ππ▼":COLOR1,6,1
1510 PRINT"
|+"
1520 PRINT"
|+
1530 PRINT"
|+
1540 PRINT"
|+
1550 PRINT"
1560 PRINT"
|+
1570 PRINT"
|+
1580 PRINT"
|+
1590 PRINT"
|+
1600 PRINT"
|+
1610 B=3637:V=3663
1620 FORT=2493TO2693STEP40:POKET,112:POKET+1,112:NEXT:POKE2695,112
1630 FORT=2519TO2719STEP40:POKET,54:POKET+1,54:NEXT:POKE2718,54
1640 POKEB,64:POKEB+1,65:POKEB+40,66:POKEB+41,67:POKEB+80,68:POKEB+81,69
1650 POKEV,70:POKEV+1,71:POKEV+40,72:POKEV+41,73:POKEV+80,74:POKEV+81,75
1660 CHAR1,1,23,"LEBEN: ":PRINTC1:COLOR1,7,3:CHAR1,29,23,"LEBEN: ":PRINTC2
1670 RETURN
1680 REM ***** COMPUTER *****
1690 ONSPGOTO1710,1700
1700 JJ=JOY(2):RETURN
1710 JJ=128:RETURN
1720 ONSPGOTO1740,1730

```

# programme

```

1730 J2=JOY(2):RETURN
1740 IFPEEK(3657)=76THENJ2=5:RETURN
1750 IFPEEK(3685)=76THENJ2=1:RETURN
1760 IFS1=0THENJ2=INT(RND(1)*5)+1:RETURN
1770 J2=128:RETURN
1780 REM ***** TITELBILD *****
1790 SCNCLR
1800 COLOR1,2,4
1810 PRINT"  "
1820 PRINT"  "
1830 PRINT"  "
1840 PRINT"  "
1850 PRINT"  "
1860 PRINT"  "
1870 PRINT"  "
1880 PRINT"  "
1890 PRINT"  "
1900 PRINT"  "
1910 PRINT"  "
1920 COLOR1,7,5:CHAR1,8,21," 1985 BY JAN-PETER GUZIK":POKE3920,96
1930 GOSUB2020
1940 COLOR1,8:CHAR1,10,16,"PRESS FIRE TO START"
1950 COLOR1,10,4:CHAR1,5,13,"1 SPIELER":CHAR1,25,13,"":SP=1:GOTO1970
1960 CHAR1,5,13,"":CHAR1,25,13,"2 SPIELER":SP=2
1970 IF(JOY(1)ORJOY(2))=3THEN1960
1980 IF(JOY(1)ORJOY(2))=7THEN1950
1990 IF(JOY(1)ORJOY(2))>127THENRETURN
2000 GOTO1970
2010 REM ***** LIED *****
2020 RESTORE2040
2030 FORT=1TO15:READA,B:SOUND1,A,B:NEXT:RETURN
2040 DATA685,20,739,10,770,10,685,20,739,10,770,10
2050 DATA739,30,770,10,810,40,739,20,681,10,643,10,685,20,739,20,770,40

```

*Noch wenige Programmzeilen sind einzugeben, dann können Sie Ihren Colt ziehen und die „blauen Bohnen“ fliegen lassen!*

## „MOVER“ ist ein schnelles Labyrinthspiel

*Bei diesem Programmbeitrag handelt es sich um eine Ableitung der bekannten Pac-Man-Version.*

Sie befinden sich in einem riesigen, mit zahlreichen Winkeln und Nischen versehenen Irrgarten. Ihre Aufgabe besteht darin, die im gesamten Labyrinth verstreuten Punkte einzusammeln. Dies scheint zu Beginn ein leichtes Unterfangen zu sein, doch dann taucht ein Gespenst namens „MOVER“ auf. Mover wirkt anfangs vollkommen harmlos bis desinteressiert und läßt Sie ohne Störung bei Ihrer „Punktefresserei“ gewähren. Das Ganze ändert sich schlagartig, nachdem Sie eine bestimmte Anzahl von Punk-

ten vertilgt haben. Nun setzt Mover zur Verfolgungsjagd auf Sie an, und er ist rasend schnell. Die einzige Chance, dem Verschlingen durch Mover zu entgehen, bieten die drei im Labyrinth versteckten Herzen. Wenn es Ihnen gelingt, eines der Herzen zu vertilgen, wird das Gespenst für kurze Zeit auf seiner Verfolgungsjagd zurückgeworfen. Mover ist nicht nur rasend schnell, sondern besitzt zudem noch die Fähigkeit, für einige Sekunden unsichtbar zu werden. Aber lassen Sie sich auf Ihrer „Punkte-

hatz“ nicht entmutigen und zeigen Sie Mover die Fersen! Falls Sie nach dem Verlust Ihres Lebens noch einmal spielen wollen, betätigen Sie bitte die Return-Taste, und anschließend können Sie zwischen Joystick- oder Tastatursteuerung wählen.

- Joystick in Port.1 anschließen -
- Tastatur: J = nach oben
- M = nach unten
- . = nach rechts
- INS/DEL = nach links

```

1 REM *** MOVER ***
4 REM
5 REM (C) BY HOLGER KRAUL
6 REM
7 X=0:Y=0:J=0:I=0:P=0:FA=0
8 A=198:L=47:D=44:E=36:F=40:H=46:K=260:L=113:M=72:N=83:O=810:Q=87:R=900
9 GOTO1000
10 S=50:SC=0:RU=1
50 GOSUB5000:GOSUB230
100 BS=3072:FA=BS-2048:I=RND(-TI):MO=162
110 GOSUB200:X=B:POKEX,81:POKEX-FA,72
120 GOSUB200:Y=B:POKEY,MO:POKEY-FA,50
130 PU=0:VOL7:PC=87:T=1:PO=46:YM=1:XM=0:XT=0:Z=50:H2=102:TN=1005
199 GOTO400
200 DO:B=BS+INT(RND(1)*20)+10+40*(INT(RND(1)*22)+1):P=PEEK(B)
210 LOOPWHILEP=102ORP=83ORP=81ORP=87
220 RETURN
230 PRINT"SPUNKTE : "SC+PUTAB(15)"RUNDE: "RUTAB(28)"HI: "HI:RETURN
400 DO: IFX=YTHENEXIT:LOOP:GOTO700
410 IFJTHEN440
420 I=PEEK(A): IFI=CTHENXT=-1
422 IFI=DTHENXT=1
424 IFI=ETHENXT=F
426 IFI=34THENXT=-F
428 GOTO450
440 P=JOY(1): IFP=7THENXT=-1
442 IFP=3THENXT=1
444 IFP=5THENXT=F
446 IFP=1THENXT=-F
450 IFPEEK(X+XT)<>H2THENXM=XT
460 P=PEEK(X+XM): IFP=H2THENS10

480 IFP=HTHENPU=PU+1:SOUND3,TN,1:IFPU=KTHENEXIT:LOOP:GOTO800
490 POKEX-FA,L:POKEX,XM+F:X=X+XM:POKEX,PC:POKEX-FA,M
492 IFP=NTHENEXIT:LOOP:GOTO900
495 IFX=YTHENEXIT:LOOP:GOTO700
510 IFT=1THENS60
520 SOUND1,R,2:POKEY,32:Y=Y+YM:YM=PEEK(Y)-F:POKEY-FA,Z:POKEY,MO
530 IFABS(YM)<>FTHENIFABS(YM)<>1THENSOUND1,800,5:T=1:PO=YM+F:GOTO600
550 LOOP
560 P=PEEK(Y+YM): IFP=H2THEN600
570 POKEY,PO:POKEY-FA,0:Y=Y+YM:POKEY-FA,Z:POKEY,MO:PO=P
580 I=ABS(P-F): IFI=1ORI=FTHENT=0:YM=P-F:PO=32

590 LOOP
600 P=INT(RND(1)*4): IFP=0THENYM=-1
605 IFZ=50THENZ=113:ELSEZ=50
610 IFP=1THENYM=1
620 IFP=2THENYM=F
630 IFP=3THENYM=-F
640 LOOP
700 SC=SC+PU:PU=0: IFSC>HITHENHI=SC
705 PRINTCHR$(27)"W"CHR$(27)"V";
710 GOSUB230:POKEX-FA,50
720 VOL8:FORI=1020TO0STEP-10:SOUND3,I,1
730 POKEX,PC: IFPC=81THENPC=160:ELSEPC=81
740 NEXT:POKEX,MO:POKEX-FA,0
750 PRINT"ERWISCHT!"

760 PRINT"NEUES":PRINT"SPIEL":PRINT"MIT":PRINT"RETURN"
780 DO:GETA$:LOOPUNTILA$=CHR$(13)
790 GOTO1000
800 GOSUB230:SC=SC+PU:PU=0
810 FORI=1TO20:SOUND1,500,2:SOUND1,700,1:SOUND1,750,2:NEXT
850 PRINT"RUNDE":PRINT"GESCHAFFT":PRINT"RETURN"
860 DO:GETA$:LOOP UNTILA$=CHR$(13)
870 RU=RU+1:S=S+50:GOTO50
900 POKEY-FA,50:SOUND1,560,3:FORI=1TO100:NEXT:POKEY-FA,0:POKEY,PO
905 GOSUB230
910 GOSUB200:PO=P:Y=B:POKEY,MO:POKEY-FA,50:Z=50
920 IFP=39ORP=41ORP=0ORP=80THENT=0:YM=P-40:PO=32
930 FORI=1TO100:NEXT:GOTO400

```

# programme

```

1000 COLOR4,12,5:COLOR0,2:GOSUB5000:PRINT"MOVER":PRINT"
1005 PRINT"SPIEL"
1010 PRINT"MIT":PRINT"JOYSTICK"
1020 PRINT"ODER":PRINT"ASTATUR":PRINT"J,.,."

```

```

1030 PRINT"(J/T):";
1040 DO:GETA$:LOOPUNTIL A$="J"ORA$="T"
1050 PRINTA$";:J=0:IFA$="J"THENJ=1
1080 FORI=1TO400:NEXT:GOTO10
5000 PRINT"TAB(9)CHR$(27)"T"
5010 PRINT"
5020 PRINT"
5030 PRINT"
5040 PRINT"
5050 PRINT"
5060 PRINT"
5070 PRINT"
5080 PRINT"
5090 PRINT"
5100 PRINT"

```

```

5110 PRINT"
5120 PRINT"
5130 PRINT"
5140 PRINT"
5150 PRINT"
5160 PRINT"
5170 PRINT"
5180 PRINT"
5190 PRINT"
5200 PRINT"
5210 PRINT"
5220 PRINT"
5230 PRINT"
5310 RETURN

```



## „Freß-Man“

- Ein Spiel nach dem Motto „Fressen oder gefressen werden!“

Sie befinden sich in einem Labyrinth und versuchen, die in den Gängen und Winkeln versteckten Punkte einzusammeln. Bei Ihrer Punktejagd bleiben Sie natürlich nicht ungestört, denn die im Labyrinth herumrasenden Monster stellen eine äußerst ernsthafte Bedrohung für Ihr Leben dar. Falls Sie mehr als 3mal mit den Monstern kollidieren, ist Ihr Spielerdasein beendet.

Für Sie selbst besteht ebenfalls die Möglichkeit, den Monstern den Garaus zu machen. Falls es Ihnen gelingt, einen der im Irrgarten versteckten Äpfel zu vertilgen, verwandeln sich die Monster kurzzeitig in Kreuze. Innerhalb weniger Sekunden können Sie nun selbst Ihren Feinden zu Leibe rücken. Jedes gefressene Monster be-

schert Ihnen entscheidende Punktgutschriften.

### Eingabetips:

Nach dem Eintippen das Programm bitte absaven und dann mit RUN starten. Unter Umständen kann sonst die Datasette während des Absavens einige Schwierigkei-

ten bereiten. Falls sich im Programm ein Fehler befindet, erscheint ein unerklärliches Wirrwarr auf dem Bildschirm. Um diesen Zustand zu beheben, bitte gleichzeitig RUNSTOP-Taste und RESET-Taste drücken und dann die RESET-Taste zuerst loslassen. Anschließend „X“ und RETURN eingeben.

```

10 REM *****
20 REM ** PAC-MAN **
30 REM **
40 REM ** (C) BY **
50 REM **

```

```

60 REM ** ANDREAS LUPERT **
70 REM *****
80 COLOR0,1:COLOR4,1:GOSUB930:GOTO100
90 FRINTHTAB(16)E:RETURN
100 SCNCLR:VOLS

```

```

110 PRINT"
120 PRINT"
130 PRINT"
140 PRINT"
150 PRINT"
160 PRINT"

```

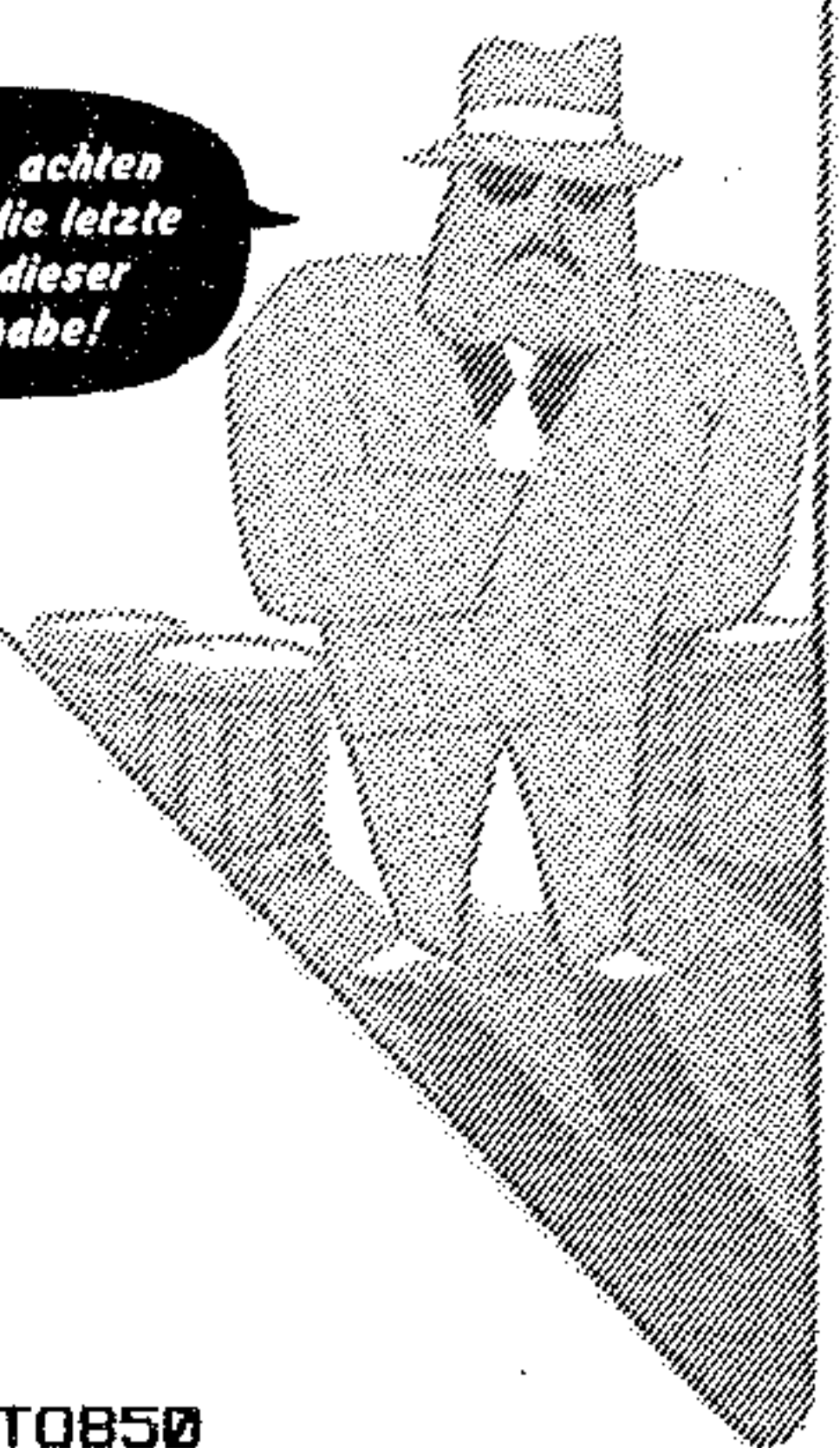


```

170 PRINT"
180 PRINT"
190 PRINT"
200 PRINT"
210 PRINT"
220 PRINT"
230 PRINT"
240 PRINT"
250 PRINT"
260 PRINT"
270 PRINT"
280 PRINT"
290 PRINT"
300 PRINT"
310 PRINT"
320 PRINT"
330 PRINT"
340 PRINT"
350 FORI=0TO39:POKE3072+960+I,67:NEXT
360 H$="50000000000000000000"
370 QQ=INT(RND(0)*40)+1
380 ZX=INT(RND(0)*20)*40+3072+QQ:IFPEEK(ZX)<>66THENGOTO370
390 P=9:A$(1)="+1000":A$(2)="+1500":A$(3)="+2000":A$(4)="+3000"
400 POKEZX,71:QQ=3144:WW=3232:
410 FORGT=1TO3:G1=3811:G2=3434:Y2(1)=1:Y2(2)=-1:Y2(3)=40:Y2(4)=-40:U1=3:G3=3374:
Y3(1)=1
420 Y3(2)=-1
430 Y3(3)=40:Y3(4)=-40:U2=3:A=3374-3:C=64:BN=65
440 FORQ=1TO71:READX,Y1:FORO=1TOX:POKEG1,32:G1=G1+Y1
450 POKEG1,BN:POKEG2,66
460 G2=G2+Y2(U1)
470 IFU1=<1THENGOTO600
480 IFPEEK(G2)=67THENG2=G2-Y2(U1):U1=INT(RND(0)*4)+1
490 POKEG2,BN:POKEG3,66:G3=G3+Y3(U2)
500 IFPEEK(G3)=67THENG3=G3-Y3(U2):U2=INT(RND(0)*4)+1
510 POKEG3,BN:IFG1=AORG2=AORG3=ATHEN870
520 GETB$:IFLEN(B$)>0THENA$=B$
530 IFA$="4"THENPOKEA,32:A=A+1:B=-1:C=64:GOTO570
540 IFA$="A"THENPOKEA,32:A=A-1:B=1:C=64:GOTO570
550 IFA$="W"THENPOKEA,32:A=A+40:B=-40:C=64:GOTO570
560 IFA$="3"THENPOKEA,32:A=A-40:B=40:C=64
570 REM
580 GOTO660
590 POKEA,C:NEXTO,Q:RESTORE1150:GOTO440
600 IFPEEK(G2+40)<>67THENU1=3:BN=65:GOTO480
610 IFPEEK(G2-40)<>67THENU1=4:GOTO630
620 GOTO480
630 IFS=0THENIFRND(0)*10>0THENPOKE3497,INT(RND(0)*3)+68:S=1:GOTO850
640 IFSTHENS=S+1:IFS=4THENPOKE3497,32:S=0
650 GOTO480
660 IFPEEK(A)<>67THENGOTO680
670 A=A+B:A$="":GOTO590
680 IFPEEK(A)=66THENPOKETR,16:E=E+10:GOSUB90:VOL8:SOUND3,1020,3:GOTO590
690 IFPEEK(A)=68THENE=E+1000:GOSUB90:I=1:GOTO760
700 IFPEEK(A)=69THENE=E+1500:GOSUB90:I=2:GOTO760
710 IFPEEK(A)=70THENE=E+2000:GOSUB90:I=3:GOTO760
720 IFPEEK(A)=71THENE=E+200:GOSUB90:BN=72:VOL8:SOUND3,1010,20:GOTO590
730 IFPEEK(A)=72THENE=E+3000:GOSUB90:I=4:GOTO760
740 IFPEEK(A)=65THEN870
750 GOTO590

```

... psst, achten Sie auf die letzte Seite dieser Ausgabe!







# programme

```

550 DATA17,58,52,88,144,44,66,130
560 DATA16,56,48,88,148,58,68,132
570 DATA0,247,247,247,0,254,254,254
580 DATA255,0,255,0,255,0,255,0
590 DATA60,66,153,161,161,153,66,60
600 DATA0,0,34,120,170,34,34,34
    
```

```

610 DATA0,2,6,6,6,38,62,28
620 DATA0,120,124,102,124,120,96,96
630 DATA0,66,102,102,102,102,126,60
640 DATA0,126,102,102,126,126,102,102
650 DATA0,66,102,126,102,102,102,102
660 DATA0,102,116,110,110,102,102,102
    
```

## „Poker“



## Wenn „Bluffen“ Ihre Stärke ist, sind Sie hier richtig!

Wer kennt Sie nicht, die berühmte Filmszene aus so manchem Westernklassiker: Ein verräucherter Saloon, ein großer runder Tisch, der mit den unterschiedlichsten Spielertypen besetzt und von zahlreichen Kibitzen umlagert ist.

Setzen Sie Ihr Pokerface auf und nehmen Sie an dieser Tischrunde zu einer Partie Poker Platz. Unser Programmbeitrag bietet Ihnen die Möglichkeit, Ihre Kartenkünste unter Beweis zu stellen. (S.G.)

Abtippen – Speichern – Mit „RUN“ starten. Nachdem dies erfolgt ist, den eigenen Vornamen und die Namen der Mitspieler (maximal 9) eingeben. Die Mitspieler werden vom Computer simuliert. Anschließend erscheinen die Standardwerte, die, falls erwünscht (mit „J“ beantworten), gleich angezeigt werden.

Möchte man die Werte selbst bestimmen

(„N“ eingeben), erscheinen die Kriterien und die entsprechenden Zahlen müssen nun eingegeben werden. Ist die Eingabe abgeschlossen, können Sie auf dem Screen das Spielbild und eine Tabelle, die alle wichtigen Informationen enthält, erkennen. Poker wird nach den bekannten Regeln gespielt. Also auf geht's nach dem Motto „Gut geblufft ist halb gewonnen!“

```

10 REM *****
20 REM *      P O K E R      *
30 REM *
40 REM *      WRITTEN 1986  *
50 REM *      BY          *
60 REM *      THORSTEN FIEN *
70 REM *
80 REM *      3118 BAD BEVENSEN *
90 REM *****
    
```

```

100 COLOR 0,1:COLOR 4,1
110 FORI=1TO9:READA:NEXT
120 DIMN$(9):FORI=1TO9:READN$(I):NEXT
130 FORI=1TO8:J=INT(RND(1)*(9-I))+I+1
140 N$(J)=N$(I):N$(I)=N$(J):N$(J)=N$:NEXT
150 DATA24E5,18E5,42E4,68E3,23E3,9514,
4418,674,52
160 INPUT"HALLO, WIE HEISST DU DENN";N$
170 IFLEN(N$)>5THENN$=LEFT$(N$,4)+". "
180 IFLEN(N$)<5THENN$=N$+" ":GOTO180
    
```

```

190 INPUT"OWIE VIELE MITSPIELER BRAUCHST DU";AN:A1=AN+1:IFAN<10RAN>9THEN190
200 PRINT"MOECHTEST DU DIE MITSPIELER SELBST AUSWAELLEN?"
210 GETYY$:IFYY$=""THEN210
220 IFYY$<>"J"THEN280
230 FORYY=1TOAN:PRINT"NAME DES";YY;". MITSPIELERS"
240 INPUT" ";N$(YY)
250 IFLEN(N$(YY))>5THENN$(YY)=LEFT$(N$(YY),4)+". "
260 IFLEN(N$(YY))<5THENN$(YY)=N$(YY)+" ":GOTO250
    
```

```

270 NEXT
280 DATAJULIA,FRITZ,SILKE,"LENE ","ERIK ","PETER","TINE ","HANS ","JOHN "
290 :DIMAZ(AN),BZ(AN),CZ(AN),EZ(AN),FZ(AN),KZ(AN,4),LZ(4),WZ(AN)
300 PRINT"@STANDARTWERTE ?"
310 GETYY$:IFYY$=""THEN310
320 IFYY$<>"J"THEN390
330 AV=2000:GR=100:KO=50:LI=1000
340 PRINT"@ANFANGSVERMOEGEN: 2000 $"
350 PRINT"@GRUNDEINSATZ : 100 $"
360 PRINT"@KOSTEN PRO KARTE: 50 $"
370 PRINT"@LIMIT : 1000 $"
380 GOTO440
390 INPUT"@ANFANGSVERMOEGEN: ";AV
400 INPUT"@GRUNDEINSATZ : ";GR
410 INPUT"@KOSTEN PRO KARTE: ";KO
420 INPUT"@LIMIT : ";LI
430 KO=INT(KO):LI=INT(LI)

440 AV=INT(AV):GR=INT(GR):IFAV<LIORLI=<GRORGR<0ORKO>LIORKO<0ORAV>32767/A1THEN460
450 GOTO490
460 PRINT"@FALSCH EINGABE! NOCHMAL EINGEBEN!"
470 IFTI-T<60THEN470:NEXT
480 PRINT"@":GOTO390
490 FORI=0TOAN:AZ(I)=AV:NEXT:GE=INT(RND(-RND(-1/TI))*A1)
500 L$="" :L$=L$+L$+L$+"":L$="S"+L$+L$+" "
510 HI=GR+(SGR(AV*LI)-GR)/10
520 DEFFNC(X)=X-5*INT(X/5)
530 DEFFNA(X)=INT(X/4)
540 DEFFNB(X)=X-4*FNA(X)
550 DEFFNJ(X)=X+A1*(X>AN)
560 DEFFNP(X)=(I-X=2)OR(I-X=3)
570 S$="":FORJ=0TO51:S$=S$+CHR$(J):NEXT
580 FORJ=0TOAN:CZ(J)=0:AZ(J)=AZ(J)-GR:BZ(J)=GR:NEXT:CZ(0)=1:PO=A1*GR
590 IFAZ(0)<0THEN2110
600 IFAZ(1)<0ANDAN=1THEN2130
610 FORJ=1TOAN:IFAZ(J)=>0THENNEXT:GOTO660
620 PO=PO+AZ(J):PRINTN$(J)" SCHEIDET AUS.
630 IFJ<>ANTHENPRINT"|| UND TAUSCHT SEINEN PLATZ MIT "N$(AN)
640 AZ(J)=AZ(AN):A1=AN:AN=AN-1:N$(J)=N$(A1)
650 GOTO600
660 FORJ=0TOAN:FORI=0TO4:GOSUB2140:KZ(J,I)=A:NEXT:NEXT
670 GOSUB1520:PRINT"
680 GOSUB1360:FORJ=GE+1TOGE+A1:Q=FNJ(J):PRINTL$: :IF0THEN840
690 INPUT"SO WIE VIELE KARTEN WILLST DU "
700 YY=A:A=ABS(A)
710 IFA<0ORA>4THEN690
720 IFLEN(S$)<ATHENPRINT"SO ES SIND NUR NOCH"LEN(S$)"KARTEN IM STOCK.":GOTO690
730 IFA=0THEN990
740 IFAZ(0)<KO*ATHENPRINT"SO DAZU HAST DU NICHT MEHR GENUG GELD.":GOTO690
750 FORI=1TOA
760 IFYY<0THENLZ(I-1)=I
770 IFYY<0THEN800
780 PRINTL$,I
790 INPUT"SO NUMMER DER || TEN ALTEN KARTE "
800 LZ(I-1)=LZ(I-1)-1:IFLZ(I-1)<0ORLZ(I-1)>4THEN790
810 IFI>1THENFORK=0TOI-2:IFLZ(I-1)=LZ(K)THEN790
820 IFI>1THENNEXT
830 NEXTI:GOTO980
840 I=VAL(WZ(Q)):A=(I=2)-2*(I<4)-(I<2)-(I=0):IFA>2THENA=A-1-INT(RND(1)*(Q+1)/Q-1/Q)
850 P=0:IFKO*A>AZ(Q)THENA=INT(AZ(Q)/KO):P=-1
860 IFLEN(S$)<ATHENA=LEN(S$):P=-1
870 IFA=0THEN960
880 FORK=0TOA-1:LZ(K)=K:NEXT
890 IFFZ(Q)=-1THEN930
900 IFI<1ORPTHENA=1:LZ(0)=FZ(Q):GOTO960

```

# programme

```

910 IFRND(1)*(1.5+SQR(Q*AN)*4/(KZ(Q,4)+1))>1THENA=1:LZ(Q)=FZ(Q)
920 GOTO960
930 IFEZ(Q)=-1THEN960
940 IFPORI<1THENA=1:LZ(Q)=EZ(Q):GOTO960
950 IFRND(1)*1.3+SQR(Q*AN)*3/(KZ(Q,4)+1)>1THENA=1:LZ(Q)=EZ(Q)
960 PRINTLN$N$(Q)" NIMMT SICH"A"NEUE KARTE";:IFA=1THENPRINT:GOTO980
970 PRINT"N"
980 IFATHENA%(Q)=A%(Q)-KO*A:PO=PO+KO*A:FORI=0TOA-1:GOSUB2140:KZ(Q,LZ(I))=A:NEXT
990 T=TI
1000 IFTI-T<60THEN1000
1010 NEXTJ:GOSUB1520
1020 G$="":MA=GR:AZ=A1:BI=GE
1030 GOSUB1360:PRINTLN$G$:T=TI
1040 IFTI-T<120THEN1040
1050 BI=FNJ(BI+1):IFAZ=1THEN1250
1060 IFMA=B%(BI)ANDMA>GRTHEN1290
1070 IFC%(BI)>1THEN1050
1080 G$="DU SETZST":PRINTLN$:IFBITHEN1100

1090 INPUT"SO WIE VIEL $ SETZT DU          ██████████";EI:GOTO1190
1100 G$=N$(BI)+" SETZT":RESTORE:FORP=-1TOVAL(W$(BI)):N=Z:READZ:NEXT
1110 N=N-(N-Z)*(ASC(MID$(W$(BI),2))-70)/26:Q=LOG(1+N):N=LOG(1+RND(1)*N)
1120 Z=14.7-LOG(A1+1)*MA/BI:IFZ<0THENZ=0
1130 EI=MA-B%(BI):IFZ>0THEN1180
1140 IFZ>0-1THENIFRND(1)*(1+2/BI)<0GOTO1190
1150 IFB%(BI)=50RRND(1)<.01THENIFZ>NTHENQ=N:EI=EI+PO*2/(A1*5):GOTO1180
1160 IFEI*RND(1)*A1*10/PO-1<2*VAL(W$(BI))THEN1190
1170 EI=0:GOTO1190
1180 EI=EI+1+RND(1)*(Z-Q)*HI/SQR(A1)
1190 EI=INT(EI):IFEI>A%(BI)THENEI=A%(BI)
1200 IFEI>LI+MA-B%(BI)THENEI=LI+MA-B%(BI)
1210 IFEI<MA-B%(BI)THENEI=0
1220 G$=G$+STR$(EI):G$=G$+"$ ZUSAETZLICH."
1230 IFEI=0THENC%(BI)=2:AZ=AZ-1:GOTO1030
1240 B%(BI)=B%(BI)+EI:A%(BI)=A%(BI)-EI:PO=PO+EI:MA=B%(BI):GOTO1030
1250 FORI=0TOAN:IFC%(I)>1THENNEXT
1260 CZ(I)=3:A%(I)=A%(I)+PO:I=AN:NEXT
1270 GOSUB1360
1280 GE=FNJ(GE+1):GOTO570
1290 FORI=0TOAN:IFC%(I)=0THENC%(I)=1
1300 NEXT:GOSUB1360:T=TI
1310 W$="":K=0:FORI=0TOAN
1320 IFC%(I)<2THENC%(I)=2:IFW$(I)>W$THENW$=W$(I):CZ(K)=2:CZ(I)=3:K=I
1330 NEXT:A%(K)=A%(K)+PO
1340 IFABS(TI-T)<3000ANDPEEK(151)=255THEN1340
1350 GOTO1270
1360 PRINT"5",,"██████POKER██████":IFGE>0THENPRINTN$(GE)" GIBT DIE KARTEN.
1370 IFGE=0THENPRINT"DU MUESSTEST GEBEN - ICH GEBE FUER DICH.";
1380 PRINT"0SPIELER KARTEN"," EINSATZ VERMOEGEN
1390 PRINT"
1400 FORJ=0TOAN:PRINT"|";:IFJTHENPRINTN$(J);:GOTO1420
1410 PRINTN$;
1420 PRINT"|";:ONC%(J)GOTO1470,1430,1450:PRINTSPC(15);:GOTO1480
1430 PRINT"[-AUFGEBEN- █";:GOTO1480
1440 PRINT"[-AUFGEBEN- ";:GOTO1480
1450 PRINT"█ * GEWONNEN * █";:GOTO1480
1460 PRINT"██ * GEWONNEN * █";:GOTO1480
1470 FORI=0TO4:A=KZ(J,I):GOSUB2100:PRINT"█"A$"██ ";:NEXT

1480 PRINT"|";:A$=STR$(B%(J)):A$=" "+A$:PRINTRIGHT$(A$,6);:PRINT"|";:A$=STR$(
A%(J))
1490 A$=" "+A$:PRINTRIGHT$(A$,8);:PRINT"|
1500 NEXTJ:PRINT"
1510 PRINT"AUF DEM TISCH LIEGEN"PO"███";:RETURN
1520 FORJ=0TOAN:FORI=0TO4:A=KZ(J,I):K=FNA(A):L=I:IFI=0THEN1620
1530 FORH=0TOI-1

```

```

1540 M=L%(H): IFL=IANDK<M THEN L=H
1550 IF (MAND15)-K THEN NEXT: GOTO1620
1560 K=M+16: L=H: N=H: FORH=HTOH+M/16: LZ(H)=K: IFK%(J,H)<A THEN L=L+1
1570 NEXT
1580 IFH=I THEN1620
1590 P=L%(H): IFP>K THEN1620
1600 Q=K%(J,H): N=N+1: L=L+1: FORQ=HTONSTEP-1: K%(J,Q)=K%(J,Q-1): LZ(Q)=L%(Q-1)
1610 NEXT: K%(J,Q)=Q: LZ(Q)=P: H=H+1: GOTO1580
1620 IFL=I THEN1640
1630 FORH=I-1 TOLSTEP-1: K%(J,H+1)=K%(J,H): LZ(H+1)=L%(H): NEXT: K%(J,L)=A
1640 LZ(L)=K: NEXT I
1650 I=INT(L%(4)/16): K=INT(L%(3-1)/16): IFI=3 AND K%(J,0)=47 AND K%(J,4)=51 THEN K=1
1660 Q=4: IF I THEN1690
1670 M=4: FORL=0 T04: Q=L%(M)-L%(L): M=L: IFQ-4 AND Q<>-9 THEN NEXT: GOTO1690
1680 K=1: Q=L-1: L=4: NEXT: IFQ<0 THEN Q=4
1690 W#=MID$("04123679", I*2+K+1, 1): FL=-1: IF I THEN1780
1700 Q=-1: H=0: M=FNB(K%(J,0))
1710 FORL=1 T04: A=FNB(K%(J,L)): IFA=M THEN Q=M: ONFL+1 GOTO1770: GOTO1750
1720 IF NOT H THEN IFA=HOR(A=HANDO=M) THEN1770
1730 IFA=H THEN FL=0: Q=H: GOTO1750
1740 H=A: FL=L
1750 NEXT: IFH=-1 THEN W#=MID$("58", K+1, 1)
1760 GOTO1780
1770 FL=-1
1780 F%(J)=FL: IFW#>"0" AND W#<"5" THEN1800
1790 FORQ=4 T00STEP-1: W#=W#+CHR$(FNA(K%(J,Q))+70): NEXT: W#=W#+CHR$(FNB(K%(J,4))+48)
): GOTO1820
1800 W#=W#+CHR$(FNA(K%(J,Q))+70): IFVAL(W#)=2 THEN W#=W#+CHR$(FNA(K%(J,2))+65)
1810 W#=W#+CHR$(FNB(K%(J,Q))+48)
1820 E%(J)=-1: ONVAL(W#)+1 GOTO1830, 1990: GOTO2090
1830 M=3: N=M: P=-1: H=P: Q=P: L=L%(4)-13: FORI=0 T04: K=L%(I): A=K-L: L=K
1840 ONAGOTO1960, 1910
1850 IFM=2 AND N=1 THEN2090
1860 M=M-1: IFM=0 THEN2090
1870 IFM<2 THEN IFFNP(P) THEN2090
1880 IFM<2 THEN P=P-4*(4=I-P): GOTO1960
1890 IFN=1 THEN P=Q+(Q-H)*FNP(H): GOTO1860
1900 P=I: GOTO1960
1910 N=N-1: IFN=0 THEN2090
1920 IFN=2 THEN H=I: GOTO1960
1930 IFM=1 THEN2090
1940 Q=I: IFM=3 THEN1960
1950 M=1: Q=FNP(P) AND ABS(P-H)=2: P=P-(P+2=HORP-1=H)*(H-P)-(I-P)*(P+3=H): IFQ THEN2090
0
1960 NEXT: Q=M=2 AND N=2: IFQ AND (H=P-1 OR H=P+4) THEN P=H
1970 IFQ THEN I=H: IFFNP(P) AND P=1 THEN P=0
1980 E%(J)=P: GOTO2090
1990 R=L%(3)-16
2000 M=2: P=-1: Q=P: O=P: N=M: L=L%(2)-13: FORI=0 T02: K=L%(I): A=K-L: ONAGOTO2060, 2040
2010 IFA=3 AND (R=L+1 OR R=K-1 OR R=K+12 OR R=L-12) THEN N=N-1: P=0: ONSGN(N)+1 GOTO2090, 2060
2020 M=M-1: O=K: Q=L-13*(L<0): IFM=0 GOTO2090
2030 GOTO2060
2040 IFL+1=R OR L+14=R THEN P=0: GOTO2060
2050 N=N-1: IFN=0 THEN2090
2060 L=K: NEXT: IF NOT P THEN2080
2070 FORI=0 T013STEP13: FORK=1 TON: IFR<>O-K+I AND R<>Q+K-I THEN NEXT: NEXT: GOTO2090
2080 E%(J)=3
2090 W%(J)=W%: NEXT J: RETURN
2100 A#=MID$("◆◆◆◆", FNB(A)+1, 1): B#=MID$("23456789ZBDKA", FNA(A)+1, 1): A#=A#+B%: RET
URN
2110 PRINT"DU BIST PLEITE!"
2120 PRINT"TSCHUESS": END
2130 PRINT"JETZT HABE ICH KEIN GELD MEHR.": GOTO2120
2140 A=LEN(S%): H=RND(1)*A: A#=MID$(S%, H+1): S%=MID$(S%, 1, H)+MID$(A%, 2): A=ASC(A%): R
ETURN

```

## Turmspringen – Turmspringen – Turmspringen – Turmspringen

Messen Sie sich als Turmspringer im olympischen

Wettkampf mit Ihren Gegnern!

**Eine weitere Disziplin in der Palette der olympischen Sportarten stellt das C16-Programm „TURMSPRINGEN“ dar. Stürzen Sie sich von der obersten Plattform des Sprungturmes und beeindrucken Sie die Punktrichter durch gekonnte Drehungen und wasserspritzerloses Eintauchen.**

Nach dem Start mit „RUN“ erscheint das Titelbild inclusive Lauf-schrift. Anschließend bitte die Space-Taste drücken, Anzahl der Spieler (1 – 8), Namen und Startberechtigung für das jeweilige Land eingeben. Jetzt wird auf dem Screen die Sportarena sichtbar und zusätzlich am linken oberen Bildschirmrand der Name des aktuellen Springers angezeigt.

Sie, als wagemutiger Turmspringer befinden sich auf der obersten Plattform des Sprungturmes und konzentrieren sich auf den ersten Durchgang, in dem ein „dreifacher Auerbachsalto“ gefordert ist.

Durch Betätigen des Feuerknopfes heben Sie vom Brett ab. In der Flugphase kann der Sprung per Joystick folgendermaßen beeinflusst werden:

- Joystick nach links – langsame Drehung vorwärts
- Joystick nach unten – schnelle Drehung rückwärts
- Joystick nach rechts – schnelle Drehung rückwärts
- Loslassen des Joysticks – Streckung des Springers

Besonders wichtig für die Qualität des Sprunges ist das möglichst senkrechte

Eintauchen ins Wasser und die Anzahl der Drehungen. Ein Internationales Punktrichtergremium, das aus 10 Personen besteht, bewertet die jeweilige sportliche Leistung. Es können maximal 1000 Punkte pro Durchgang erzielt werden. Sollten

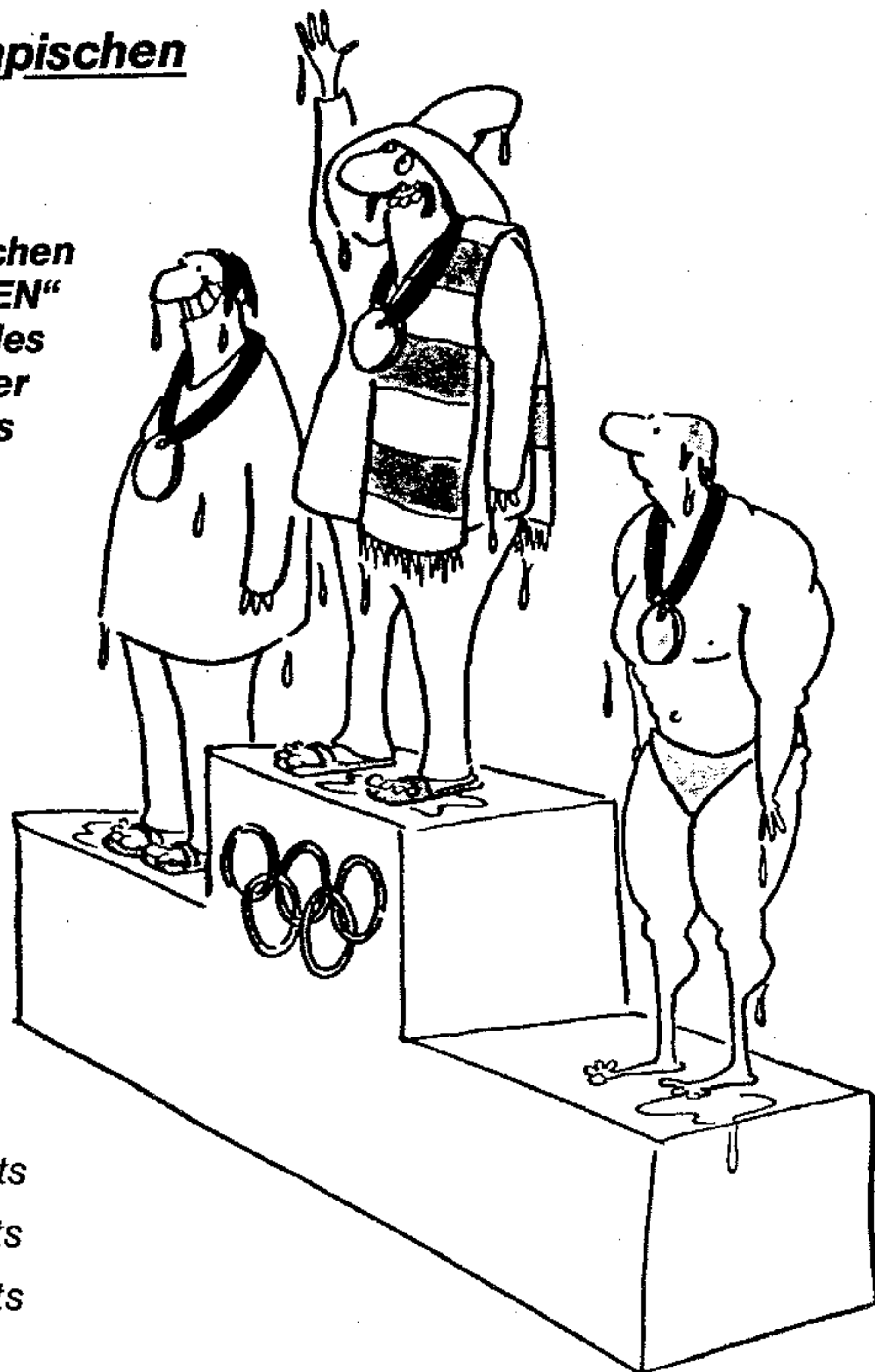
Sie mehr als 900 Punkte bei einem Sprung erreichen, belohnt Sie die Zuschauerkullisse mit tosendem Beifall. Der gesamte Wettkampf setzt sich aus fünf Durchgängen zusammen.

Am Ende der Veranstaltung folgt dann die Siegerehrung und die Medaillenverteilung.

5 GOT080

```
10 : *****
20 : * PLATFORM DIVING C-16 *
30 : * BY *
40 : * RALF ADLING *
50 : * LINGENER STRASSE 6 *
60 : * 4576 BERGE *
70 : *****
```

```
80 POKE56,47:CLR:I=65280:B=3300:VOL8:COLOR0,1,0:COLOR4,1:POKEI+18,PEEK(I+18)AND2
51
90 POKEI+19,PEEK(I+19)AND3OR48:FORT=832T0849:READA:POKET,A:NEXT:SYS832:DIMPG(11)
100 FORI=12800TO13180:READA:IFA>-1THENPOKEI,A:NEXT:RESTORE65535:RU=0
110 FORI=32512TO32537:READA:POKEI,A:NEXT
120 FORI=1TO10:READJ$(I):READN$(I):NA$(I)=J$(I):NEXT:GOSUB690
130 B=3132:D=2:F=0:G=0:H=40
```









```

1361 REM
1362 REM ZWISCHENSTAND II
1363 REM
1370 SCNCLR:COLOR1,2:CHAR1,4,2,"GROUND":PRINTRU"█":COLOR1,7,7:GOSUB1471
1380 FORI=C+1TO10:M=INT(RND(1)*1000+130):IFM>1000THENM=1000
1381 PG(I)=PG(I)+M:NEXT
1390 FORI=1TO10:CHAR1,3,I+5,NA$(I):CHAR1,10,I+5,N$(I):CHAR1,30,I+5,"●":PRINTPG(I)
)
1400 NEXTI
1410 SYS32512:GETA$:IFA$=" "THEN1430
1420 GOTO1410
1430 RETURN
1431 REM
1432 REM ZWISCHENSTAND III
1433 REM

1440 COLOR1,6,5:FORX=1TO10:CHAR1,6,11+X,J$(X):CHAR1,16,X+11,"":PRINTX:NEXT
1450 INPUTN:COLOR1,2
1460 IFN<1ORN>10THEN1450
1470 NA$(S)=J$(N):RETURN
1471 CHAR1,1,0,"":FORI=1TO23:CHAR1,1,I,""
1472 CHAR1,38,I,"":NEXT:CHAR1,1,23,"":RETURN
1479 REM
1480 REM ZEICHENSATZDATA'S
1490 REM
1500 DATA162,0,189,0,208,157,0,48,189,0,209,157,0,49,202,208,241,96

1510 DATA255,255,255,255,255,255,255,255
1520 DATA0,126,192,192,192,192,192,126
1530 DATA0,127,193,193,193,193,193,127
1540 DATA0,96,96,96,96,96,96,127
1550 DATA0,127,96,96,127,3,3,127
1560 DATA0,97,97,97,97,97,97,127
1570 DATA0,48,48,48,48,48,48,48
1580 DATA0,127,97,97,127,96,96,96
1590 DATA0,97,97,97,127,12,12,12
1600 DATA0,127,101,101,101,97,97,97
1610 DATA73,146,36,73,146,36,73,146
1620 DATA48,72,135,0,12,18,225,0
1630 DATA255,128,183,167,169,167,128,
255:REM REKLAME
1640 DATA0,17,149,255,0,66,82,255
1650 DATA238,0,119,0,235,0,238,0
1660 REM**** ECKZEICHEN ****
1670 DATA1,3,7,15,31,63,127,255
1680 DATA255,127,63,31,15,7,3,1
1690 DATA128,192,224,240,248,252,253,255
1700 DATA255,253,252,248,240,224,192,128
1710 REM *** SPRUNGBRETT ****
1720 DATA255,255,255,255,255,255,0,0
1730 DATA136,136,136,248,136,136,136,248
1740 REM *****JUMPER*****

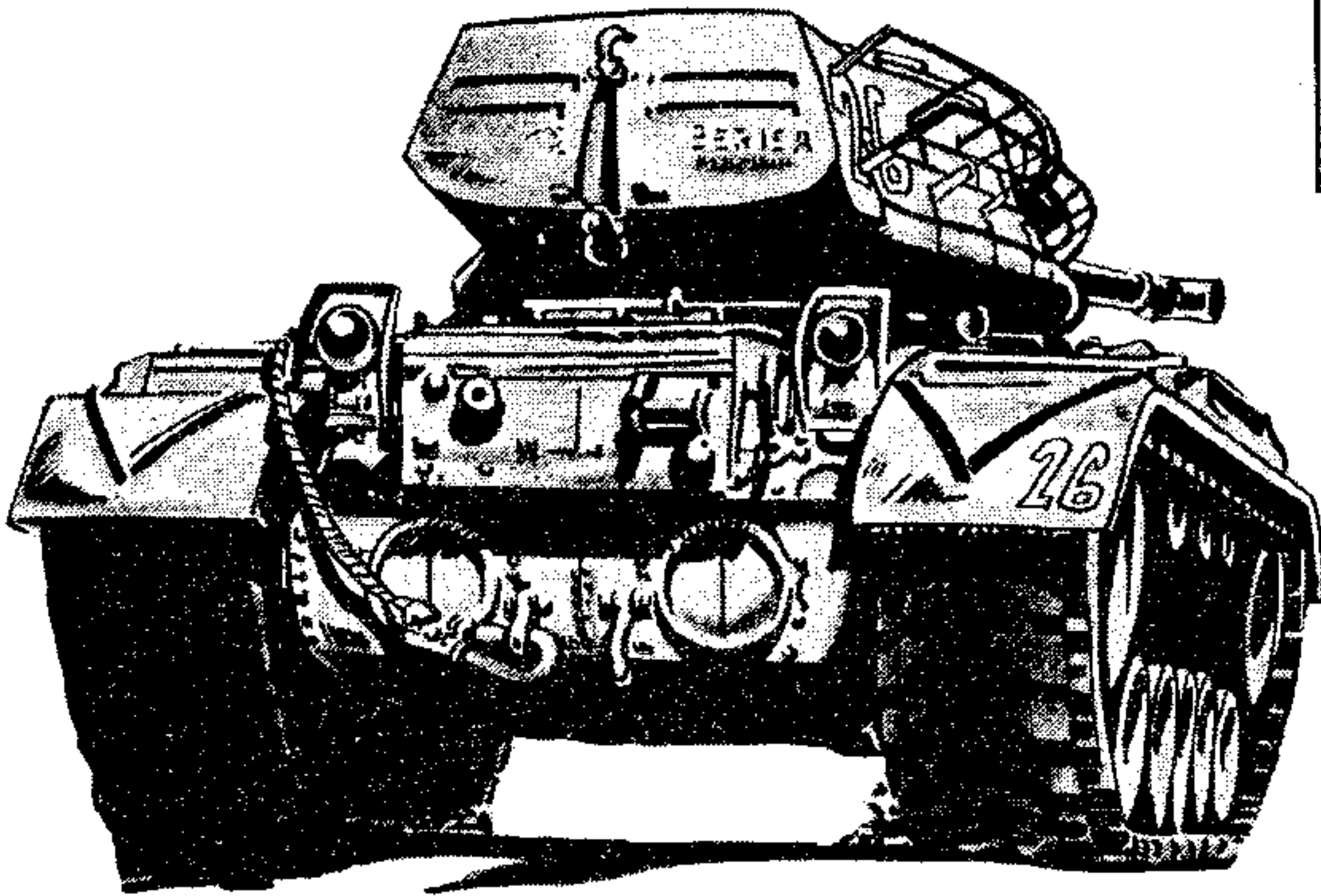
1750 DATA16,16,24,16,16,16,16,16
1760 DATA128,64,48,16,8,4,2,1
1770 DATA0,0,0,32,255,0,0,0
1780 DATA1,2,4,8,48,32,64,128
1790 DATA16,16,16,16,16,48,16,16
1800 DATA128,64,32,16,12,4,2,1
1810 DATA0,0,0,4,255,0,0,0
1820 DATA1,2,12,8,16,32,64,128
1830 REM **** DREHER ****
1840 DATA28,34,193,65,65,128,0,0
1850 DATA56,64,128,128,128,64,56,36
1860 DATA0,0,2,130,130,131,68,56
1870 DATA4,28,2,1,1,1,2,28
1880 DATA136,32,130,144,96,68,28,48
1890 DATA240,15,239,224,251,50,215,239
1900 DATA3,3,3,3,3,3,3,3
1910 DATA254,33,18,1,2,66,33,127
1920 DATA16,248,20,18,24,20,35,16
1930 DATA16,50,92,144,16,40,68,130
1935 DATA0,0,255,145,145,145,145,145
1940 DATA-1
1941 REM
1942 REM DATA'S FUER MASCHINENSPR.UPRO
1943 REM
1950 DATA110,88,50,110,92,50,234

1960 DATA110,33,51,110,35,51,110,37,51,110,32,51,110,34,51,110,39,51,96
1966 REM
1967 REM KONKURRENZ & LIEDDATAS
1968 REM
1970 DATAUSA,HUGHINS,URS,WASSILY,FRG,KUNZE,GDR,LEISTNER,JAP,MATAKONE,GB.,WALKER
1980 DATAFRA,GIRAUD,ITA,DILETTI,IRE,CRUWISH,AUS,GEMMERER
1990 DATA118,7,118,118,118,169,118,262,169,345,262,118,162,118,7,162,118,7,7

```

## Aus Spaß wird sehr leicht Ernst!

*Diese Erfahrung können Sie mit Hilfe des folgenden Programmes selbst sammeln. Es handelt sich hierbei nämlich zunächst um ein harmloses Manövrieren von tonnenschweren Kampfmaschinen, doch es wird schnell zu einer ernsten Auseinandersetzung mit Ihrem Gegner.*



## Tank-Wars

Sie und Ihr Freund haben sich in einem Labyrinth getroffen, um gemeinsam Panzerfahrübungen durchzuführen und dabei Ihre Fahrkünste unter Beweis zu stellen. Doch nach kurzer Zeit wird Ihrem Freund das ewige Kutschieren zu langweilig und er greift Sie an. Sie müssen nun durch geschickte Ausweichmanöver und schnelle Reaktionen diese Angriffe abwehren und Ihre Verteidigung einleiten.

Mit Hilfe des Joysticks haben Sie die Möglichkeit, verschiedene Fahrrichtungen einzuschlagen und Bomben oder Schüsse abzufeuern.

### Steuerung der Kampfmaschinen

Joystick rechts	=	Panzer rechts
Joystick links	=	Panzer links
Joystick oben	=	Panzer oben
Feuertaste	=	Schuß (geht in Richtung des Panzerrohres)
Joystick zurück + Feuertaste	=	Bombe legen (Wird am Standpunkt des Panzers abgelegt; Panzer rückt ein Feld in Richtung des Rohres. Auf diese Weise ist es auch möglich, Bomben und Mauern zu durchgehen).

Zu Anfang des Spieles besitzen Sie 500 Kraftpunkte, die jedoch bei jedem Schuß (30 Kraftpunkte) und bei jeder Bombe (100 Kraftpunkte) verringert werden. Ihr Gegner hat verloren, wenn er einen Punktestand von Null erreicht hat.

**Ein neues Spiel kann mit der Feuertaste gestartet werden.**



```
150 GOSUB2000:CLR
300 COLOR0,1:COLOR4,1:COLOR1,2
310 SCNCLR:VOLB:X1=70:X2=70
320 GOSUB590:COLOR1,2
330 S1=3300:S2=4000:FC=500:FD=500:BC=20:BD=20
340 D(1)=-40:D(3)=1:D(7)=-1
350 F(1)=-40:F(3)=1:F(7)=-1
360 R1=JOY(1):IFR1>127THENB30
```

```
370 R2=JOY(2):IFR2>127THEN910
380 P1=D(R1):P2=F(R2)
390 IFR1=3THENX1=70
400 IFR1=7THENX1=71
410 IFR2=3THENX2=70
420 IFR2=7THENX2=71
430 IFF1=0THEN570
```

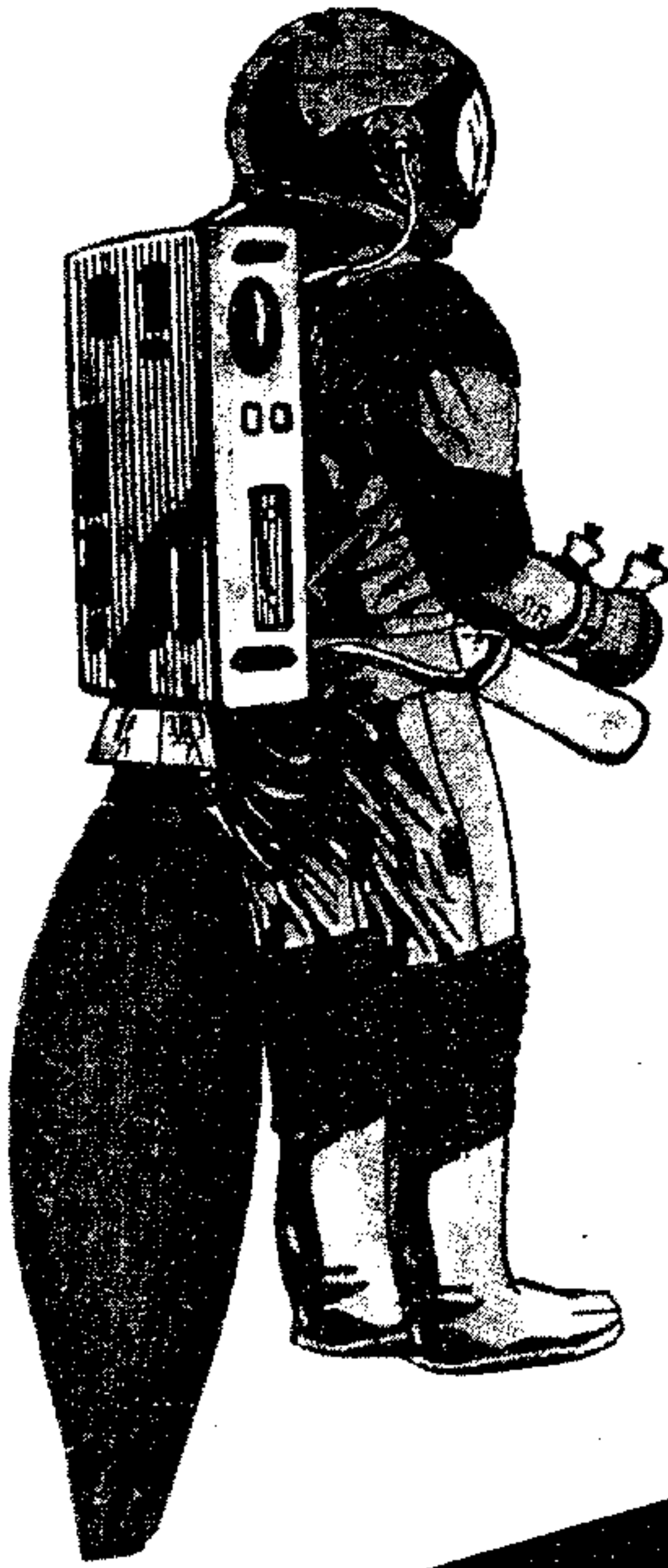
```
440 SOUND3,950,5:POKES1,32:IFPEEK(S1+P1)=67THENS1=S1+P1:GOTO990
450 IFPEEK(S1+P1)<>32THENP1=0
460 IFF2=0THEN490
470 SOUND3,900,5:POKES2,32:IFPEEK(S2+P2)=67THENS2=S2+P2:GOTO1040
```





# Fliegen Sie als C-16-Rocketman durch die Lüfte

Bei der Eröffnungsfeier der Olympischen Sommerspiele von Los Angeles 1982 war der fliegende Raketemann die Sensation und ist seitdem auch in der Bundesrepublik ein Begriff. Alle C-16-Besitzer haben jetzt durch dieses Programm die Gelegenheit, sich selbst einmal als Raketemann zu versuchen.



Als „ROCKETMAN“ haben Sie die Aufgabe, in der ersten Höhle mindestens 5 Diamanten einzusammeln. Dieses Vorhaben können Sie nur realisieren, indem Sie direkt auf die Diamanten zusteuern. Aber konzentrieren Sie sich nicht nur ausschließlich auf die Diamantenjagd, sondern weichen Sie gleichzeitig der gefährlich umherschwirrenden Kugel aus. Dieses „Flugobjekt“ strahlt tödliche Stromstöße aus. Falls Ihr Rocketman davon dreimal berührt wird, ist sein „fliegendes“ Dasein beendet.

Nach dem Einsammeln der Diamanten müssen Sie versuchen, die Plattform am rechten Screenrand zu erreichen. Ist Ihnen dies gelungen, ohne daß Sie gegen eine der zahlreichen Ecken, Kanten und Winkel geflogen sind, erscheint ein neues Bild. Hier haben Sie nun einen der vier abgebildeten Generatoren einzuschalten (erfolgt durch Überfliegen), erneut den schwierigen Weg zu meistern, um dadurch in den nächsten Level zu gelangen. Jeder eingeschaltete Generator erhöht die Flugeschwindigkeit des Raketemannes.

**Dieses Spiel wird selbst den erfahrensten „Joystick-Künstlern“ einige Schwierigkeiten bereiten. Doch die Mühe lohnt sich!**

Action mit dem Rocketman

```

1 PRINTCHR$(27) "N"
2 PRINT "ROCKETMAN" ROCKETMAN TEIL 1
3 PRINTCHR$(27) "T"
4 PRINT "MIT DIESEM PROGRAMM ERZEUGEN SIE
5 PRINT "EINEN NEUEN ZEICHENSATZ UND EIN KURZES
6 PRINT "MASCHINENPROGRAMM."
7 PRINT "WENN SIE KEINEN FEHLER BEI DER EINGABE
8 PRINT "DER DATAZEILEN GEMACHT HABEN, KÖNNEN
9 PRINT "SIE DAS PROGRAMM ABSPEICHERN ODER
10 PRINT "LOESCHEN.
11 PRINT "DRUECKEN SIE NUN EINE TASTE, UM DAS
12 PRINT "PROGRAMM ZU STARTEN.
13 GETA$: IFA$="" THEN GOTO 13
14 PRINT " "
15 COLOR 4,3,0: COLOR 0,1: COLOR 1,10,4: VOL 8
16 POKE 56,47: V=65280
17 POKE V+18,PEEK(V+18) AND 251
18 POKE V+19,PEEK(V+19) AND 30R48
19 PR=0: FORT=832T0849: READ A: PR=PR+A: POKET,A: NEXT
20 IF PR<>2115 THEN PRINT "FEHLER IN DATAS TEIL 1": END
21 SYS 832
22 PR=0: FORT=12800T013100: READ A: PR=PR+A: IFA>-1 THEN POKET,A: NEXT
23 IF PR<>26688 THEN PRINT "FEHLER IN DATAS TEIL 2": END
    
```







# programme

```
880 AO=3150:AU=3190:S=3271:K=0
890 GOSUB1220:GOSUB1250:GOSUB1270
900 POKEAO,85:POKEAU,86
910 POKEAO-1024,127:POKEAU-1024,127
920 COLOR1,4,6:CHAR1,4,20,"*":CHAR1,3,21,"|*":CHAR1,4,22,"+"
930 CHAR1,5,8,"*":CHAR1,4,9,"|*":CHAR1,5,10,"+"
940 CHAR1,9,5,"*":CHAR1,8,6,"|*":CHAR1,9,7,"+"
950 CHAR1,18,19,"*":CHAR1,17,20,"|*":CHAR1,18,21,"+"
960 FORX=1TO4:POKEA(X),B(X):NEXT
970 RETURN
980 IFZ=1THENGOSUB300:GOTO 995
990 IFZ=2THENGOSUB610:GOTO1085
995 GOSUB1281:GOSUB1283
1000 IFJOY(1)=0ANDPEEK(AU+40)=77THENGOTO1040:ELSEGOSUB1330
1010 POKEAU-F,32:POKEAO-F,32
1020 POKEAU,B:POKEAU-1024,127
1030 POKEAO,A:POKEAO-1024,127
1040 GOSUB1620
1050 W=W-.2:GOSUB1250
1060 IFW<=1THENGOTO1180
1065 IFK=4ANDAU=STHENGOSUB300:S=3913
1070 IFAU=SANDC>=5THENLETZ=Z+1:L=1:GOTO980
1080 GOTO1000
1085 GOSUB1281:GOSUB1283

1090 IFJOY(1)=0ANDPEEK(AU+40)=77THENGOTO1130:ELSEGOSUB1330
1100 POKEAU-F,32:POKEAO-F,32
1110 POKEAU,B:POKEAU-1024,127
1120 POKEAO,A:POKEAO-1024,127
1130 FORF=1TOR:NEXT
1140 W=W-.2:GOSUB1250:GOSUB1760:SYS294
1150 IFW<=1THENGOTO1180
1155 IFAU=SANDG=4THENGOTO2000
1160 IFAU=STHENLETZ=1:L=0:C=0:GOTO980
1170 GOTO1090
1180 REM GAME OVER

1190 COLOR1,2:PRINT"§"LEFT$(CU$,12)LEFT$(CR$,9)"|| B A M E O V E R ! "
1200 FORX=1TO300:NEXT:GOSUB1930:FORX=1TO5000:NEXT:RUN
1210 GOTO1200
1220 PRINT"§"SPC(9);:PRINT"||"USING"#####";SC
1230 IFPEEK(CR)<>32THENGOTO1290
1240 RETURN
1250 PRINT"§"SPC(22);:PRINT"||"USING"#####";INT(W)
1260 RETURN
1270 PRINT"§"SPC(36);:PRINT"||"USING"##";ME
1280 RETURN
1281 PRINT"§"LEFT$(CU$,24)LEFT$(CR$,13);
1282 PRINT"||"USING"##";C:RETURN
1283 PRINT"§"LEFT$(CU$,24)LEFT$(CR$,33);
1284 PRINT"||"USING"##";G:RETURN
1290 CR=INT(RND(1)*900)+3112
1300 IFPEEK(CR)<>32THENGOTO1290
1310 POKECR,42:POKECR-1024,238
1320 RETURN
1330 IFJOY(1)>10THENLETF=-40:PR=AO+F:GOSUB1400:GOTO1390
1340 IFJOY(1)=3THENGOTO1370
1350 IFJOY(1)=7THENGOTO1380
1360 F=40:PR=AU+F:GOSUB1400:GOTO1390
1370 F=1:A=82:B=84:PR=AU+F:GOSUB1400:PR=AO+F:GOSUB1400:GOTO1390
1380 F=-1:A=85:B=86:PR=AU+F:GOSUB1400:PR=AO+F:GOSUB1400:GOTO1390
1390 AO=AO+F:AU=AU+F:RETURN
1400 IFPEEK(PR)=42THENSOUND1,500,4:SOUND2,700,4:C=C+1:DI=DI+1:SC=SC+50:GOTO1430
1405 IFPR=KBTHENGOTO1511
1410 IFPEEK(PR)<>32THENGOTO1520
1420 RETURN
1430 POKECR,32:GOSUB1220:GOSUB1281
1440 GOSUB1290:IFC<>5THENGOTO1420
1450 FORY=1TO2:FORX=500TO1000STEP50:SOUND1,X,2:SOUND2,X-20,2:NEXT:NEXT
```

```

1460 SC=SC+1000:W=W+60:GOSUB1220:GOSUB1250:RETURN
1511 T=T+1
1512 POKEAU-1024,64:POKEAD-1024,64:POKEKG-1024,127
1513 FORX=1020TO980STEP-1:POKEKG,92:SOUND3,X,1:POKEKG,94:NEXT
1514 IFT=3THENGOTO1520
1519 POKEKG,94:RETURN
1520 POKEAU,32:POKEAD,32
1530 POKEAU+F,92:POKEAD+F,92
1540 POKEAU+F-1024,127
1550 POKEAD+F-1024,127
1560 FORS=800TO300STEP-20:SOUND3,S,1:NEXT
1570 POKEAU+F,93:POKEAD+F,93:T=0
1580 FORS=400TO100STEP20:SOUND3,S,1:NEXT
1590 POKEAU+F,32:POKEAD+F,32
1600 ME=ME-1:IFME=0THENGOTO1180
1601 IFZ=2ANDL=0THENLETB(X)=95:POKEA(X),B(X):G=G-1
1602 GOSUB1283
1610 Z=1:C=0:GOSUB1281:GOTO980
1620 V=INT(RND(1)*8)+1
1630 ONV GOTO1640,1650,1660,1670,1680,1690,1700,1710

1640 K=2:GOTO1720
1650 K=-2:GOTO1720
1660 K=80:GOTO1720
1670 K=-80:GOTO1720
1680 K=82:GOTO1720
1690 K=-82:GOTO1720
1700 K=78:GOTO1720
1710 K=-78:GOTO1720
1720 IFPEEK(KG+K)<>32
THENRETURN
1730 POKEKG,32:KG=KG+K
1740 POKEKG,94:POKEKG-
1024,68
1750 RETURN

1760 IFL=1THENIFPEEK(AU+40)=88THENGOTO1780
1770 RETURN
1780 IFPEEK(AU+80)=96THENRETURN
1790 FORX=1TO4:IFA(X)<>AU+80THENNEXT
1800 FORT=1020TO100STEP-20:SOUND1,T,2:SOUND2,T-30,2:NEXT:G=G+1:W=W+30
1804 R=R-3:GOSUB1283
1810 FORT=100TO1020STEP20:SOUND1,T,2:SOUND2,T-30,2:NEXT
1820 POKEA(X),96:B(X)=96:L=0
1830 C=0:GOSUB1281:RETURN
1840 KG=INT(RND(1)*900)+3112
1850 IFPEEK(KG)<>32THENGOTO1840
1860 POKEKG,94:POKEKG-1024,68
1870 RETURN
1880 RESTORE1900:FORX=1TO9:READA,B,C:SOUND1,A,B:SOUND2,A+09,B:FORY=1TOC:NEXT:NEX
T
1890 FORI=1TO400:NEXT:FORU=1TO8:VOLB-U:FORI=1TO40:NEXT:NEXT
1900 DATA 516,09,130,453,06,090,516,06,090,453,06,090,596,10,120,643,10,120
1910 DATA 569,10,120,643,10,120,685,50,1
1920 RETURN
1930 RESTORE1940:FORX=1TO11:READA,B,C:SOUND1,A,B:SOUND2,A+5,B:FORY=1TOC:NEXT:NEX
T
1940 DATA 262,35,550,262,30,001,169,25,310,262,40,120,383,30,120,345,25,140
1950 DATA 345,35,120,262,30,450,262,35,001,169,30,310,262,60,1
1960 RETURN
2000 PRINT"Q"
2010 PRINT"000 HERZLICHEN GLUECKWUNSCH !!"
2020 PRINT"000 SIE HABEN ALLE GENERATOREN EINGE-
2030 PRINT" SCHALTET."
2040 PRINT" DABEI ERREICHTEN SIE ";SC;" PUNKTE UND"
2050 PRINT" HATTEN NOCH EINEN SAUERSTOFFVORRAT"
2060 PRINT" VON ";W;" EINHEITEN."
2070 PRINT"000 MOECHTEN SIE NOCH EINMAL SPIELEN ?"
2080 PRINT"Q (J/N)"
2085 GOSUB1880
2090 GETA$:IFA$="J"THENRUN
2100 IFA$="N"THENPRINT"Q":END
2110 GOTO2090

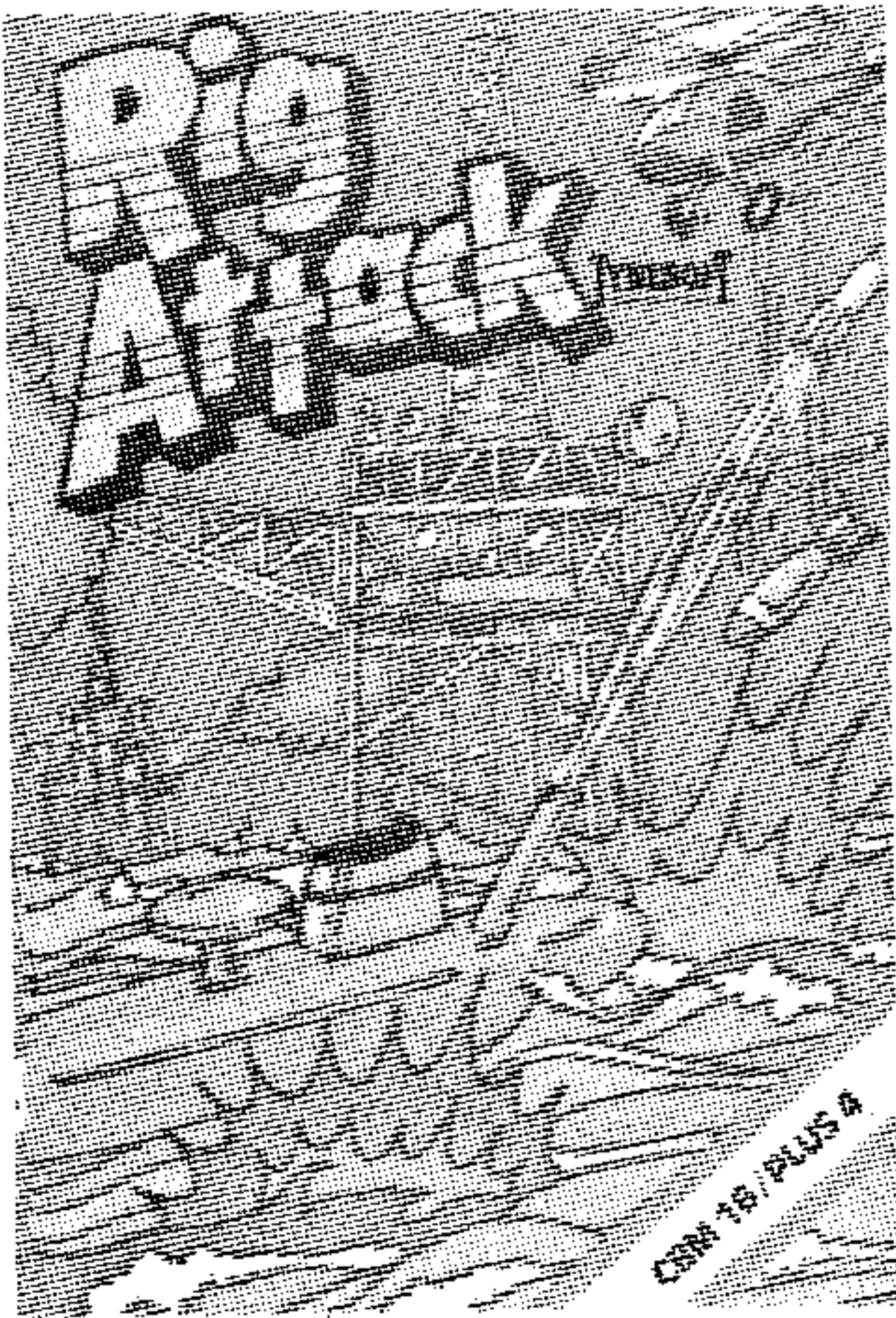
```

**An dieser Stelle ist der erste große Bereich – Spielprogramme – abgeschlossen.  
Auf den nächsten Seiten finden Sie noch einige Tips & Tricks und eine Vielzahl von  
Anwendungsprogrammen für Ihr System!**

Auf den nächsten Seiten finden Sie die neuesten Softwarevorstellungen für den C-16/116/Plus 4. Wir haben uns für unsere Leser stark gemacht und werden in der nächsten Zeit eine ganze Reihe von Programm-Software aus vielen Software-Häusern vorstellen.

## Rig Attack

„RIG ATTACK“ ist ein Action-Game mit ausgezeichneter Grafik und insgesamt sieben verschiedenen „scrolling screens“. Das Spielgeschehen ist äußerst interessant aufgebaut und sehr abwechslungsreich, so daß keine Langeweile auftreten kann.



Als Hubschrauberpilot müssen Sie gefährliche Aufträge erfüllen und die Angriffe feindlicher U-Boote überstehen. Dabei ist es aber gar nicht so einfach, den tödlichen Torpedos zu entweichen. Zu Ihrer

Verteidigung können Sie über das Wasser fliegen und Wasserbomben abwerfen. Das Spiel hinterließ in unserer Programmierabteilung durch die hervorragende Grafik und das abwechslungsreiche Spielgeschehen einen positiven Eindruck. **Hersteller: Tynesoft Computer Software, Blaydon**

★ ————— ★

## Formula 1-Simulator

*Die absolut realistische Darstellung eines Formel-1-Laufes.*

Dieses Programm der Software-Firma Mastertronic bereitet schon beim ersten Einstieg das reinste Spielvergnügen. Formula-1-Simulator zeichnet sich durch eine hervorragende realistische Darstellung des Rennstreckenverlaufs aus. Die Steuerung Ihres Rennwagens ist sehr direkt, so daß es nach einiger Übung nicht schwer fällt, die Ideallinie zu finden.

Was tatsächlich, auch dem erfahrenen Joystickpiloten, Schwierigkeiten bereitet, sind die komplizierten Überholvorgänge. Taucht ein anderer Rennwagen vor Ihnen auf, müssen Sie in Sekundenbruchteilen den richtigen Moment abpassen, die Lücke erspähen und an Ihrem Konkurrenten vorbeiziehen.

Für das Zurücklegen einer bestimmten Strecke und für erfolgreiche Überholvorgänge erhält man Punktgutschriften. Jede

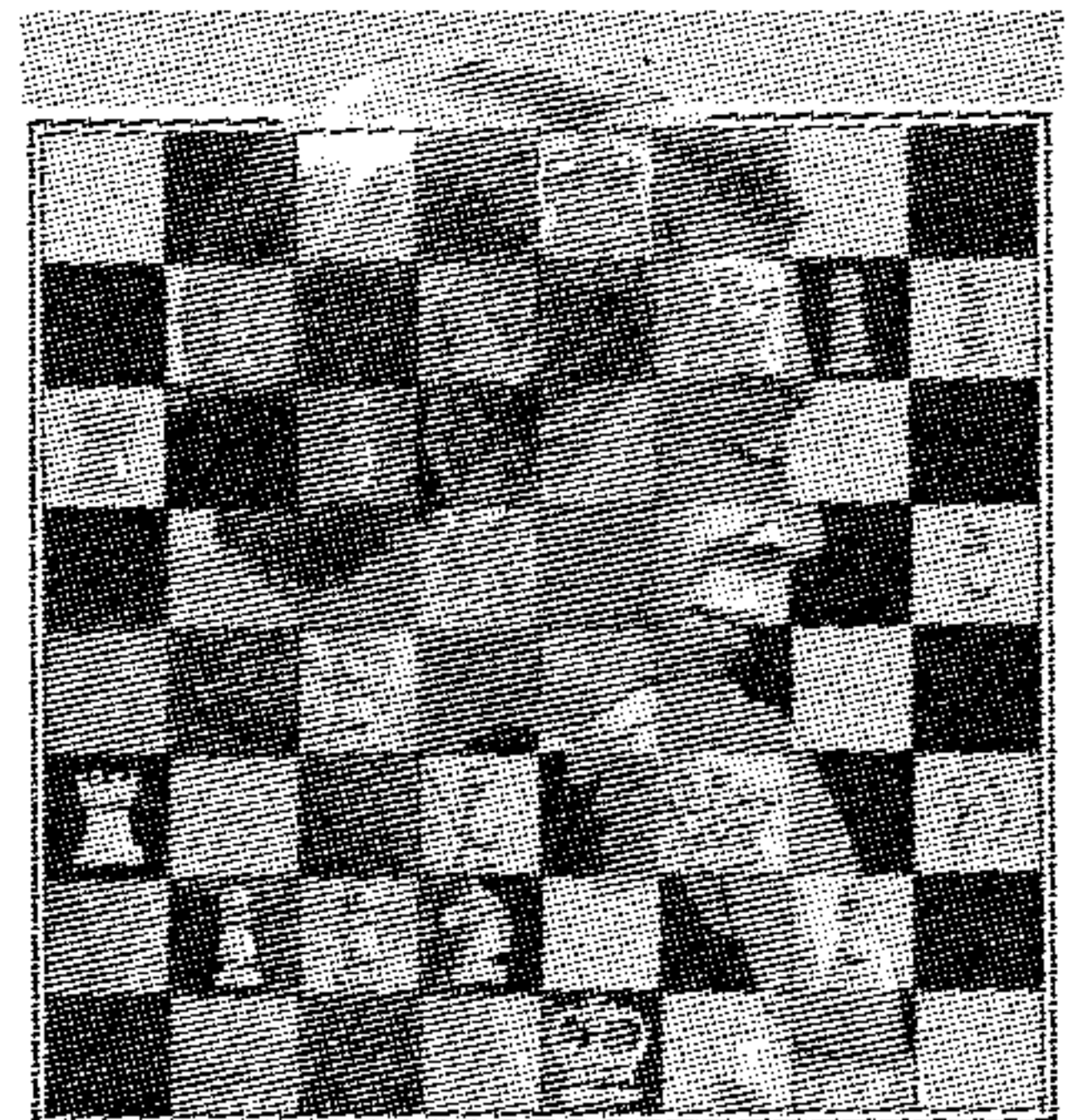
Kollision mit einem anderen Fahrzeug oder jedes Abkommen von der Straße wirft Sie entscheidend zurück. Das gesamte Renngeschehen ist natürlich dem Zeitfaktor unterworfen. Also wieder einmal ist ACTION im Höchsttempo gefordert!

Fazit: Für denjenigen, der vom Formel-1-Geschehen fasziniert ist und selbst einmal ein solches „Geschoß“ mit über 400 PS unter der Haube steuern möchte, ist die Anschaffung dieses Programms sehr zu empfehlen!

★ ————— ★

## GRAND MASTER

CHESS



Grandmaster für den Commodore 16/116 ist die neueste Version des bereits überall erhältlichen Programmes für die legendären Systeme C-64 und VC-20.

Dieses Schachspiel zählt nach Auskunft des Herstellers zu den besten Schachspie-

# Software

# REVIEWS

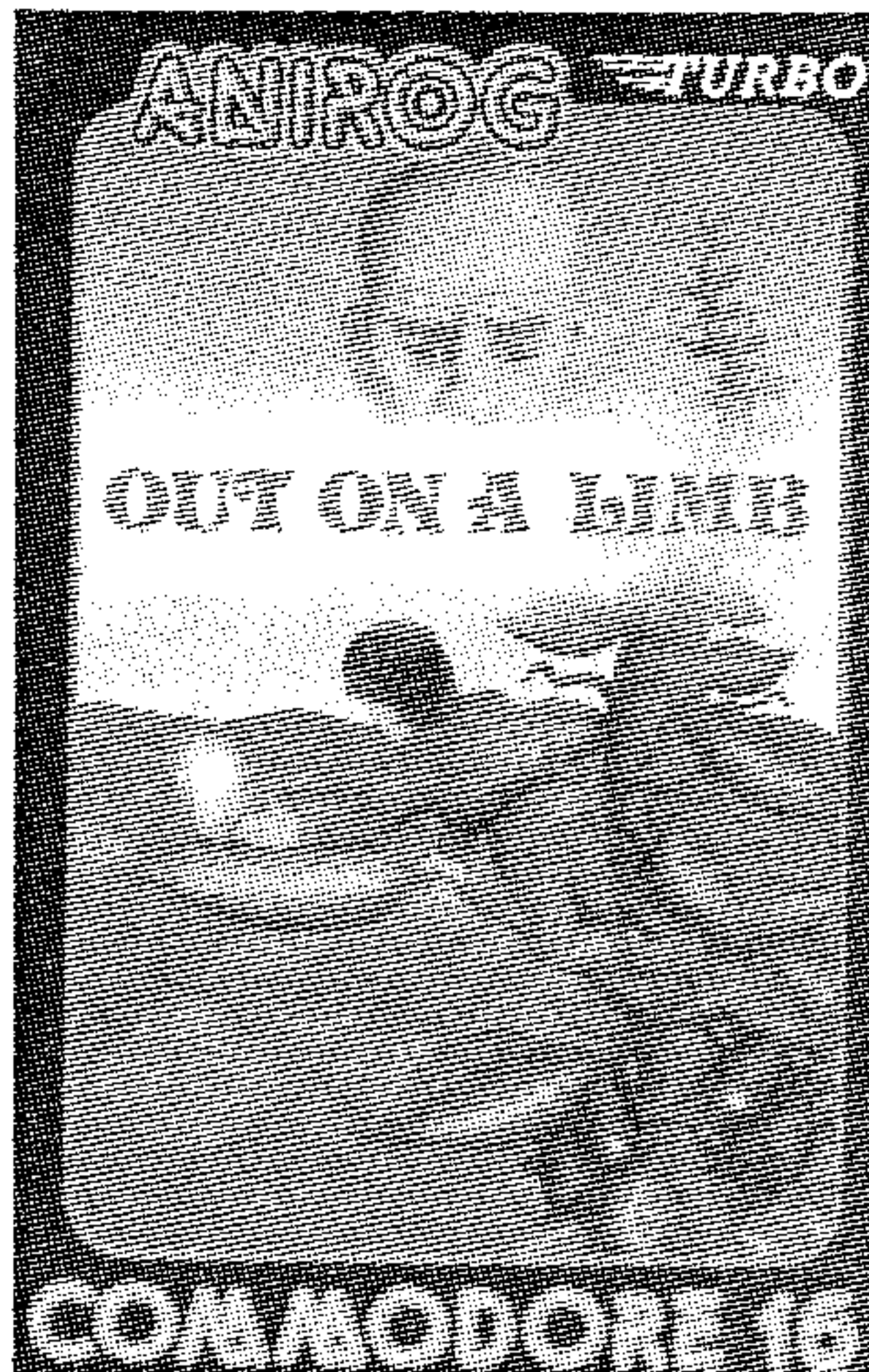
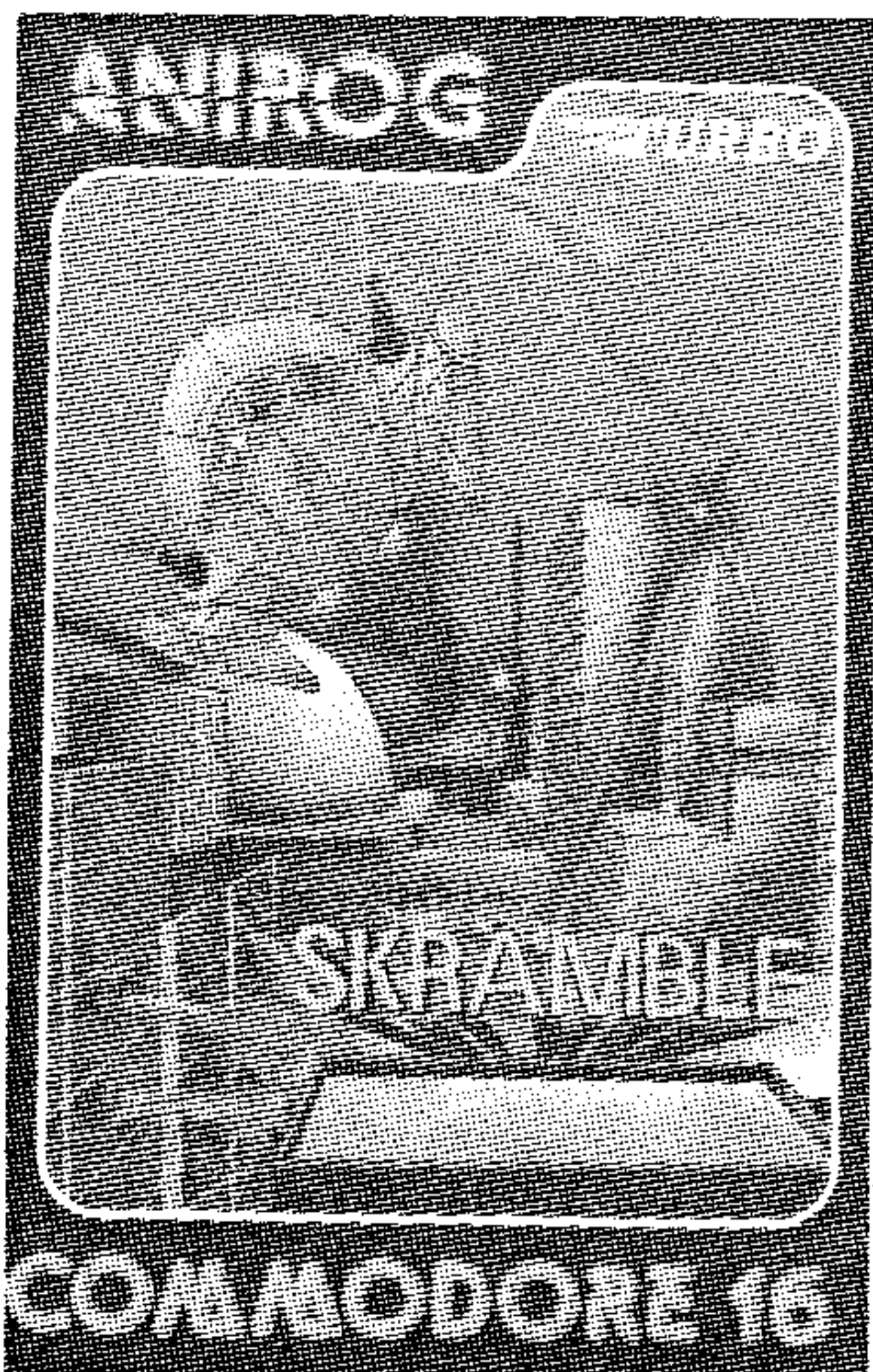
len, die auf dem deutschen Software-Markt derzeit angeboten werden. Es ist für Anfänger und Fortgeschrittene geeignet und verfügt über einige spezielle Funktionen, um Ihnen größtmöglichen Bedienungskomfort zu bieten. So können z. B. verschiedene Spielstufen gewählt werden, automatisches Spiel ist möglich, Wahl der Bildschirmfarben u. v. m. Grandmaster ist die gelungene Schachversion für den C-16/116 und sollte jedem schachinteressierten Computer-Besitzer eigentlich bekannt sein.

Hersteller: KING-SOFT, Roetgen



## SKRAMBLE

Das Spiel Skramble versetzt seinen Besitzer in die Lage, der Kommandant eines



Raumschiffes im Jahr 2184 zu sein und die Erde von der Herrschaft der superintelligenten „Command-Module“ zu befreien. Dazu müssen Sie über das Land fliegen und die stationierten feindlichen Raketen vernichten. Dazu steht Ihnen eine Laserkanone und eine unbegrenzte Anzahl an Bomben zur Verfügung.

Das Spiel zeichnet sich durch eine sehr gute scrollende Grafik aus. Der Spielgedanke und die Spielidee sind nicht mehr so ganz neu. Ein weiteres Plus ist aber die Spielgeschwindigkeit und das Maß an Geschicklichkeit, das man Ihnen bei diesem Spiel abverlangt.

## OUT ON A LIMB

Das Programm besteht hauptsächlich durch eine einzigartige Graphik, großen Spielwitz und technische Raffinessen.

Diese Version ist angelehnt an die Originalversion für den Commodore 64. Allerdings ist sie auf Grund des geringeren Speicherplatzes nicht ganz so spielerisch ausgestattet. Die Hauptaufgabe des kleinen Jack, das Schloß des Riesen zu erreichen und dort den Schatz finden, ist jedoch gleich. Natürlich muß er aber auch hier auf seinem Weg allerlei Abenteuer und Prüfungen überstehen.

Neben diesen Programmen hat die englische Software-Firma Anirog noch weitere Programmangebote für den C-16/116:

Petch, Star Commander, 3 D Time Trek, Catacombs sowie ein Programmangebot mit insgesamt vier Topprogrammen auf einer Kasette:

Flight Path 737, Moonbuggy, Las Vegas, Zodiac



## MONKEY MAGIC

Join Monkey fliegt auf einer Wolke durch die Lüfte und muß sich gefährlichen Angreifern zur Wehr setzen, die nach seinem Leben trachten. Sie haben insgesamt drei Leben zur Verfügung und müssen schon erheblich viel Geschick und Schnelligkeit vorweisen, um den Angreifern zu entgehen oder diese zu vernichten.

Die gute Grafik und das schnelle Spielgeschehen lassen bei diesem Spiel viel Spielfreude zu.

Das Spiel wechselt innerhalb drei verschiedener Levels, in denen die Angreifer aus unterschiedlicher Richtung und mit unterschiedlicher Geschwindigkeit auf Sie zukommen.



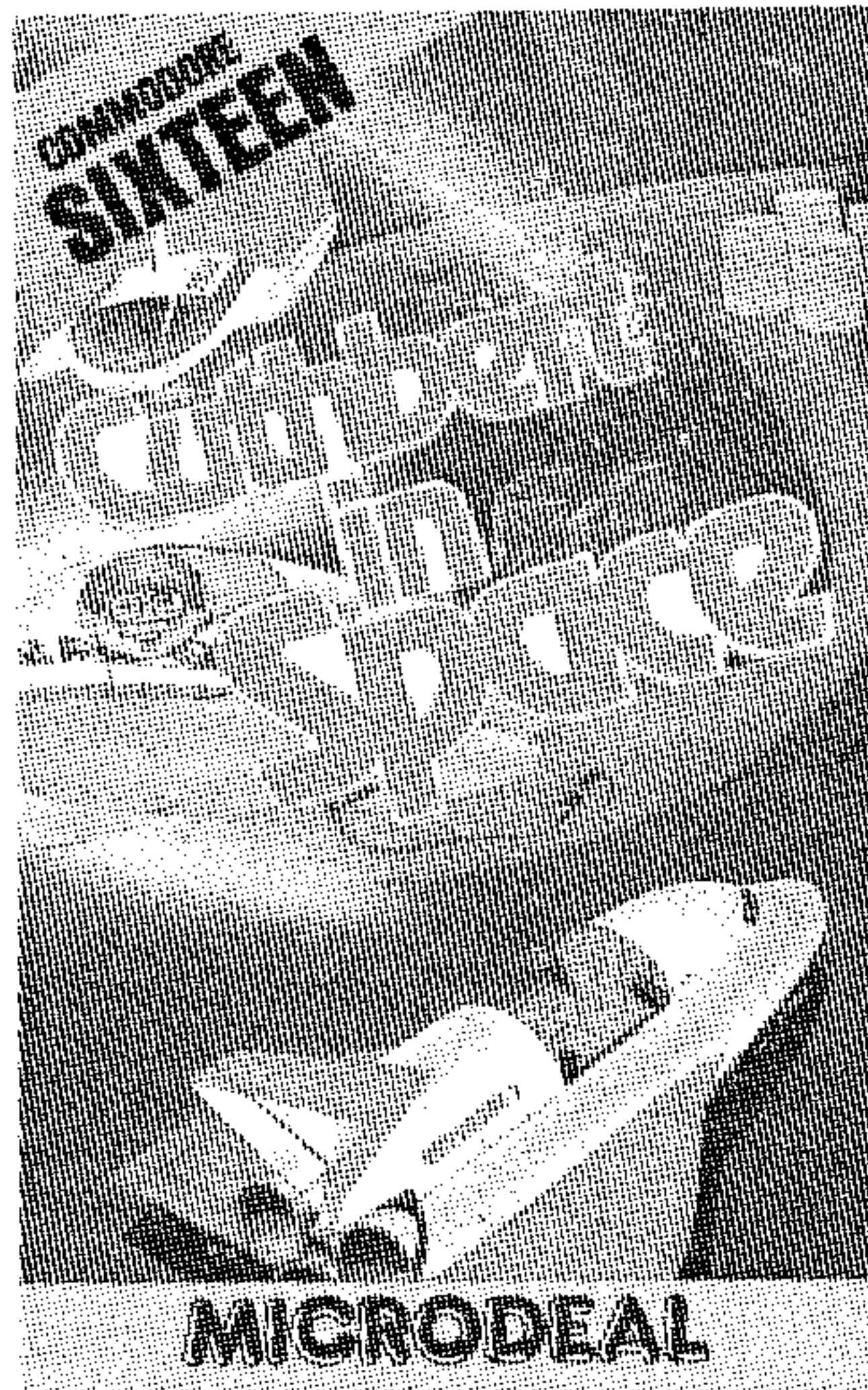
Weitere Programmangebote für den C-16/116 von der Firma Solair Software: Robin to the Rescue, Invasion 2000 A.D., Galaxions



## CUTHBERT IN SPACE

Das bekannte Software-Haus Microdeal stellt für den Commodore 16/116 ebenfalls zwei neue Programme vor. In der bereits bewährten und vielen „Hackern“ bekannten Cuthbert-Programmreihe gibt es den neuen Titel „Cuthbert in Space“. Es beschäftigt sich mit den Abenteuern des Cuthbert im All. Dieses Programm ist wie auch die anderen Cuthbert-Programme mit einer bestechenden Grafik ausgestat-

tet und der Spielwitz und die Geschicklichkeit sind hervorragend. Auch bei dem zweiten Cuthbert-Programm, mit dem Titel „Cuthbert in the Cooler“ ist es dem Software-Haus gelungen, die gewohnte Qualität der Programme beizubehalten.

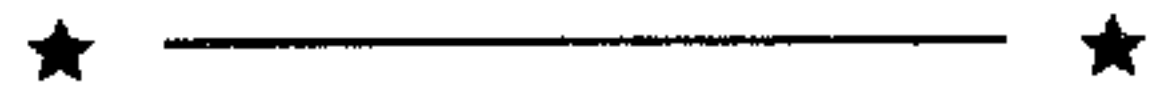
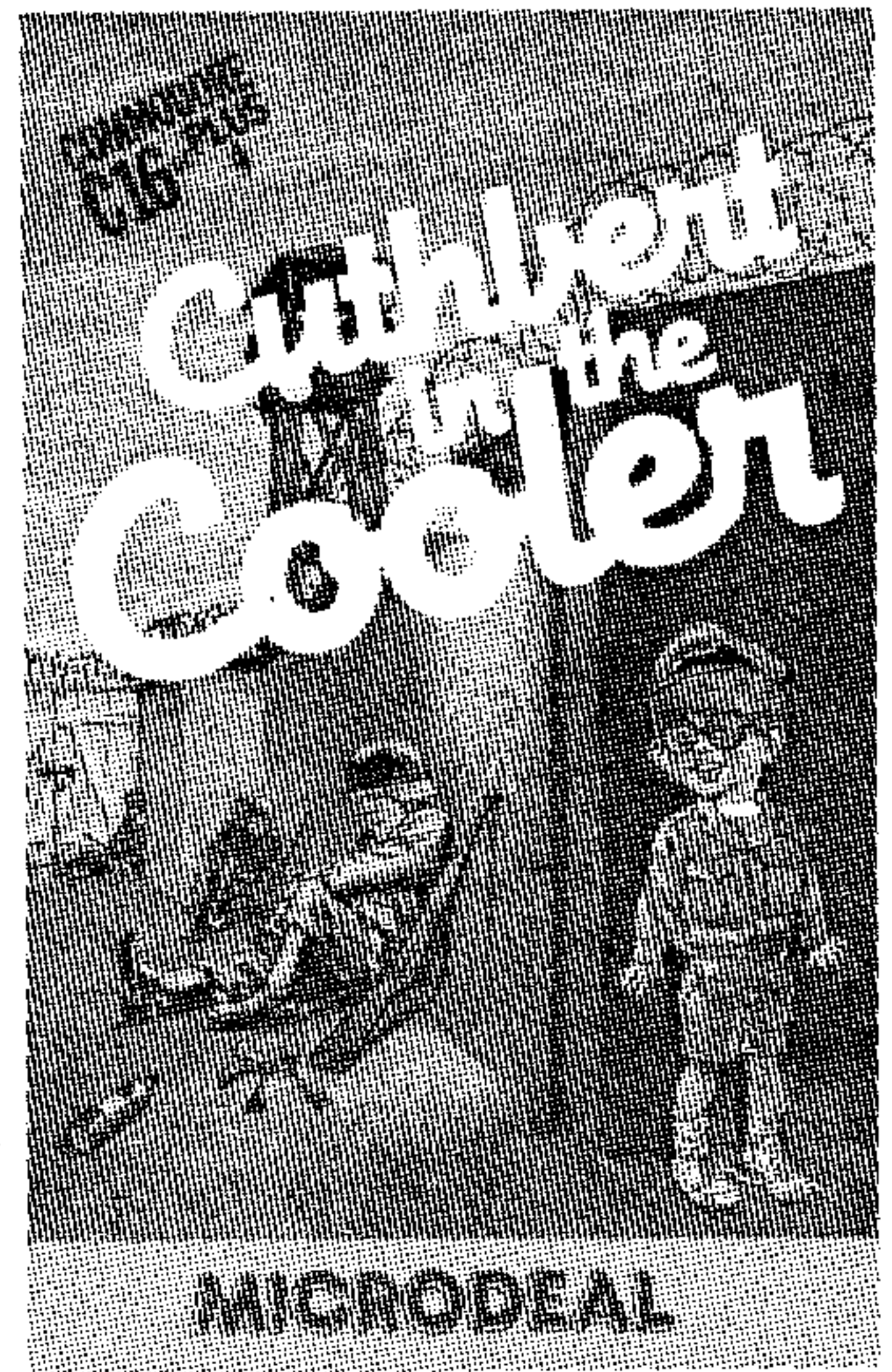


Hersteller:  
Microdeal, 41 Trauro-Rd, St. Austel, Cornwall  
PL 25, 5 JE



## CUTHBERT IN THE COOLER

„Cuthbert in the Cooler“ ist ein Action-Spiel, bei dem Sie mit Strategie und Einfallsreichtum an die Lösung der gestellten Aufgaben gehen müssen. Beide Programme stellen eine interessante Softwarebereicherung dar.



## COMMODORE 16 GAMES PACK I

Das Melbourne House bietet in seinem Software-Angebot neben verschiedenen einzelnen Programmpaketen für den C-16/116 auch zwei „GAMES PACK's“ an. Das hier abgebildete GAMES PACK I bietet insgesamt 15 verschiedene Spiele die Ihre Reflexe testen, Ihre Logik trainieren, Ihre Strategie und Ihre Intelligenz testen usw.

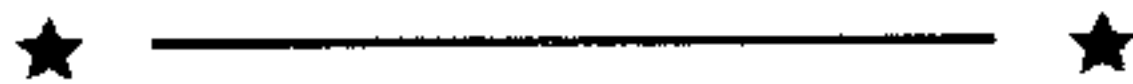
Auf der Kassette befinden sich folgende Programme:

**Micro Minotaur, Breakin', Warlock, Unscramble, Blockade, Hangman, Dragons Lair, Blackjack, Penetrator, Sam, Siege, 2D Maze, Zapp, Star, Trader, Looney Landa**



Das zweite „GAMES PACK“ beinhaltet folgende Programme:

Minotour, Hunter, Letterbox, Ganymede, Dumper, Alien Overrun, Dungeon, Chess, Scrambler, Chrash Barrier, Snakepit, Reel Thing, Mind Quiz, Tank Ambush, Road Patrol



## VEGAS JACKPOT

Wer kennt es nicht, das Mekka des Glücksspiels, „Las Vegas“. Das Spiel „Vegas Jackpot“ z. Z. auf Platz 20 in der englischen Software-Hitliste – versetzt Sie in die Welt des bezahlten Glücksspiels. Hier können Sie Ihrer Spielleidenschaft frönen, ohne gleich Haus und Hof verkaufen zu müssen, wenn Sie einmal die Pechsträhne

verfolgt. „Vegas Jackpot“ simuliert einen Geldspielautomaten, bei dem Sie Ihre Einsätze selbst bestimmen. Mit Hilfe der Risikovariante können Sie dann diese Einsätze verdoppeln oder max. £ 100 erreichen.



Die Bedienung ist relativ einfach, und die Grafik läßt die Vermutung zu, man hat einen echten Spielautomaten vor sich. Wenn Sie auch zu den „Spielernaturen“ gehören, dann sollte dies Programm nicht in Ihrer Sammlung fehlen.

Weitere Mastertronic-Programmangebote für den C-16/116:

**Spectipede**  
**Rockmann**  
**Tutti Frutti**  
**Squarm**  
**BMX-Races**  
**Bic-Mac**

In unseren nächsten Ausgabe von „Compute mit“ werden wir noch weitere Programme für den C-16/116 vorstellen. Wir glauben, hiermit einen wichtigen Beitrag für unsere Leser zu leisten.

Insgesamt sind derzeit in Deutschland über 60 Programme für den C-16/116 erhältlich. Die nachfolgende Liste gibt Ihnen einen Überblick über die Herstellerfirmen und die angebotenen Programme:

Hersteller: **Tynesoft**  
Programmangebote: **Climb it, Conoe Slalom, Hoppit, Zap-em, Munchit, Olympiad, Shoot it, Lunar Docking**

Hersteller: **Razar Soft**  
Programmangebote: **Mega Zap, Mail Trail, Seastrike, Raider**

Hersteller: **Solair Software**  
Programmangebote: **Robin to the Rescue, Invasion 2000 A.D., Galaxions.**

Hersteller: **Bubble Bus Software**  
Programmangebote: **Cave Fighter, Hustler**

Hersteller: **Adventure International**  
Programmangebote: **Gremlins, Spider-Man**

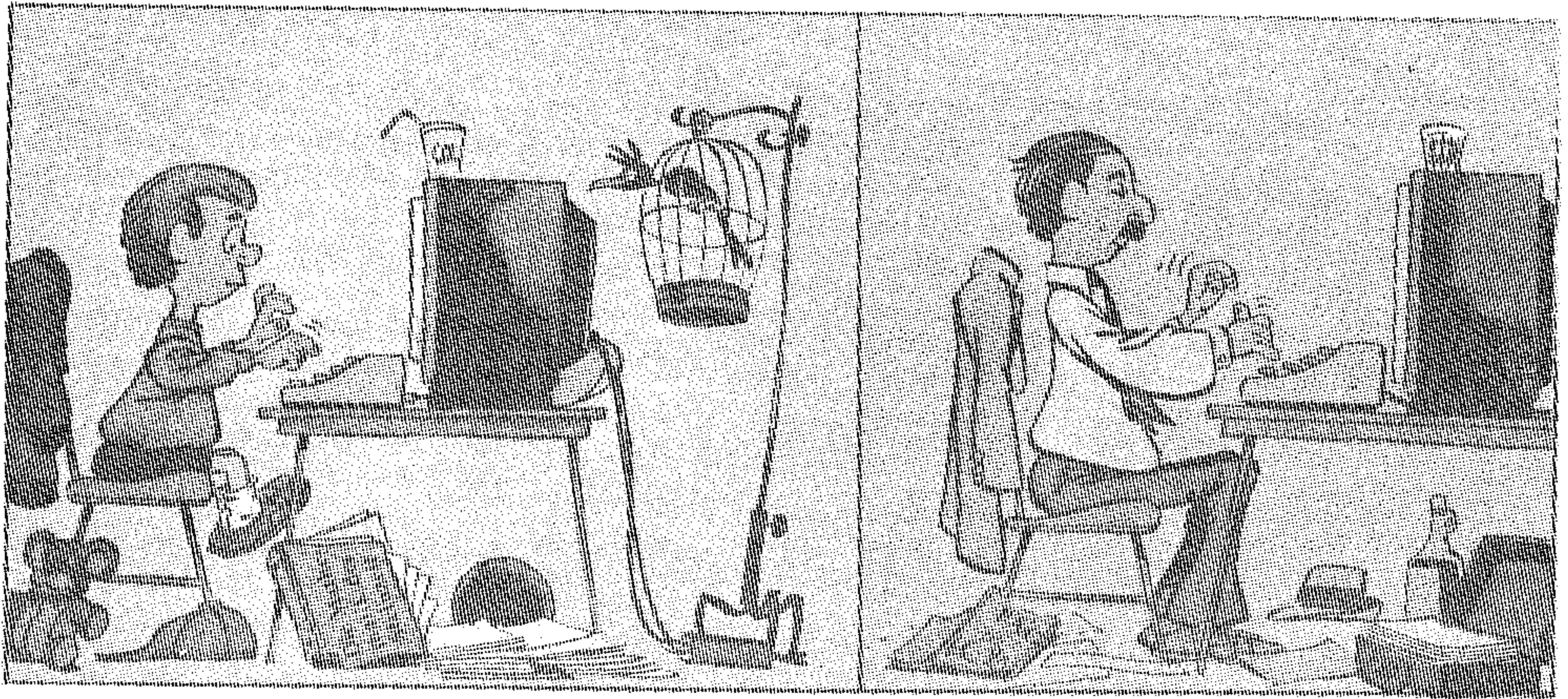
Hersteller: **Gremlin Graphics**  
Programmangebote: **Darks Dielemma, Tycoon Tex, Xargon Wars, Petals of Doom u.v.m.**

Alle hier vorgestellten Programme können direkt in Deutschland bestellt werden.

Die Bestellanschrift für die Programme lautet:

**RUSH-WARE GmbH**  
An der Gampkesbrücke  
4404 Kaarst  
Tel.: 0 21 01 / 6 00 40

# Software-Service



**Wenn Sie nicht wie diese beiden netten Herren alle Programme dieser Ausgabe abtippen möchten, füllen einfach die beiliegende Bestellkarte aus und senden Sie diese ein.**

*Wir liefern Ihnen alle Programme dieser Ausgabe auf Kasette oder Diskette innerhalb nur*

## 1 Woche

*nach Eingang der Bestellung in unserer Software-Abteilung.*

Programmpaket Kasette:

Bestell-Nr.: CSOK 1  
Preis: 49,50 DM

**Bitte beachten:** Bei telefonischen Bestellungen erhebt die Post noch eine Nachnahmegebühr von 5,- DM. Auslandslieferungen erfolgen nur gegen Vorkasse (Bargeld oder Verrechnungsscheck).

Programmpaket Diskette:

Bestell-Nr.: CSOD 1  
Preis: 59,50 DM

Beide Programmpakete enthalten alle Programme dieser Ausgabe.

### Weitere C-16/116 Programmangebote aus dem Tronic-Verlag:

Programm	Kasette	Diskette	Best.-Nr.
Fantasy-Country/Horror Castle	18,-- DM	-	023
Zeichendesigner	14,-- DM	-	VO 8/85
OLD-Routine/Merge-Routine/Change Type	14,-- DM	-	VO 1/86



# ??? Preisrätsel ???

## C-16/116 das unbekannte Wesen?

Geben Sie doch bitte einmal im Direktmodus folgenden Befehl ein:

**SYS 52652 (Return)**

Sicherlich wundern Sie sich auch über die „Reaktion“ Ihres Computers.

Preisfrage für alle Insider!

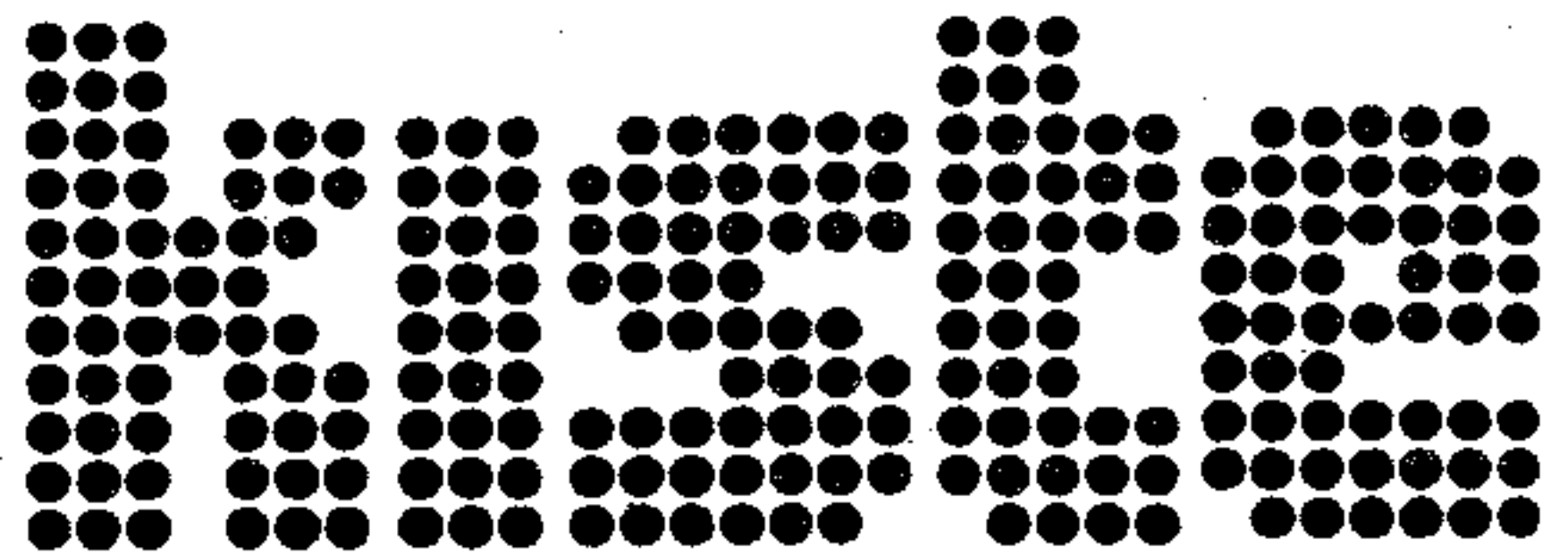
**Was hat das zu bedeuten?**

Die ersten 10 richtigen Einsender unserer Preisfrage erhalten eine Programmkassette mit allen veröffentlichten Programmen aus der neuen C-16/116 Sonderausgabe.  
Nutzen Sie Ihre Chance und lösen Sie unser kleines Rätsel. Der Rechtsweg ist ausgeschlossen.



Friedrichsstr.18 - Kassel - Tel.0561/13513

# software



**Software:** Flugsimulatoren, Psycho, Lotto, Diagnose, Datenprogr., Buchhalter, Assemblerkurs, Schach, KopierschutzK37, Horoskop, Auto-Kosten, Spiele ab 10.-DM ... und ... und.

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

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

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

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

**SCHNEIDER**

**mükra**  
DATEN-TECHNIK

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

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

Händler-  
anfragen  
erwünscht

**COMMODORE 16/116** ◀

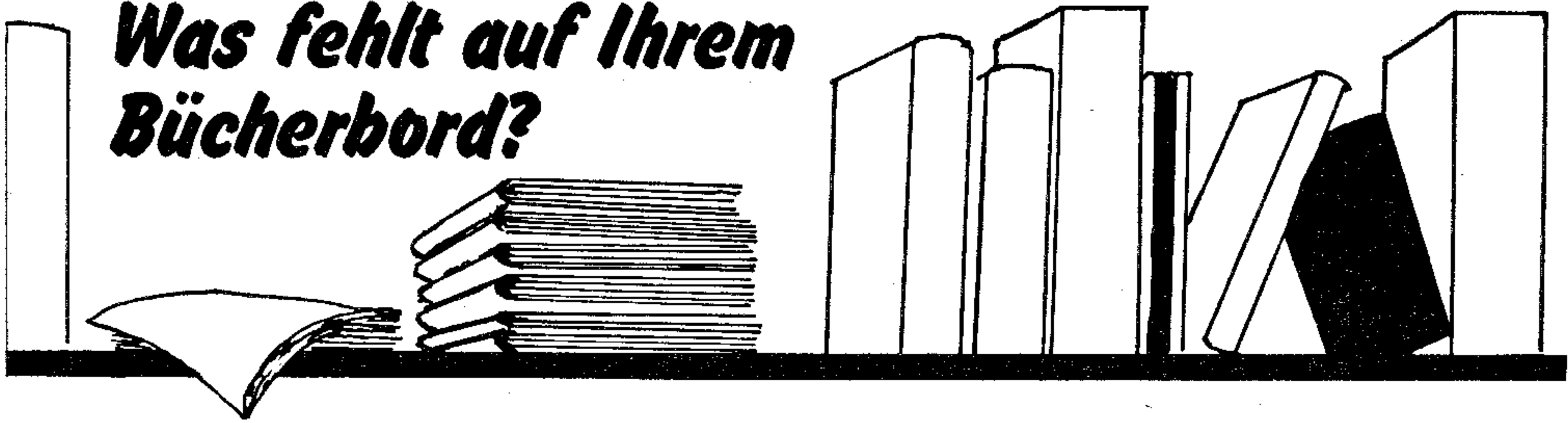
SOFT- UND HARDWARE IN GROSSER AUSWAHL!  
(JOYSTICKS/SPIELE/BÜCHER ETC.)  
KOSTENLOSE INFOS ANFORDERN BEI:

▶ **Software-Versand**  
**A. BACHLER**  
**POSTFACH 429**  
**4290 BOCHOLT**

SOFTWARE AUCH FÜR ANDERE COMPUTER!

COMMODORE

## Was fehlt auf Ihrem Bücherbord?



### **C-16 für Einsteiger, Norbert Szczepanowski**

Dieses Buch liegt vollständig in der so erfolgreichen Linie der großen Data Becker-Bücherfamilie. Sowohl in seiner typisch rot-weißen Broschur, als auch inhaltlich stellt es ein Äquivalent zu den übrigen „...für Einsteiger“-Titeln von Szczepanowski (ein Name, den man sich auf der Zunge zergehen lassen sollte) dar.

„C-16 für Einsteiger“ will sich als „das erste Buch zum COMMODORE 16“ verstanden wissen und nur unter diesem Aspekt kann man es richtig beurteilen. So entsprechen die ersten drei Kapitel dieses Buches im Grunde den ersten drei Abschnitten aus dem Handbuch. Es werden nacheinander das Auspacken und der Anschluß des C-16, sowie der wichtigsten Peripheriegerä- te, die Bedienung der Tastatur und der Umgang mit fertig gekaufter Software erklärt. Szczepanowski schreibt dabei in einem leicht-lockeren Stil, der dem Einsteiger diese trockenen Einleitungskapitel um einiges schmackhafter macht, als dies im Handbuch geschieht.

Zusätzlich werden alle Ausführungen mit einer Unzahl von Bildschirm- und Tastaturfotos dokumentiert, wodurch es dem Leser beinahe unmöglich wird, etwas nicht zu verstehen oder falsch zu machen. Im vierten Kapitel („Der erste Befehl“) erfährt dann der PRINT-Befehl eine Würdigung, die seiner Bedeutung beim Programmieren wohl gerecht wird. In einem ganzen Kapitel wird er sehr ausführlich und in allen Varianten bis in den äußersten Winkel ausgeleuchtet. Um eventuellen Verwirrungen aus dem Wege zu gehen, ist

es auch derselbe PRINT-Befehl, der im nächsten Kapitel („Das erste Programm“) dazu dient, zu erläutern, was ein Programm ist. Dabei wird nicht Wert darauf gelegt, ein besonders sinnvolles Programm zu schaffen, sondern es wird sehr detailliert auf Aufbau und Sinn und Zweck eines Programms eingegangen.

Im sechsten Kapitel („Programmierhilfen“) wird schließlich der letzte Grundstein gelegt, um endlich mit der eigentlichen Programmierung beginnen zu können. Es werden die Befehle erläutert, die den Programmierer bei der Erstellung von Programmen unterstützen sollen (AUTO, RENUMBER etc.). Der Autor macht dabei erstmals von einem, wie wir meinen sehr

sinnvollen Konzept Gebrauch. Neue Befehle werden eingeführt, indem sie in einem grau abgesetzten Kasten nach folgenden Kriterien vorgestellt werden:

**Problemstellung:** Hier wird genau die Wirkung des Befehls erläutert.

**Befehl:** Der Befehl wird in seiner genauen Syntax und mit allen Parametern genannt.

**Parameter:** Die Parameterliste wird genau erläutert.

**Beispiel:** Für jeden Befehl wird ein typisches Beispiel geliefert.

**Bemerkungen:** Spezifische Besonderheiten werden gesondert herausgestellt.

Diese Art der Befehlseinführung ist sehr übersichtlich und leicht verständlich, bei späterem Nachschlagen dient sie hervorragend als Gedächtnisstütze.

Im siebten Kapitel wird es dann interessant. Stück für Stück entwickelt der Autor eine kleine Adressenverwaltung in Basic, wobei jeder neue benötigte Befehl nach dem oben beschriebenen Konzept eingeführt wird. Da jeder neue Programmabschnitt sehr ausführlich dokumentiert wird, dürfte es auch für den Einsteiger kein Problem sein, den Programmablauf nachzuvollziehen und zu verstehen. Was besonders angenehm in diesem Abschnitt auffällt, ist die sehr gute Lesbarkeit des Programmlistings.

Der letzte Abschnitt in diesem Buch befaßt sich schließlich noch mit der sequentiell-



len Datenspeicherung auf Diskette und Kassette, sowie der Abfrage der Joysticks von Basic aus. Er ist wie alle anderen sehr ausführlich und leicht verständlich. Was leider negativ an diesem Buch auffällt sind zwei gewichtige Punkte: zum einen werden nicht alle Befehle und Anweisungen, die der C-16 versteht, in diesem Buch erwähnt (so fehlen allein die Grafik- und Soundbefehle gänzlich), und zum an-

deren sind DM 29,-- für ca. 200 Seiten Computerbuch einfach zu viel. Dennoch kann man, unserer Meinung nach, dieses Buch jedem empfehlen, der mit dem C-16 wirklich ganz neu in die Computerei einsteigt und für den der Computer noch ein großer schwarzer Kasten ist. Die vielen Umsteiger dagegen, die vom VC-20 auf den C-16 wechseln, werden kaum etwas entdecken, was für Sie

interessant wäre. Ob ein erweitertes Handbuch im Endeffekt DM 29,-- wert ist, muß jeder für sich entscheiden und hängt sicherlich vom Geldbeutel des jeweiligen Anwenders ab. **Waldemar Raaz**

**Titel:** C-16 für Einsteiger  
**Autor:** Norbert Szczepanowski  
**Verlag:** Data Becker, Düsseldorf  
**Bestellnr.:** ISBN 3-89011-068-1

## Der Commodore 16 in der Praxis, David Lawrence

Dieses Buch für den Commodore 16 entspricht inhaltlich dem Titel „Der Commodore 64 in der Praxis“, der ja mittlerweile eine Art Standard für C-64-Anwender geworden ist. Es handelt sich dabei um die deutsche Ausgabe des englischen Originaltitels „The Working Commodore 16“ und ist von David Lawrence, dem wohl erfolgreichsten Autoren innerhalb der Commodore Sachbuchreihe, geschrieben. Am Inhalt und Stil ist leicht zu erkennen, daß es sich nicht gerade um das Erstlingswerk des Autors handelt. Der Text liest sich sehr gut und ist leicht zu verstehen, bis auf einige Druckfehler, die gerade den Anfänger sehr stark verwirren können, da sie zum Teil den logischen Gedankenablauf verfälschen.

Im Gegensatz zu Data Beckers „C-16 für Einsteiger“, könnte man dieses Buch vielleicht am besten als das Buch nach dem Handbuch bezeichnen. Die formale Kenntnis aller Befehle des C-16-Basic wird in diesem Buch von Anfang an vorausgesetzt. Ziel ist es nun, mit Hilfe dieser formalen Kenntnis dem Anfänger, und nicht nur dem, die nötige Erfahrung im Umgang mit diesen Befehlen, sowie die wichtigsten Programmier-techniken zu vermitteln. Dabei greift Lawrence auf ein vielfach erprobtes Konzept zurück: er entwickelt im Grunde gemeinsam mit dem Leser eine Anzahl an Programmen (in diesem Buch 17 an der Zahl), die zum einen, alle interessanten Bereiche der Computeranwendung abdecken und zum anderen im abgetippten Zustand eine ansehnliche Pro-

grammbibliothek bilden, was bei dem mageren Softwareangebot für den C-16 nicht zu unterschätzen ist. Dies mag den Eindruck erwecken, es handle sich bei diesem Buch um eine simple Programmsammlung für den C-16. Dieser Eindruck ist auch gar nicht falsch. „Der C-16 in der Praxis“ ist auch eine Programmsammlung. Doch werden die Programme, wie gesagt, nicht einfach abgedruckt, sondern modulweise entwickelt. Dabei wird zunächst die Problemstellung, der das Programm gerecht werden soll, theoretisch ausgeleuchtet, und dann wird das Programm abschnittsweise abgedruckt, wobei jeder Abschnitt zeilenorientiert sehr, sehr ausführlich dokumentiert und erklärt wird. Mit diesen Erläuterungen sollte es Niemandem Schwierigkeiten bereiten, den Programmablauf zu verstehen und nachzuvollziehen. Die Eingabe wird also nicht etwa zur stupiden Abtipperei, sondern der Anwender, zumindest wenn er nicht nur auf die Programme erpicht ist und die Erläuterungen übergeht, lernt nebenher eine ganze Menge über die Programmierung in Basic. Lawrence hat dabei versucht zusätzlich solche Programme auszusuchen, die einerseits nicht zu schwierig und vor allem nicht zu lang sind, und die andererseits trotzdem nützlich und sinnvoll sind. Diese Mischung ist ihm hervorragend gelungen. Es sind Programme aus folgenden Bereichen enthalten: verschiedene Darstellungen der Uhrzeit, Grafiken im Low- und Hi-Res-Modus, Zeichen im hochauflösenden Grafikmodus, Musikprogramme

zum Erstellen von Musikstücken, Verwaltung von Daten, Erstellen von Texten, Verwaltung von Konten.

Im großen und ganzen also aus allen klassischen Gebieten der Computeranwendung. In den Listings sind übrigens alle Steuercodes in einen Klartext-Code übersetzt, was enorm zur Lesbarkeit der Listings beiträgt. Negativ ist uns an diesem Buch eigentlich nur ein Punkt aufgefallen: die Übersichtlichkeit scheint aufgrund der kleinen Auflage, in der das Buch erschienen ist, in den Hintergrund getreten zu sein. So ist der Beginn eines neuen Kapitels schon schwerlich auszumachen, doch nach dem Beginn eines neuen Programms innerhalb eines Kapitels muß man schon einige Zeit suchen, wenn man einmal ein Programm nachschlagen möchte. Dies ist jedoch ein Punkt, der nach unserer Meinung nicht sehr schwerwiegend ist. Wieder erfreulich ist hingegen der Preis dieses 210 Seiten umfassenden Buches. Man muß exakt DM 29,80 locker machen, um in den Genuß dieses Werkes zu kommen. Ein Preis, der gemessen an dem Inhalt des Buches durchaus gerechtfertigt ist. Leider scheint es gar nicht so einfach zu sein, dieses Buch käuflich zu erwerben. In allen Computerläden, wurde sofort abgewinkt: „Führen wir nicht! Bestellen wir auch nicht!“ Eine äußerst seltsame Konstellation. Ich kann nur empfehlen, das Buch in einer Buchhandlung zu bestellen. Unbedingt zu erwähnen bleibt, daß „Der C-16 in der Praxis“ natürlich auch für alle Be-

sitzer des Commodore 116 geeignet ist, obwohl der Titel dies verschweigt, und auch sonst nirgends im Buch diese Möglichkeit erwähnt wird. Für alle, die es immer noch nicht wissen oder glauben können: der C-16 und der C-116 sind bis auf das Gehäuse völlig identisch.

Waldemar Raaz

**Titel:** Der Commodore 16 in der Praxis  
**Autor:** David Lawrence  
**Verlag:** Commodore Sachbuchreihe  
**Bestellnr.:** ISBN 3-89133-018-9

## Weitere Fachbücher für den Commodore 16/116:

Froitzheim, D./J. Kausmann

### Effektiv & Kreativ mit dem Commodore Plus/4

Arbeiten mit der eingebauten Software. Data-Becker, D. 1985. 244 S. Kart. DM 49,00  
ISBN 3-890110-73-8

### PLUS/4 ROM-Listing

Sachbuchreihe Bd. 6. Commodore, F. 1984. 280 S. Brosch. DM 59,00  
ISBN 3-891330-06-5

## Commodore 16 mit 116

Der Einsteiger-Computer mit Aufsteigerqualitäten. Kiehl, L. 1984. 164 S. DM 29,80  
ISBN 4-470-80481-8

● Scharnbacher, K.

### Lerne BASIC mit dem C 116 / 16 / plus 4

Programmierte Unterweisung. Dt. Betriebswirteverlag, G. 1985. Ca. 212 S. Brosch. DM 32,80  
ISBN 3-88640-026-3

Die Buchbesprechungen dieser Titel erfolgt in den nächsten Ausgaben von „Compute mit“.

## Der C-16/C-116:

# Schlafender Riese im Zwergenlook!

Wie viele Besitzer eines C-16/C-116 mögen sich schon gefragt haben, warum das Einsteigermodell von Commodore keinen Userport besitzt?

Für den C-116 ist die Frage nach einem Blick in das Innere des Gehäuses leicht zu beantworten: Es fehlt einfach an Platz. Da man nicht noch mehr in der Bauart voneinander abweichende Geräte wollte, hat man den C-16 eben auch „hinten ohne“ gebaut.

Ein tiefer Blick in den Schaltplan auf den letzten Seiten der Bedienungsanleitung läßt jedoch hoffen. (Es gibt übrigens Bedienungsanleitungen, in denen der Schaltplan und die Beschreibung des Maschinenmonitors fehlt. In diesem Fall sollte der Händler aber in der Lage sein, ein anderes Exemplar zu beschaffen.)

Ein weiteres Ärgernis ist der magere Speicherplatz, der nur etwa ein Viertel des Speicherplatzes des C64 beträgt. Obwohl Betriebssystem und Basic im ROM bleiben, sind trotzdem keine umfangreichen Programme möglich.

Auch hier läßt sich jedoch preiswert Abhilfe schaffen, wenn man bereit und in der Lage ist, durch wenige Lötungen und Leiterbahnenunterbrechungen den Rechner

nachzurüsten.

So wurden folgende Erweiterungen an einem C-116 realisiert:

1. Anschluß eines Versatile Interface Adapters, der einen kompletten Eingabe/Ausgabe-Sektor auf einem Chip enthält (parallel/seriell, Interrupt-Steuerung etc.). Der Anschluß dieses Chips ist unabhängig von einer Speichererweiterung des Gerätes.
2. Einbau einer auf Original-Zustand umschaltbaren Speichererweiterung auf 64 Kbyte, davon 60 Kbyte frei.
3. Anschluß eines kleinen Interfaces, das den Gebrauch von zwei Datasetten erlaubt.  
Eine Datasette bleibt in LOAD-Position, d.h. die Play-Taste ist gedrückt, und die andere in SAVE-Position, d.h. die Record- und Play-Taste sind betätigt.

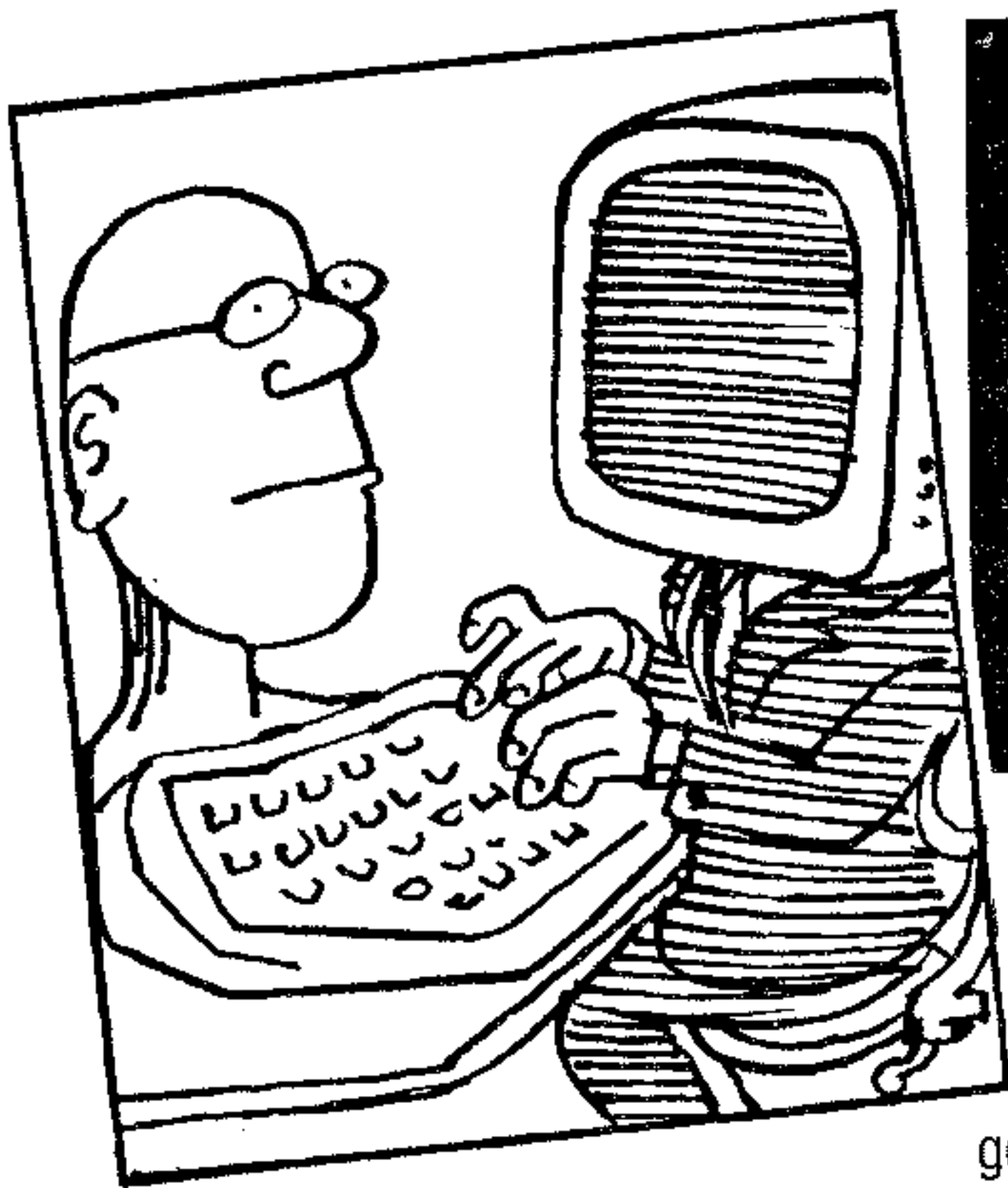
Bei den Befehlen LOAD bzw. SAVE startet automatisch die jeweils entsprechende Datasette. So kann man beispielsweise häufig benötigte Programme auf einer Endloskassette zu-

sammenstellen, welche dann im Gerät mit der LOAD-Position verbleibt. Das Abspeichern von Daten kann dann ohne Cassettenwechsel geschehen. Das Interface wird an den Cassettenausgang des Rechners angeschlossen und funktioniert ohne Eingriff in das Gerät. Dies ist eine preiswerte Alternative zur Diskette.

In den nächsten „Compute mit“-Ausgaben werden in loser Reihenfolge Erläuterungen und Anweisungen mit Schaltplänen zum Bau der oben beschriebenen Erweiterungen gegeben. Schreiben Sie uns bitte, welcher Erweiterung Ihr größtes Interesse gilt.

Da das Hardware-Angebot für dieses System äußerst spärlich ist, möchten wir versuchen, auf diese Weise eine Lücke zu schließen. Für den Anwender tritt dabei eine erfreuliche Begleiterscheinung auf, denn er kommt so an billige Hardwareerweiterungen und ist nicht mehr auf die Angebote der wenigen Händlerbetriebe angewiesen.

Michael Lausch



# Tips & Tricks

## zum richtigen Umgang mit Ihrem Computer

### COMFORT-LIST

Dieses Maschinen-Programm erleichtert Ihnen die Fehlersuche in Programmen, da die eigentliche List-Routine des C-16/116 wesentlich verbessert wird.

Nach dem Abtippen ist der Basiclader mit 'RUN' zu starten. Es wird automatisch an-

```

10 REM #####
15 REM # #
20 REM # COMFORT-LIST #
25 REM # BASICLADER #
30 REM # STARTADRESSE #
35 REM # SYS 1630 #
40 REM # #
45 REM # WRITTEN BY #
50 REM # GUNTER MASSAR #
55 REM # #
60 REM #####

```

```

65 :
70 SCNCLR: FORK=1630TO1730: READA: POKEK, A: IFK=1730THENB5
75 S=S+A: L=L+1: IFL=10THENREADP: ELSENEXTK
80 IFP<>STHEN90: ELSES=0: L=0: NEXTK
85 IFA=96THEN95
90 D=PEEK(64)*256+PEEK(63): CHAR1,8,10,"": PRINT"DATA FEHLER IN ZEILE"D": END
95 SCNCLR: CHAR1,8,6,"BITTE RETURN DRUECKEN !"
100 CHAR1,0,17,"": PRINT"S"; CHR$(34); "LIST/SYS1630"; CHR$(34); ",1,065E,06C3"
105 CHAR1,1,12,"": MONITOR
110 DATA162,105,142,6,3,162,6,142,7,3,738
115 DATA96,166,209,240,6,202,134,209,76,176,1514
120 DATA6,166,208,208,18,173,67,5,56,233,1140
125 DATA4,208,49,162,1,134,208,32,181,6,985
130 DATA76,153,6,173,67,5,56,233,4,208,981
135 DATA8,32,181,6,134,208,76,176,6,173,1000
140 DATA67,5,56,233,1,208,7,162,160,134,1033
145 DATA209,76,176,6,173,67,5,56,233,2,1003
150 DATA208,217,177,95,76,110,139,162,160,169,1513
155 DATA255,133,210,198,210,208,252,202,208,245,2121,96

```

gezeigt, wenn ein Data-Fehler vorliegt und in welcher Zeile er ist. Sind die Datas ohne Fehler, drücken Sie die 'RETURN-Taste' und schalten Sie Ihre Datasette ein. Jetzt wird das Maschinenprogramm abgespeichert.

Nun kann es jederzeit zu einem im Speicher befindlichen Programm mit der Eingabe von 'MONITOR' und 'L' hinzu geladen werden. Ist es geladen 'X' eingeben um zum Basic zurück zu kommen.

Starten können Sie COMFORT-LIST mit 'SYS 1630'.

Wird während dem Listen die 'CONTROL-Taste' betätigt, stoppt der Listvorgang und kann durch Druck auf die 'COMMODORE-Ta-

ste' fortgesetzt werden. Wird die 'COMMODORE-Taste' dauernd gedrückt, hat diese die gleiche Funktion wie beim normalen List. Drückt man die 'SHIFT-Taste', werden mehrere Zeilen gelistet.

Die Anzahl dieser Zeilen kann man durch 'POKE1698,0...255' ändern.

Wird die 'CONTROL-Taste' ein zweites mal betätigt, schaltet sich COMFORT-List wieder ab und kann jederzeit beim Listen erneut eingeschaltet werden.

Diskettenbesitzer ändern in Zeile 100 die '1' in eine '8' und schon kann COMFORT-List eingesetzt werden.

## Berechnung einer Prüfsumme für MC-Programme

Dieses kurze Listing berechnet die Prüfsumme für Maschinen-Programme. Es sollte eigentlich bei jeder Programmierung in Maschinensprache angewandt werden, da die Fehlersuche in solchen Programmen äußerst schwierig ist. Der Profi und der Anfänger wissen um diese Probleme und werden diese kurze Routine sehr schätzen.

```

11 ?„Eingabe der Adr. in Hex“:?
12 Input „Startadresse“;SS
13 Input „Ziel-Adresse“;TS
14 S = DEC (SS)
15 T = DEC (TS)
16 For N = S to T
17 R = R+Peek(N)
18 Next N
19 ?„Pruefsumme :“;R:End
    
```

Nachdem die Old- und Merge-Routine bereits in der Zeitschrift „CPU“ veröffentlicht wurde und auf den Programmpaketen POK 1/85 und POK 2/85 angeboten wurde, beschäftigen wir uns an dieser Stelle erneut mit diesem Thema.

So ist uns eine wesentlich vereinfachtere und kürzere Programmversion dieser beiden Routinen angeboten worden.

Hier entfällt das Eingeben der jeweiligen Basic-Zeilen gänzlich und das bedeutet Zeit- und Speicherplatzersparnis.

Ihnen als Programmierer müßten diese beiden Argumente eigentlich ausreichen, um sich nochmals näher mit diesen Routinen zu beschäftigen.

## Old- und Merge-Routine in wesentlich einfacherer Form:

### OLD-Routine

### Merge-Routine

```

1 Monitor
2 M 0226
3 > 0226 20 91 94 20 6B A8 38 A5
  > 022E 20 E9 02 AA A5 2E E9 00
  > 0236 A8 A9 00 20 D5 FF 20 18
  > 023E 88 20 4B 88 20 93 8A 4C
  > 0246 03 87 00 00 00 00 00 00
4 S "Merge", 1, 0226, 0248
5 V "Merge", 1
6 Monitor
7 L "Merge", 1
8 X
9 Laden des 1. Programms
10 SYS 550, "", 1
    
```

```

Monitor
M 00D8
> 00D8 A9 01 A8 91 2B 20 18 88
> 00E0 20 4B 88 20 93 8A 4C 03
> 00E8 87 00 00 00 00 00 00 00
S "OLD", 1, 00D8, 00E9
V "OLD", 1
-----
Monitor
L "OLD", 1
X
SYS 216
    
```

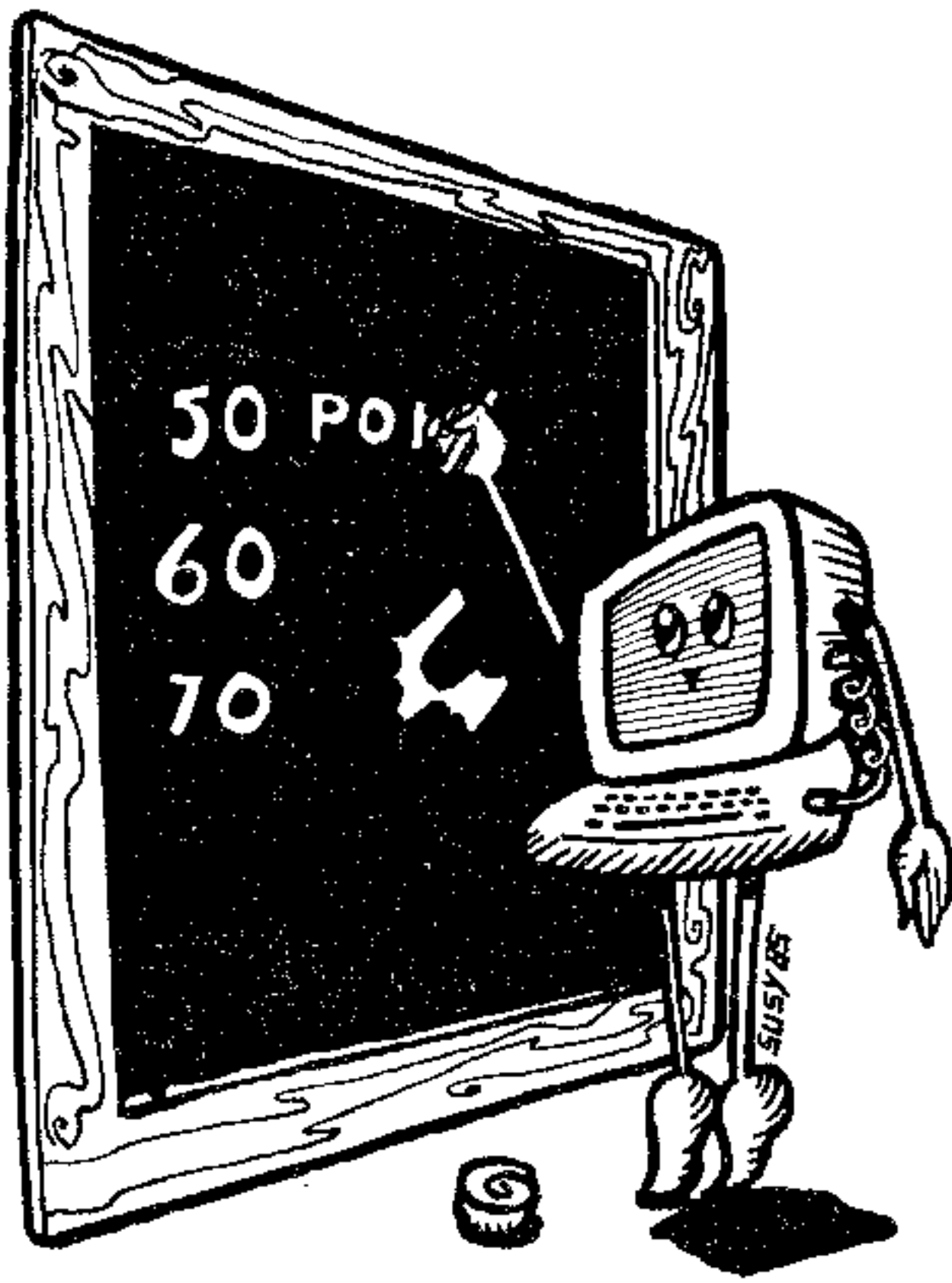
mit < S "Merge", 1, 0226, 0248 > auf Cassette,  
mit < S "Merge", 8, 0226, 0248 > auf Diskette.

↓                      ↓  
Startadresse,      Zieladresse + 1

5. V = Verify  
Vergleichen des Programms auf richtige Abspeicherung mit < V "Merge", 1 > für Cassette, mit < V "Merge", 8 > für Diskette.
6. Für die Anwendung muß es vom Tedmon-Monitor aus geladen werden! (Vermerk auf Cassette wichtig!)
7. Ladebefehl (L = Load), von Diskette mit < L "Merge", 8 >
8. Rücksprung in den Basic-Bereich.
9. Normales Laden des 1. Programms
10. Laden des 2. Programms, von Diskette mit < SYS 550, "", 8 > anstatt "" kann der Progr. Name des 2. Programms eingefügt werden.

1. Sprung in den Tedmon-Monitor.
2. Speicherbereich wird angewählt. Adr. Eingabe in Hex.
3. Die Inhalte des angewählten Speicherbereichs werden in Hex angezeigt.  
Sie müssen nun mit den Bytes des MC-Programms überschrieben werden.  
Am Ende jeder Zeile werden mit < Return > die geänderten Daten in den Speicher des Rechners übernommen.
4. S = Save  
Abspeichern

# POKE-Trickkiste



Dieser Programmbeitrag enthält verschiedene Pokes für Listschutz, Saveschutz u. a.

Wenn man POKE's für Save- und Listschutz verwendet, ist es am sinnvollsten, diese gleich an den Anfang des Programms zu setzen.

Die verwendeten POKE's werden durch den ersten Programmstart aktiviert und durch einen List- oder Saveversuch ausgelöst. Die Werte hinter den Kommastellen der Save-POKE's sind gleich den Werten der List-POKE's.

Die Bemerkungen hinter den POKES's zeigen an, was nach der Auslösung passiert. Durch die Bezeichnungen

„Bleibt“ oder „Stürzt“ wird darauf hingewiesen, ob nun der Computer durch das Drücken der Stop-Taste angehalten werden kann, oder ob dies nur noch durch Drücken der RESET-Taste möglich ist. Am unteren Bildschirmrand wird angezeigt, welche POKE's nun angezeigt werden.

Nach Drücken der Taste „W“ werden die POKE's weiter aufgelistet, bis „Tabellen-Ende“ zu sehen ist.

Wird Taste „A“ gedrückt, führt der Computer einen RESET durch.

Die Anfangsschrift wird durch Data-Zahlen erzeugt, da diese einfacher als Grafiken abzutippen sind.

```

0 REM #####
1 REM "
2 REM " POKE-TRICKKISTE "
3 REM "
4 REM " COPYRIGHT 1985 BY "
5 REM "
6 REM " GUNTER MASSAR "
7 REM "
8 REM #####
9 COLOR1,1:PRINTCHR$(19)CHR$(19):CLR:SCNCLR:COLOR0,2,4:COLOR4,2,4
10 VOL8:FORK=1000TO750STEP-25:SOUND1,K,1:NEXTK
11 FORK=1TO91:READA:POKEA,81:NEXTK:POKE0,15:FORK=1TO2000:NEXTK
12 SOUND1,800,10:SCNCLR:COLOR0,1:COLOR4,1:COLOR1,8,4
13 CHAR1,10,7,"POKE-TRICKKISTE FUER"
14 CHAR1,13,13,"COMMODORE 16"
15 FORK=1TO2000:NEXTK
16 COLOR1,5,3:A$="*COPYRIGHT (C) 1985 BY GUNTER MASSAR"
17 CHAR1,38,22,"*":FORX=1TO37:CHAR1,0,22,""
18 PRINTUSING"#####>";MID$(A$,1,X)
19 FORK=1TO50:NEXTK:NEXTX
20 FORK=1TO2000:NEXTK:COLOR0,2,4:COLOR4,7,5:COLOR1,1:SCNCLR:SOUND1,800,10
21 CHAR1,10,3,"LISTSCHUTZ"
22 CHAR1,10,7,"SAVESCHUTZ"
23 CHAR1,10,11,"TIPS&TRICKS"
24 CHAR1,10,15,"WARTER"
25 CHAR1,10,19,"ABBRUCH"
26 FOR K=1TO1000:NEXTK:CHAR1,7,23,"BITTE TASTE DRUECKEN":GOSUB169
27 SCNCLR:SOUND1,800,10
28 CHAR1,0,22,""
29 COLOR1,3,1:CHAR1,1,23,"LISTSCHUTZ SAVESCHUTZ TIPS&TRICKS"
30 CHAR1,1,24," WARTER ABRUCH "
31 COLOR1,7,1:POKE2021,21:PRINT" "
32 PRINT:PRINT "
33 PRINT:PRINT " POKE774,164/POKE775,242"
34 PRINT:PRINT " VERURSACHT -RESET-"
35 PRINT:PRINT " :COLOR1,1
36 PRINT:PRINT" POKE 774,0 ERSTE ZEILENNR. WIRD"
37 PRINT:PRINT" GELISTET/STUERZT AB"
38 PRINT:PRINT" POKE 774,27 WERDEN WIRRE BEFEHLE"
    
```

# tips & tricks

```

39 PRINT:PRINT"                GELISTET/STUERZT AB"
40 PRINT:PRINT" POKE    774,50  WERDEN WIRRE BEFEHLE"
41 PRINT:PRINT"                GELISTET/BLEIBT":GOSUB169
42 PRINT:PRINT" POKE    774,70  DAUERND WIRRE BEFEHLE"
43 PRINT:PRINT"                /STUERZT AB":GOSUB169
44 PRINT:PRINT" POKE    774,120 LISTET END/BLEIBT":GOSUB169
45 PRINT:PRINT" POKE    774,140 BILDSCHIRM VOLLER"
46 PRINT:PRINT"                STRICHE/STUERZT AB":GOSUB169
47 PRINT:PRINT" POKE    774,150 BILDSCHIRM VOLLER"
48 PRINT:PRINT"                END/STUERZT AB":GOSUB169
49 PRINT:PRINT" POKE    774,158 WIRRES BILD/BLEIBT":GOSUB169
50 PRINT:PRINT" POKE    775,200 BRINGT WIRRES BILD"
51 PRINT" POKE    774,0    /STUERZT AB":GOSUB169
52 PRINT:PRINT" POKE    775,190 WIRRES BILD/STUERZT AB":GOSUB169
53 PRINT:PRINT" POKE    775,136 LISTET NUR ZEILENNR.":GOSUB169
54 PRINT:PRINT" POKE   4098,3  BLOCKIERT -LIST-
55 PRINT:PRINT"                VERHINDERT NEUSTART":GOSUB169
56 COLOR1,7,1:PRINT:PRINT:PRINT:PRINT"
57 PRINT:PRINT"                POKE774,110/POKE775,139"
58 PRINT:PRINT"                HEBT LISTSCHUTZ AUF"
59 PRINT:PRINT"                ":GOSUB169
60 COLOR1,1:GOSUB169:PRINT:PRINT:PRINT"                TABELLEN ENDE":GOTO60
61 SCNCLR:SOUND1,800,10
62 CHAR1,0,22,"
63 COLOR1,3,1:CHAR1,1,23,"LISTSCHUTZ  SAVESCHUTZ  T-IPS&TRICKS"
64 CHAR1,1,24,"                NEW-EITER  AB-ABBRUCH                "
65 COLOR1,7,1:POKE2021,21:PRINT"
66 PRINT:PRINT:PRINT"
67 PRINT:PRINT"                POKE816,164/POKE817,242"
68 PRINT:PRINT"                VERURSACHT -RESET-"
69 PRINT:PRINT"                ":COLOR1,1
70 PRINT:PRINT:PRINT" POKE    0,0  DATASSETTE LAEUFT NICHT"
71 PRINT:PRINT:PRINT" POKE    0,15  NORMALFUNKTION":GOSUB169
72 COLOR1,7,1:PRINT:PRINT:PRINT"
73 PRINT:PRINT"                POKE816,164/POKE816,241"
74 PRINT:PRINT"                HEBT SAVESCHUTZ AUF"
75 PRINT:PRINT"                ":COLOR1,1:GOSUB169
76 PRINT:PRINT:PRINT"                TABELLEN ENDE":GOSUB169
77 GOTO76
78 SCNCLR:SOUND1,800,10
79 CHAR1,0,22,"
80 COLOR1,3,1:CHAR1,1,23,"LISTSCHUTZ  SAVESCHUTZ  T-IPS&TRICKS"
81 CHAR1,1,24,"                NEW-EITER  AB-ABBRUCH                "
82 COLOR1,1:POKE2021,21:PRINT"
83 PRINT:PRINT:PRINT:PRINT" POKE   153,0  STUERZT SOFORT AB"
84 PRINT:PRINT:PRINT" POKE   153,3  NORMAL"
85 PRINT:PRINT:PRINT" POKE   205,192 WIRRES BILD BEI -RUN-"
86 PRINT:PRINT"                /STUERZT AB"
87 PRINT:PRINT:PRINT" POKE   205,5  NORMAL":GOSUB169
88 COLOR1,7,1:PRINT:PRINT"
89 PRINT:PRINT:PRINT" WIRD IN DIE SPEICHERSTELLEN "
90 PRINT:PRINT" 4097/4098/45/46 DIE VOR DEM "
91 PRINT:PRINT" -RESET- O. -NEW- VORHANDENEN WERTE"
92 PRINT:PRINT" GEPOKT ,SO KANN DAS PROGRAMM "
93 PRINT:PRINT" ZURUECK GEHOLT WERDEN !"
94 PRINT:PRINT:PRINT"                ":COLOR1,1:GOSUB169
95 COLOR1,7,1:PRINT:PRINT:PRINT"
96 PRINT:PRINT" NACH -NEW- ODER -RESET- POKE 4098,10"
97 PRINT:PRINT" UND UNBENUTZTE ZEILENNR. EINGEBEN"
98 PRINT:PRINT" PROGRAMM IST WIDER VORHANDEN , KANN"

```



```

99 PRINT:PRINT" JEDOCH NICHT BEARBEITET WERDEN "
100 PRINT:PRINT" SONST STUERZT ES AB"
101 PRINT:PRINT:PRINT" _____":COLOR1,1:GOSUB169
102 PRINT:PRINT:PRINT" POKE 1344,0 SCHALTET REPEAT DER"
103 PRINT:PRINT" TASTEN AB"
104 PRINT:PRINT:PRINT" POKE 1344,128 NORMAL FUNKTION":GOSUB169
105 PRINT:PRINT:PRINT" POKE1345,X VERZOEGERUNG DER"
106 PRINT:PRINT" REPEATFUNKTION DER TASTATUR":GOSUB169
107 PRINT:PRINT:PRINT" PEEK (198) TASTATUR ABFRAGE":GOSUB169
108 PRINT:PRINT:PRINT" POKE 4096,1-255 BEI <RUN>,ERROR"
109 PRINT:PRINT:PRINT" POKE 4096,0 NORMAL FUNKTION":GOSUB169
110 PRINT:PRINT:PRINT" PEEK (1) DATASSETTE ABFRAGE"
111 PRINT:PRINT" 200=AUS
112 PRINT" 192=PLAY/REW./F.FWD"
113 PRINT" 208=PLAY+REC."
114 PRINT" 216=STOERUNG":GOSUB169
115 PRINT:PRINT:PRINT" POKE 205,ZEILENNR. POSISTIERT CURSOR":GOSUB169
116 PRINT:PRINT:PRINT" POKE 65287,70 ERZEUGT STREIFEN"
117 PRINT:PRINT:PRINT" POKE 65287,8 NORMAL":GOSUB169
118 PRINT:PRINT:PRINT" POKE 65299,40 GRIESELBILD"
119 PRINT:PRINT:PRINT" POKE 65299,208 NORMAL":GOSUB169
120 PRINT:PRINT:PRINT" POKE 65300,X WIRRES BILD"
121 PRINT:PRINT:PRINT" POKE 65300,10 BILD NORMAL":GOSUB169
122 PRINT:PRINT:PRINT" POKE 65290,0 STUERZT AB"
123 PRINT:PRINT:PRINT" CURSOR BLINKT":GOSUB169
124 PRINT:PRINT:PRINT" POKE 65297,X VERSCH. TOENE":GOSUB169
125 PRINT:PRINT:PRINT" POKE 65290,10 ERZEUGT 'BREAK'"
126 PRINT:PRINT:PRINT" STUERZT AB":GOSUB169
127 PRINT:PRINT:PRINT" POKE 65301,X BILDSCHIRMFARBE":GOSUB169
128 PRINT:PRINT:PRINT" POKE 65305,X RAHMENFARBE":GOSUB169
129 PRINT:PRINT:PRINT" POKE 65309,200 LAEST BILD LAUFEN":GOSUB169
130 PRINT:PRINT:PRINT" POKE 65287,40 BILDSCH. SCHWARZ":GOSUB169
131 PRINT:PRINT:PRINT" SYS 65529 ERZEUGT <RESET>":GOSUB169
132 COLOR1,7,1:PRINT:PRINT:PRINT" _____"
133 PRINT:PRINT:PRINT" POKE 2022,Z/POKE 2023,S"
134 PRINT:PRINT" LINKE OBERE ECKE DES BILDFENSTERS"
135 PRINT:PRINT:PRINT" POKE 2021,Z/POKE 2024,S"
136 PRINT:PRINT" RECHTE UNTERE ECKE DES BILDFENSTERS"
137 PRINT:PRINT:PRINT" PRINTCHR$(19)CHR$(19)"
138 PRINT:PRINT" SCHALTET FENSTER AB"
139 PRINT:PRINT:PRINT" _____":COLOR1,1:GOSUB169
140 PRINT:PRINT:PRINT" POKE 65286,PEEK(65286)AND239"
141 PRINT:PRINT" SCHALTET BILDSCHIRM AUS":GOSUB169
142 PRINT:PRINT:PRINT" POKE 65286,PEEK(65286)OR16"
143 PRINT:PRINT" SCHALTET BILDSCHIRM EIN":GOSUB169
144 PRINT:PRINT:PRINT" POKE2039,1-255 BLOCKIERT -CTRL-/-S-"
145 PRINT:PRINT" BEIM LISTEN"
146 PRINT:PRINT" POKE2093,0 NORMALFUNKTION":GOSUB169
147 PRINT:PRINT:PRINT" POKE240,1-255 WARTET BIS EINE TASTE"
148 PRINT:PRINT" GEDRUECKT WIRD":GOSUB169
149 PRINT:PRINT:PRINT" PEEK(1347) ABFRAGE FUER"
150 PRINT:PRINT" -SHIFT-/-CTRL-/-COMMODORE-"
151 PRINT:PRINT" 1=SHIFT"
152 PRINT" 2=COMMODORE"
153 PRINT" 3=SHIFT+COMMODORE"
154 PRINT" 4=CTRL"
155 PRINT" 5=SHIFT+CTRL"
156 PRINT" 6=COMMODORE+CTRL":GOSUB169
157 PRINT:PRINT:PRINT" POKE1351,128 BLOCKIERT UMSCHALTUNG"
158 PRINT:PRINT" VON GROSS- AUF "

```

```
159 PRINT:PRINT"                KLEINSCHREIBEMODUS"  
160 PRINT:PRINT"                ODER UMGEKEHRT"  
161 PRINT:PRINT:PRINT" POKE1351,0  NORMALFUNKTION":GOSUB169  
162 COLOR1,7,1:PRINT:PRINT:PRINT"  
163 PRINT:PRINT" POKE19,5  SCHALTET -INPUT-"  
164 PRINT:PRINT" FRAGEZEICHEN AUS . NACH DER INPUT-"  
165 PRINT:PRINT" ABFRAGE MUSS JEDOCH NORMAL WERT "  
166 PRINT:PRINT" 19,0 EINGEPOCKT WERDEN"  
167 PRINT:PRINT"                ":COLOR1,1:GOSUB169  
168 GOSUB169:PRINT:PRINT:PRINT"  TABELLEN ENDE":GOTO168  
169 A=PEEK(198)  
170 IFA=42THEN27  
171 IFA=13THEN61  
172 IFA=22THEN78  
173 IFA=10THENSYS65529  
174 IFA=9THENRETURN  
175 GOTO 169  
176 DATA3359,3360,3361,3362,3398,3403,3438,3443,3478,3483,3518,3523,  
177 DATA3558,3559,3560,3561,3562,3598,3638,3678,3718  
178 DATA3367,3368,3369,3370,3406,3411,3446,3451,3486,3491,3526,3531,3566,3571  
179 DATA3606,3611,3646,3651,3686,3691,3727,3728,3729,3730  
180 DATA3374,3379,3414,3418,3454,3457,3494,3496,3534,3535,3574,3575  
181 DATA3614,3616,3654,3657,3694,3698,3734,3739  
182 DATA3382,3383,3384,3385,3386,3387,3422,3462,3502,3542,3582,3583,3584,3585  
183 DATA3586,3587,3622,3662,3702,3742,3743,3744,3745,3746,3747
```

## Werkstatt:

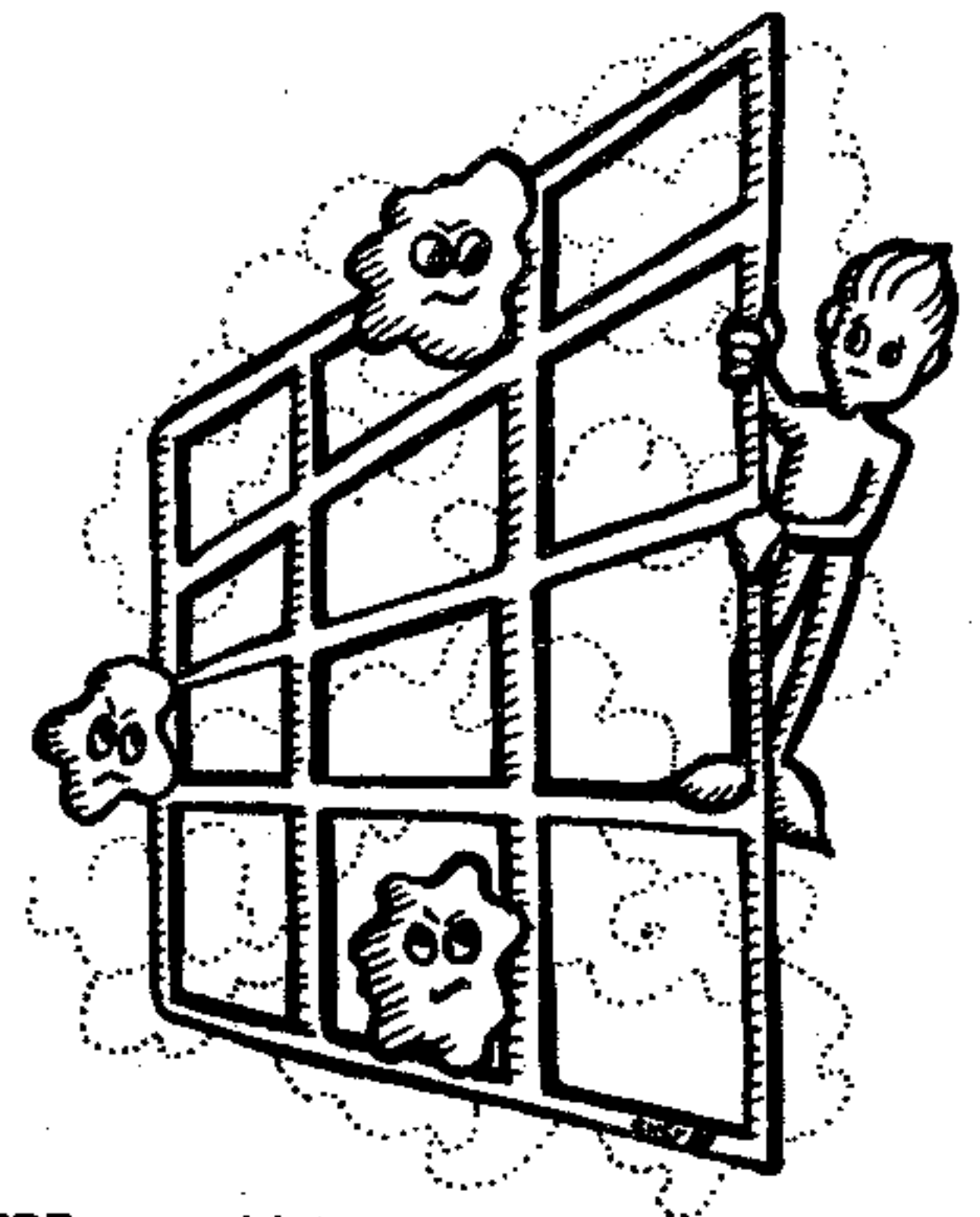
# WINDOW-MANAGER

*Diese Routine ermöglicht den schnellen Wechsel von einem Window in das Nächste*

Laut Handbuch ist es möglich, durch die Befehle ESC-T und ESC-B Fenster auf dem Bildschirm des C-16 zu definieren. Jedoch ist es sehr umständlich, aus einem Fenster in ein anderes zu gelangen. Für diesen Vorgang sind ellenlange ESC-Sequenzen nötig. Um die Umständlichkeit dieses Vorganges abzukürzen, schuf der Autor diese Routine, die es ermöglicht, ohne großen Aufwand von einem definierten Fenster in ein anderes zu springen.

Bei diesem Windowmanager handelt es sich um einen Basic-Lader, der Datazeilen

liest und die Werte im Tape-Puffer ablegt. Gestartet und initialisiert wird er durch **SYS819**. Durch diesen Aufruf werden die 16 definierbaren Fenster auf Gesamtgröße gestellt (wie ESC-N ohne SCNCLR). Will man nun ein Fenster definieren, so schreibt man DEF POS n,x1,y1,x2,y2. n ist die Fensternummer und erlaubt Werte von 1 bis 16. x1,y1 ist die Spalten-Zeilen-Position links oben und x2,y2 entsprechend rechts unten. Diese Parameter haben folgende Grenzen:  $0 < x1 \leq x2 < 41$  und  $0 < y1 \leq y2 < 26$ . Werden die Grenzen nicht eingehalten, so wird ILL. QUANTITY



ERR. gemeldet.

Ein Beispiel: DEF POS 3,2,6,38,19 definiert Fenster 3 als ein Feld von Spalte 2/Zeile 6 bis Spalte 38/Zeile 19. So werden nun die Fenster definiert, aber noch nicht aufgerufen. Dazu benutzt man den Befehl POS n, wobei n wiederum die Fensternummer darstellt.

Mit POS 5 z. B. wird ein Fenster aufgerufen, welches zuvor mit DEF POS 5, ... definiert worden ist. Nach dem Aufruf wird innerhalb des neuen Fensters noch ein HOME ausgeführt, damit man einen definierten Ausgangspunkt hat. Bei POS n ist auch 0 (Null) für n zugelassen. Durch den Be-

fehl POS 0 z. B. wird ein ESC-N ohne SCNCLR ausgeführt, der gesamte Schirm also als Fenster definiert.

Das gesamte ML-Programm benötigt ca. 130 Byte und noch einmal ca. 70 Byte als Variablen. Die Befehle dürfen nur innerhalb eines Programms stehen, im Direkt-

Modus verursachen sie einen Syntax-Error. Da die Routine im Tape-Puffer liegt, sei gesagt, daß keine Daten auf Band geschrieben oder gelesen werden können. Für den PLUS/4-Anwender sollte auch die RS232-Schnittstelle tabu sein, da sonst die Fenster-Daten futsch sind.

```

100 REM*****
110 REM*
120 REM* W I N D O W M A N A G E R *
130 REM*
140 REM* VERSION 1.03 FUER C16 & CO *
150 REM*
160 REM* VERWALTET 16 FENSTER UEBER *
170 REM*
180 REM* 'DEF POS N,X1,Y1,X2,Y2'
190 REM*
200 REM* ZUR DEFINITION DER GROESSE *
210 REM*
220 REM* UND 'POS N' ZUM AUFRUFEN
230 REM*
240 REM* N IST DIE FENSTERNUMMER;
250 REM*
260 REM* (X1,Y1) DIE POSITION LINKS
270 REM*
280 REM* OBEN; (X2,Y2) DIE POSITION
290 REM*
300 REM* RECHTS UNTEN; N IST ZU-
310 REM*
320 REM* LAESSIG VON 1 BIS 16
330 REM*
340 REM* (AUSNAHME: POS 0 DEFINIERT
350 REM*
360 REM* DEN GESAMTEN SCHIRM ALS
370 REM*
380 REM* FENSTER); GRENZEN FUER
390 REM*
400 REM* DIE KOORDINATEN:
410 REM*
420 REM* 0<X1<41; 0<Y1<26
430 REM*
440 REM* X1<=X2<41; Y1<=Y2<26;
450 REM*
460 REM* NACH EINEM AUFRUF DURCH
470 REM*
480 REM* 'POS N' WIRD EIN 'HOME'
490 REM*
500 REM* AUSGEFUEHRT.
510 REM*
511 REM*
520 REM*
530 REM* IM APRIL '85 VON
540 REM*
542 REM* WOLFGANG W. WIRTH

```

```

543 REM*
544 REM* HELLWIGSTRASSE 3
545 REM*
546 REM* 6600 SAARBRUECKEN
547 REM*
548 REM* TEL: 0681-61690
556 REM*
560 REM*****
570 REM
580 FOR I=819 TO 997:READ J$
590 POKE I,DEC(J$):NEXT:SYS 819:END
600 DATA0,40,D0,09,BD,E2,03,99,E6,03
610 DATA0,10,02,A2,03,88,10,F2,A2,BF
620 DATA0,03,8E,08,03,8C,09,03,60,20
630 DATAD8,9D,CA,E0,28,90,F7,4C,1C,99
640 DATA20,DB,9D,CA,E0,19,80,F5,60,20
650 DATAB1,9D,E0,11,B0,ED,8A,0A,0A,60
660 DATA20,64,03,A8,A2,03,B9,E2,03,9D
670 DATAE5,07,C8,CA,10,F6,20,80,DE,4C
680 DATA9A,D8,20,64,03,F0,CE,8D,2A,04
690 DATA20,50,03,8E,28,04,20,5B,03,8E
700 DATA27,04,20,50,03,EC,28,04,90,B7
710 DATABE,29,04,20,5B,03,EC,27,04,90
720 DATAAC,8E,26,04,A2,03,AC,2A,04,BD
730 DATA26,04,99,E2,03,C8,CA,10,F6,60
740 DATA20,73,04,20,CE,03,4C,DC,8B,20
750 DATA79,04,4C,25,8C,F0,EE,C9,B9,F0
760 DATA9B,C9,96,D0,F0,20,73,04,C9,B9
770 DATAF0,A6,4C,9D,9A,27,00,00,18

```

## Demo-Programm

```

10 REM WINDOWMANAGER - DEMO
20 :
22 REM FENSTER DEFINIEREN
23 :
30 DEF POS 1,3,3,6,10
40 DEF POS 2,20,8,25,12
50 :
60 POS 1:REM 1. FENSTER AUFRUFEN
66 FOR I= 1 TO 80:PRINT RND(1);:NEXT
76 :
80 POS 2:REM 2: FENSTER AUFRUFEN
86 FOR I= 1 TO 80:PRINT RND(1):NEXT
90 GOTO 60

```

## Grafik-Beispiele

Das folgende Programm dient allen Programmierern als Unterstützung im Umgang mit den Grafikbefehlen des C-16/116. Daß man mit Hilfe dieses Systems eine hervorragende Grafik erzielen kann, dürfte mittlerweile jedem Anwender bekannt sein. Oft wissen die Anwender aber nicht, mit welchen Befehlen die Grafik zu ermöglichen ist. Hier haben Sie die Möglichkeit, anhand des Programmes die Befehle zu studieren und bei selbsterstellten Programmen einzusetzen.

```

10 REM *****
20 REM *
25 REM *   VERSCHIEDENE GRAFIK   *
30 REM *
35 REM *           BEISPIELE           *
40 REM *
45 REM *   VON F. LOHMANN   *
50 REM *
55 REM *****
60 GRAPHIC3,1:COLOR0,1:COLOR4,7,3
65 FORA=0TO179 STEP4
70 BOX3,160,100,1,100,A
75 NEXTA
80 FORX=0TO1500:NEXTX
85 REM *****
90 REM *****   ZIEL FERNROHR   *****
95 REM *****
100 COLOR1,2:COLOR4,1:GRAPHIC1,1
110 CIRCLE1,160,100,20
115 CIRCLE1,160,100,50
120 CIRCLE1,160,100,90
125 DRAW1,160,189TO160,10
130 DRAW1,250,100TO70,100
135 FORJ=0TO1500 : NEXT
140 REM *****
145 REM ***** MEHR-FARBEN-GRAFIK *****
150 REM *****
155 REM ***** 1.GRAFIK *****
160 COLOR2,7,3:GRAPHIC 3,1
165 CIRCLE2,90,78,9,9,5:PAINT2,90,78
170 FORA=0TO178STEP8:COLOR3,6,3

```

```

175 BOX3,80,100,100,55,A:NEXTA
180 FORA=0TO178STEP8:COLOR2,3,3
185 BOX2,50,100,130,55,A:NEXTA
190 FORJ=0TO1500 : NEXT
195 REM ***** 2.GRAFIK *****
200 SCNCLR
205 FORA=0TO95STEP8:COLOR2,6,2
210 CIRCLE2,80,100,A,A,,,120:NEXTA
215 FORA=0TO1500:NEXTA
220 REM ***** 3.GRAFIK *****
225 SCNCLR:COLOR2,7,3:COLOR4,6,3
235 FORA=0TO94STEP8
240 CIRCLE2,80,100,A,A,,,90:NEXTA
245 FORX=0TO1500 : NEXT X
250 REM ***** 4.GRAFIK *****
255 SCNCLR:COLOR2,12,3:COLOR4,9,3
260 FORA=0TO180STEP8
265 BOX2,80,130,80,55,A:NEXTA
270 FORA=0TO170STEP8
275 BOX3,130,180,30,10,A:NEXTA
280 FORX=0TO1500 : NEXT X
285 REM ***** 5.GRAFIK *****
290 COLOR1,8,4:GRAPHIC1,1
295 FORA=50TO100STEP4
300 CIRCLE1,160,100,A,A,,,45
305 NEXT A
310 FORX=9TO30STEP4
315 CIRCLE1,30,30,X,X,,,45:NEXTX
320 FORX=9TO30STEP4
325 CIRCLE1,290,30,X,X,,,45:NEXTX
330 FORX=9TO30STEP4
335 CIRCLE1,30,170,X,X,,,45:NEXTX
340 FORX=9TO30STEP4
345 CIRCLE1,290,170,X,X,,,45:NEXTX
350 CHAR1,17,9," C 16 "
355 CHAR1,17,11,"GRAFIK"
360 CHAR1,17,13," IST "
365 CHAR1,16,15,"EINFACH"
370 FORA=0TO1600:NEXTA
375 REM ***** 6.GRAFIK *****
380 SCNCLR
385 FORA=140TO160STEP5:FORX=50TO100STEP8
390 CIRCLE1,A,100,X: NEXTX,A
395 FORX=0TO1500:NEXTX
400 REM ***** 7.GRAFIK *****
405 SCNCLR:FORA=0TO600STEP5
410 BOX1,100,90,A,10,A:NEXTA
415 FORA=0TO1500:NEXTA:GRAPHIC0

```

## SOFTSCROLL

**SOFTSCROLL** ist eine hervorragende Erweiterung für Ihr System. Da der C-16/116 nicht über Befehle wie etwa 'LEFT' oder 'RIGHT' verfügt, ist es für den Ersteller eines Programms, das einen erweiterten

Bildschirm simuliert, unmöglich, etwa eine Bewegung darzustellen. Dies ist aber bei einer großen Anzahl von Spielen von Wichtigkeit, da die Hardware sonst nur ein Scrollen nach oben oder unten zuläßt (sie-

he Ausnutzung der Windowmöglichkeiten). **SOFTSCROLL** beinhaltet beide vermißten Befehle und ist sogar in der Lage, eine ruckfreie (soft) Verschiebung vorzunehmen. Nun ist eine Bewegung in alle vier Himmelsrichtungen leicht programmierbar.

Tippen Sie bitte einfach das Basic-Ladeprogramm (siehe Listing) ab. Es erzeugt ein reines Assemblerprogramm in dem Bereich \$3F5A-\$3FEE. Dieses wird abso-

lut gespeichert und kann mit **LOAD 'SOFTSCROLL',8,1** platzsparend an den oberen Teil des Basicspeichers gelegt werden. Als erstes wird jetzt die Initialisierungsroutine mit **SYS 16218** aufgerufen. Sie schaltet das Bildschirmformat auf 38 Zeichen pro Zeile um und schützt das Assemblerprogramm vor ungewolltem Überschreiben mit Variablen oder Text. Dann kann jederzeit ein Aufruf der eigent-

lichen Hauptroutine erfolgen. Dies geschieht in der Form **SYS 16247, Parameter**. Der Parameter kann die Werte **0** oder **1** haben, andere werden mit einem **SYNTAX-ERROR** abgelehnt. Für ein Scrollen nach links verwenden Sie bitte Parameter **0**, für scrollen nach rechts Parameter **1**. Nun erfolgt eine Verschiebung um genau einen Punkt in die gewünschte Richtung. Dies entspricht genau einem Achtel der

Buchstabenmatrix. Hierfür verantwortlich zeichnen die Bits **0-2** des Ted-Registers **7**. Bei dem achten Aufruf der Routine wird der Originalzustand wieder hergestellt und dann in die Hardscrollroutine verzweigt. Diese kann (aus Zeitgründen) auch direkt aufgerufen werden: **SYS 16275 (links)** oder **SYS 16323 (rechts)**. Jetzt entfällt aber die gleitende Bewegung, die Verschiebung wirkt ruckartig.

```

0 REM *****
1 REM *
2 REM *      S C R O L L - R O U T I N E      F U E R   D E N   C O M M O D O R E   C 1 6 / 1 1 6
3 REM *
4 REM *      B E L E G T E R   S P E I C H E R B E R E I C H   :   V O N   $ 3 F 5 A   B I S   $ 3 F E E
5 REM *      L A D E N   D E S   A S S E M B L E R C O D E S   :   L O A D " S C R O L L   $ 3 F 5 A " , 8 , 1   B Z W .   , 1 , 1
6 REM *      S T A R T      1 )   S Y S   1 6 2 1 8           I N I T I A L I S I E R U N G
7 REM *                      2 )   S Y S   1 6 2 4 7 , X     S C R O L L A U F R U F   M I T   P A R A M E T E R
8 REM *
9 REM *      W R I T T E N   0 2 . 8 6      B Y      J U E R G E N   O T T O
10 REM *****
11 :
12 POKES1,89:POKES2,63:POKES5,89:POKES6,63
13 :
100 DATA AD,07,FF : REM LDA #$FF07 ; ACCU MIT TEDREGISTER #07 LADEN
101 DATA 29,F7 : REM AND #$7F ; BIT 3 LOESCHEN
102 DATA 8D,07,FF : REM STA $FF07 ; AUF 38 ZEICHEN UMSCHALTEN
103 DATA A9,59 : REM LDA #$59 ; LO-BYTE PROGRAMMANFANG
104 DATA 85,33 : REM STA $33 ; STRINGSPEICHERENDE HERABSETZEN
105 DATA 85,37 : REM STA $37 ; BASICSPERICHERENDE HERABSETZEN
106 DATA A9,3F : REM LDA #$3F ; HI-BYTE PROGRAMMANFANG
107 DATA 85,34 : REM STA $34 ; STRINGSPEICHERENDE HERABSETZEN
108 DATA 85,38 : REM STA $38 ; BASICSPERICHERENDE HERABSETZEN
109 DATA A9,00 : REM LDA #$00 ; LO-BYTE SCHIEBEADRESSE
110 DATA 85,D0 : REM STA $D0 ; LO-BYTE INITIALISIEREN
111 DATA A9,0C : REM LDA #$0C ; HI-BYTE SCHIEBEADRESSE
112 DATA 85,D1 : REM STA $D1 ; HI-BYTE INITIALISIEREN
113 DATA 60 : REM RTS ; INITIALISIERUNGSRoutine BEENDET
114 DATA 20,D8,9D : REM JSR $9DD8 ; GETCOM + GETBYT
115 DATA E0,00 : REM CPX #$00 ; TEST AUF PARAMETER 0
116 DATA F0,07 : REM BEQ $3F85 ; VERZWEIGUNG SCROLL LINKS
117 DATA E0,01 : REM CPX #$01 ; TEST AUF PARAMETER 1
118 DATA F0,31 : REM BEQ $3FB3 ; VERZWEIGUNG SCROLL RECHTS
119 DATA 4C,A9,94 : REM JMP $94A9 ; VERZWEIGUNG 'SYNTAX ERROR'
120 DATA AD,07,FF : REM LDA $FF07 ; ACCU MIT SCHIEBEREGISTER LADEN
121 DATA F0,04 : REM BEQ $3F8E ; WENN GRENZE ERREICHT, DANN SCROLL SCREEN
122 DATA CE,07,FF : REM DEC $FF07 ; SONST SOFTSCROLL LINKS
123 DATA 60 : REM RTS ; ROUTINE SCROLL LINKS BEENDET
124 DATA A9,07 : REM LDA #$07 ; STARTWERT IN ACCU LADEN
125 DATA 8D,07,FF : REM STA $FF07 ; UND IN SCROLLREGISTER SCHREIBEN
126 DATA A2,19 : REM LDX #$19 ; ZAEHLREGISTER AUF 25 ZEILEN SETZEN
127 DATA A0,00 : REM LDY #$00 ; SPALTENZAehler AUF NULL SETZEN
128 DATA B1,D0 : REM LDA ($D0),Y ; LINKEN BILDSCHIRM RAND IN ACCU
129 DATA 85,D2 : REM STA $D2 ; UND ZWISCHENSPEICHERN
130 DATA C8 : REM INY ; EINE SPALTE WEITER NACH RECHTS
131 DATA B1,D0 : REM LDA ($D0),Y ; NAECHSTES ZEICHEN IN ACCU LADEN
132 DATA 8B : REM DEY ; EINE SPALTE NACH LINKS RUECKEN

```

# werkstatt

```

133 DATA 91,D0      : REM STA ($D0),Y ; UND IN NEUER POSITION SPEICHERN
134 DATA C8         : REM INY          ; WIEDER ZURUECK NACH RECHTS
135 DATA C0,28      : REM CPY #$28         ; TESTE SPALTENZAehler AUF SPALTENENDE
136 DATA D0,F5      : REM BNE $3F9B        ; WENN NEIN,DANN ZUM NAECHSTEN ZEICHEN
137 DATA A5,D2      : REM LDA $D2          ; SONST WEGGESCHOBENES ZEICHEN RETTEN
138 DATA 88         : REM DEY           ; AUF RECHTEN RAND POSITIONIEREN
139 DATA 91,D0      : REM STA ($D0),Y ; UND DORT ANFUEGEN
140 DATA 20,E1,3F   : REM JSR $3FE1        ; EINE ZEILE WEITERZAEHLEN
141 DATA D0,E5      : REM BNE $3F95        ; ZEILE 25 ERREICHT ? BEI NEIN WEITER
142 DATA 4C,6E,3F   : REM JMP $3F6E        ; FERTIG, ZURUECK ZUR INITIALISIERUNG
143 DATA AD,07,FF   : REM LDA $FF07        ; ACCU MIT SCROLLREGISTER LADEN
144 DATA C9,07      : REM CMP #$07         ; TEST AUF AEUSSERE GRENZE
145 DATA F0,04      : REM BEQ $3FBE        ; WENN ERREICHT, DANN SCROLL SCREEN
146 DATA EE,07,FF   : REM INC $FF07        ; SONST SOFTSCROLL RECHTS
147 DATA 60         : REM RTS           ; ROUTINE SOFTSCROLL RECHTS BEENDET
148 DATA A9,00      : REM LDA #$00         ; SCROLLREGISTERLOESCHUNG VORBEREITEN
149 DATA 8D,07,FF   : REM STA $FF07        ; UND REGISTER ZURUECKSETZEN
150 DATA A2,19      : REM LDX #$19         ; ZAEHLREGISTER AUF 25 ZEILEN SETZEN
151 DATA A0,28      : REM LDY #$28         ; SPALTENZAehler AUF RECHTEN RAND STELLEN
152 DATA B1,D0      : REM LDA ($D0),Y     ; RECHTEN BILDSCHIRM RAND IN DEN ACCU
153 DATA 85,D2      : REM STA $D2          ; UND ZWISCHENSPEICHERN
154 DATA 88         : REM DEY           ; EINE SPALTE WEITER NACH LINKS
155 DATA B1,D0      : REM LDA ($D0),Y     ; NAECHSTES ZEICHEN IN DEN ACCU LADEN
156 DATA C8         : REM INY          ; EINE SPALTE NACH RECHTS RUECKEN
157 DATA 91,D0      : REM STA ($D0),Y     ; UND IN NEUER POSITION SPEICHERN
158 DATA 88         : REM DEY           ; WIEDER ZURUECK NACH LINKS
159 DATA D0,F7      : REM BNE $3FCB        ; SPALTENANFANG NICHT ERREICHT, WEITER
160 DATA A5,D2      : REM LDA $D2          ; WEGGESCHOBENES ZEICHEN RETTEN
161 DATA C8         : REM INY          ; AUF LINKEN RAND POSITIONIEREN
162 DATA 91,D0      : REM STA ($D0),Y     ; UND DORT ANFUEGEN
163 DATA 20,E1,3F   : REM JSR $3FE1        ; EINE ZEILE WEITERZAEHLEN
164 DATA D0,E7      : REM BNE $3FC5        ; ZEILE 25 ERREICHT ? BEI NEIN WEITER
165 DATA 4C,6E,3F   : REM JMP $3F6E        ; FERTIG, ZURUECK ZUR INITIALISIERUNG
166 DATA A5,D0      : REM LDA $D0         ; LO-BYTE DER SCHIEBEADRESSE LADEN
167 DATA 18         : REM CLC           ; ADDITION VORBEREITEN
168 DATA 69,28      : REM ADC #$28         ; UM 40 ( EINE ZEILE ) ERHOEHEN
169 DATA 85,D0      : REM STA $D0         ; UND WIEDER ZURUECKSCHREIBEN
170 DATA 90,02      : REM BCC $3FEC        ; WENN WERT < 255 DANN WEITER
171 DATA E6,D1      : REM INC $D1         ; SONST HI-BYTE DER S.ADRASSE ERHOEHEN
172 DATA CA         : REM DEX           ; ZEILENZAEHLREGISTER UM 1 WEITERSETZEN
173 DATA 60         : REM RTS           ; UNTERROUTINE ZUENDE, ZURUECK
174 :
176 REM EINLESEN DES MASCHINENPROGRAMMS
177 :
178 : FOR I=0 TO 147
179 :   READ A$
180 :   A=DEC(A$)
181 :   B=B+A
182 :   POKE I+16218,A
183 : NEXT I
184 :
185 IF B=20537 THEN 190 : REM O.K.
186 PRINT "☹ SIE HABEN EINEN FEHLER BEIM ABTIPPEN"
187 PRINT "☹ DER DATAWERTE GEMACHT.BITTE KORRIGIEREN"
188 PRINT "☹ SIE DIE ZEILEN 100 BIS 173."
189 END
190 PRINT "☺ DAS ASSEMBLERPROGRAMM WIRD JETZT"
191 PRINT "☹ ABGESPEICHERT.LEGEN SIE BITTE EINE"
192 PRINT "☹ DISKETTE EIN.DRUECKEN SIE EINE TASTE."
193 GETKEY A$

```

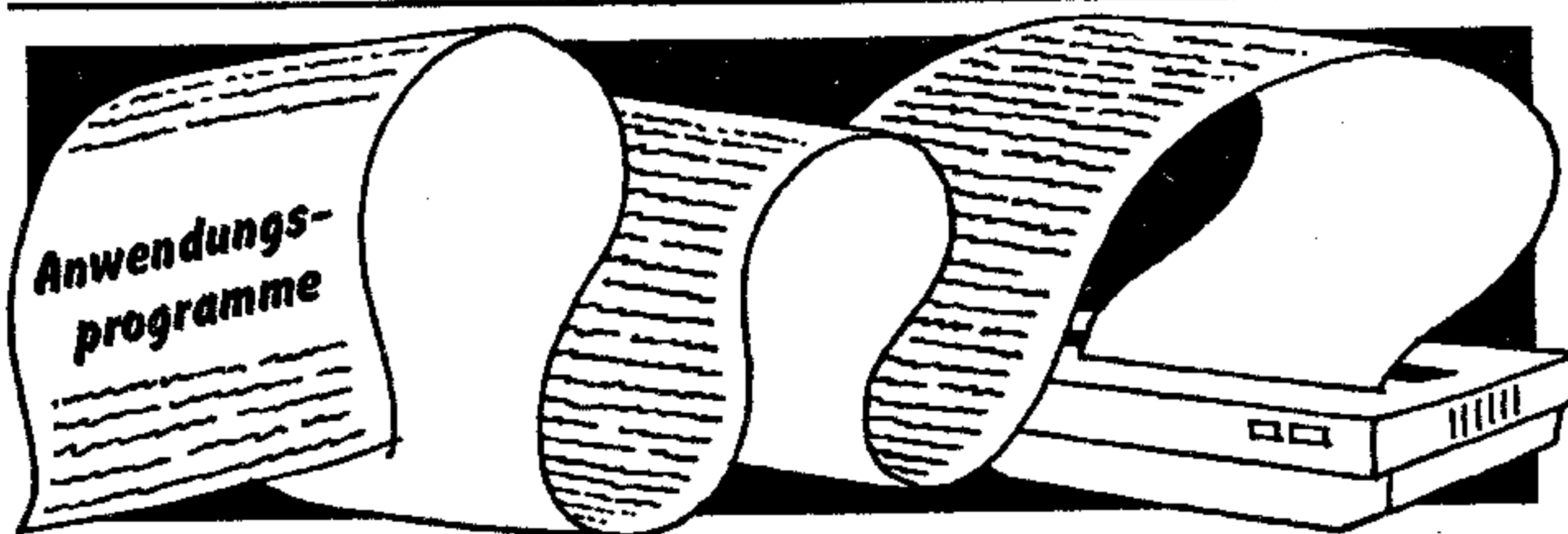
## Schon gewußt . . . ?

Nach dem Einschalten sind die Funktionstasten F2 & F5 mit DLOAD" bzw. DSAVE" belegt. Wenn Sie nun vor dem Drücken dieser beiden Tasten die ESC-Taste betätigen, erscheint an den Anfang der aktuellen Zeile der Befehl LOAD" und SAVE". Dadurch sind diese Tasten auch für den Kassettenbetrieb nutzbar.

```

194 SYS 43115 "SOFTSCROLL",8 : REM DATASTE : .... ,1
195 POKE 38,90 : POKE 39,63 : POKE 2035,238 : POKE 2036,63 : POKE 2034,38
196 SYS 65496
197 PRINT "☐ DAS PROGRAMM IST ABGESPEICHERT."
198 PRINT " BITTE LADEN SIE ES IN ZUKUNFT MIT : "
199 PRINT "☐ LOAD 'SOFTSCROLL',8,1 BZW .... ,1,1☐"
200 PRINT "☐ DAS ENDE DES BASICSPEICHERS IST NACH"
201 PRINT " EINMALIGEM AUFRUF VON SYS 16218 GEGEN"
202 PRINT " UEBERSCHREIBEN GESCHUETZT."
203 PRINT "☐ AUFRUF DER SCROLLROUTINE : SYS 16247,X"
204 PRINT TAB(10) "1) SCROLL LINKS : ☐1☐"
205 PRINT TAB(10) "2) SCROLL RECHTS : ☐2☐"
206 PRINT "☐ VERWENDETE ZEROPAGESPEICHERSTELLEN : ☐"
207 PRINT TAB(10) "1) $D0 = !208"
208 PRINT TAB(10) "2) $D1 = !209"
209 PRINT TAB(10) "3) $D2 = !210"

```



## Ligatabelle

**Erstellen sie Ihre eigene aktuelle  
Bundesligatabelle!**

Mit Hilfe dieses Programmbeitrages kann man auf einfache Weise Sportligatabellen erstellen und diese dann ohne große Mühen auf den aktuellsten Stand bringen. Nach dem Programmstart mit „RUN“, erfolgt zuerst eine kurze Anleitung und daran anschließend eine Menuetabelle, wo man unter folgenden Operationen wählen kann:

### Modul 1: Eingabe der Tabelle

Mit diesem Modul kann man jede beliebige Sportligatabelle bis maximal 20 Mannschaften eingeben. Achten Sie darauf, daß im Programm zuerst die erhaltenen Tore einzugeben sind (normalerweise zuerst die geschossenen Tore!) Die Begründung dafür liegt in der Programmierung des Sortiermoduls.

### Modul 2: Speichern der Tabelle

Hier muß man eine Nummer und einen Namen für die Datei eingeben, „PLAY“ und „RECORD“ drücken, und schon wird die Tabelle gespeichert.

### Modul 3: Laden der Tabelle

Hier sind ebenfalls die gleichen Informationen einzugeben. Anschließend ist die Taste „PLAY“ auf der Datasette zu drücken – die Tabelle wird geladen.

### Modul 4: Tabelle aktualisieren

Man gibt die erzielten und die erhaltenen Tore jeder Mannschaft ein, und schon wird die Tabelle auf den neuesten Stand gebracht.

### Modul 6: Tabelle ändern

Mit Hilfe dieses Moduls kann man eventuell auftretende Fehler bei der Eingabe korrigieren. Dies geschieht folgendermaßen:

Man gibt den Tabellenplatz der betreffenden Mannschaft und die Kennziffer des falschen Tabellenwertes (also z. B. Anzahl der Spiele, Anzahl der Siege etc.) ein. Die bereits erwähnten Kennziffern werden vorher in einer Tabelle zur Auswahl ausgegeben.

### Modul 7: Arbeit beenden

Mit der Kennziffer 7 beendet man den Programmablauf. Außerdem enthält dieses Programm, wie bereits erwähnt, ein Sortiermodul.

# programme

```
2 REM *****
3 REM *
4 REM *          LIGATABELLE
5 REM *          -----
6 REM *
7 REM *          WRITTEN BY STEFAN MESSNER
8 REM *
9 REM *          RATZENDORFERSTR. 9
10 REM*          A-9063 MARIA SAAL
11 REM*          TEL.: 04223/370 (2370)
12 REM*****

20 FOR I=0 TO 6
30 READ MA$(I),KZ$(I)
40 L(I)=35-LEN(MA$(I))
50 NEXT I
60 FOR I=0 TO 7
70 READ BI$(I)
80 NEXT I

300 PRINT"┌":PRINT"└", "┌┐ URZE ANLEITUNG ┌┐"
310 PRINT"┌┐┐┐ IT DIESEM PROGRAMM KOENNEN SIE EINE"
320 PRINT"BELIEBIGE LIGATABELLE ERSTELLEN UND SIE STAENDIG AKTUALISIEREN."
330 PRINT"┌┐┐┐ AUS EINEM MENUE WAEHLN SIE DIE          VERSCHIEDENEN OPERATIONEN
    AUS."
340 GOSUB 20000
350 DIM S(25,10)
360 DIM S$(25,10)
370 DIM M$(25)
500 REM
510 PRINT"┌"
520 PRINT"└", "┌┐ MENUEAUSWAHL ┌┐"
540 PRINT:PRINT:PRINT
550 FOR TT=0 TO 6
560 PRINT MA$(TT);
570 FOR PP=1 TO L(TT)
580 PRINT".";
590 NEXT PP
600 PRINT "┌┐"KZ$(TT)"┌┐"
610 PRINT
620 NEXT TT
630 PRINT"┌┐┐┐ ITTE WAEHLN WIE AUS !"
640 GETKEY AW$

650 IF AW$="1" THEN 1000: ELSE IF AW$="2" THEN 3000: ELSE IF AW$="3" THEN 4000
660 IF AW$="4" THEN 5000: ELSE IF AW$="5" THEN 6000: ELSE IF AW$="6" THEN 7000
670 IF AW$="7" THEN END: ELSE 640
1000 PRINT"┌┐ EINGABE DER TABELLE"
1020 PRINT"┌┐┐┐ EBEN WIE DIE GEWUENSCHTE TABELLE EIN"
1030 PRINT"VERGESSEN SIE NICHT, JEDE EINGABE MIT '┌┐' ABZUSCHLIESSEN !"
1040 GOSUB 20000
1050 PRINT"┌┐ EBEN WIE DIE ANZAHL DER ANNSCHAFTEN EIN ";:INPUT WA
1060 IF WA>20 THEN 1070: ELSE 1090
1070 PRINT"┌┐┐┐ EINGABE VON MAXIMAL 20 ANNSCHAFTEN          ERLAUBT !":FOR T=1 TO 20
00:NEXTT
1080 GOTO 1050
1090 ZA=WA-1
1100 INPUT"┌┐ ATUM";DA$
1110 FOR I=0 TO ZA
1120 PRINT"┌"
1130 PRINT I+1","
1140 INPUT"┌┐ ANNSCHAFT";M$(I)
1150 IF LEN(M$(I))>8 THEN 1140
1160 PRINT"┌┐┐┐"
1170 FOR J=0 TO 7
1180 PRINT BI$(J);TAB(30);:INPUT S(I,J)
1190 NEXT J
1200 NEXT I
1210 GOSUB 20000
2000 PRINT"┌┐┐┐ TABELLE SORTIEREN┌┐"
2020 PRINT"┌┐┐┐ IE TABELLE WIRD NUN SORTIERT !"
2030 PRINT"┌┐ ITTE ETWAS TUE !"
2040 FOR T=1 TO 1000:NEXTT
2050 Y=7
2060 FOR I=0 TO ZA-1
2070 FOR J=I+1 TO ZA
2080 IF S(I,Y)>S(J,Y) THEN 2130: ELSE IF S(I,Y)<S(J,Y) THEN GOSUB 10000:GOTO 213
0
2090 Y=Y-1
2100 IF Y<>4 THEN 2080
2110 IF M$(I)<M$(J) THEN 2130
2120 GOSUB 10000
2130 Y=7
2140 NEXT J
2150 NEXT I
2160 GOTO 500
3000 PRINT"┌┐┐┐ TABELLE SPEICHERN┌┐"

3020 PRINT"┌┐ EBEN WIE FUER DIE DATEI EINE NOCH NICHTBELEGTE NUMMER UND EINEN NAMEN EIN."
3030 INPUT NU,NA$
3040 PRINT:PRINT:PRINT"┌┐ ASSETTE EINLEGEN !"
```



```

3050 PRINT"| ASTEN ' _ _ _ ' UND ' L * I ' DRUECKEN"
3060 OPEN NU,1,2,NA$
3070 PRINT# NU,ZA
3080 PRINT# NU,DA$
3090 FOR I=0 TO ZA
3100 PRINT#NU,M$(I)
3110 FOR J=0 TO 7
3120 WW=S(I,J)
3130 PRINT#NU,WW
3140 NEXT J
3150 NEXT I
3160 CLOSE NU
3170 PRINT" L - IE - ATEI IST GESCHLOSSEN"
3180 FOR T=1 TO 1000:NEXTT
3190 GOTO 500
4000 PRINT" L | ABELLE LADEN"

4020 PRINT"| EBEN *IE DIE /UMMER UND DEN /AMEN EIN, UNTER DER DIE GEWUENSCHTE -
ATEI"
4030 PRINT"GESPEICHERT IST."
4040 INPUT NU,NA$
4050 PRINT:PRINT:PRINT"/ ASSETTE EINLEGEN !"
4060 PRINT"| ASTE ' L * I ' DRUECKEN !"
4070 OPEN NU,1,0,NA$
4080 INPUT#NU,ZA
4090 INPUT#NU,DA$
4100 FOR I=0 TO ZA
4110 INPUT#NU,M$(I)
4120 FOR J=0 TO 7
4130 INPUT#NU,WW
4140 S(I,J)=WW
4150 NEXT J
4160 NEXT I
4170 CLOSE NU
4180 PRINT" L - IE - ATEI IST GESCHLOSSEN !"
4190 FOR T=1 TO 1000:NEXTT
4200 GOTO 500
5000 PRINT" L | ABELLE AKTUALISIEREN"

5020 PRINT" L | EBEN *IE NUN DIE GESCHOSSENEN UND DIE ERHALTENEN | ORE JEDER \
ANNSCHAFT EIN "
5030 FOR T=1 TO 2000:NEXTT:PRINT" L "
5040 PRINT"- ATUM AENDERN !";:INPUT DA$
5050 PRINT" L | OEVIEVIELE \EISTERSCHAFTSRUNDEN WOLLEN *IE EINGEBEN ";:INPUT RZ
5060 PRINT" L ": PRINT ,"GESCHOSSEN","ERHALTEN"
5070 FOR Q=1 TO RZ
5080 FOR I=0 TO ZA
5090 PRINT" L | M$(I)":",:INPUT T1 :PRINT" L ",,,:INPUT T2
5100 S(I,0)=S(I,0)+1:S(I,4)=S(I,4)+T2:S(I,5)=S(I,5)+T1:S(I,6)=S(I,6)+(T1-T2)
5110 IF T1>T2 THEN 5120: ELSE IF T1<T2 THEN 5140: ELSE 5160
5120 S(I,1)=S(I,1)+1:S(I,7)=S(I,7)+2
5130 GOTO 5170
5140 S(I,3)=S(I,3)+1
5150 GOTO 5170
5160 S(I,2)=S(I,2)+1:S(I,7)=S(I,7)+1
5170 NEXT I
5180 NEXT Q
5190 GOTO 2000
6000 PRINT" L | ITTE ETWAS | EDULD !"
6010 FOR I=0 TO ZA
6020 FOR J=0 TO 7
6030 S$(I,J)=STR$(S(I,J))
6040 NEXT J
6050 NEXT I
6060 PRINT" L | ABELLENSTAND VOM "DA$
6070 PRINT" L "
6080 FOR I=0 TO ZA
6090 PRINT I+1 TAB(3);M$(I) TAB(12);S$(I,0) TAB(15);S$(I,1) TAB(18);S$(I,2);
6100 PRINT TAB(21);S$(I,3) TAB(26);S$(I,5) TAB(29);": " TAB(28);S$(I,4);
6110 PRINT TAB(36);S$(I,7)
6120 NEXT I
6130 PRINT"| ASTE DRUECKEN !":GETKEY F$
6140 GOTO 500
7000 PRINT" L | ABELLE AENDERN"
7020 PRINT" L | ALLS *IE EINEN -EHLER BEI DER -INGABE ODER AEHNLICHES GEMACHT
HABEN,"
7030 PRINT"KOENNEN SIE ES NUN KORRIGIEREN."
7040 GOSUB 20000
7050 PRINT" L | EBEN *IE BITTE DEN | ABELLENPLATZ DER BETREFFENDEN \ANNSCHAFT EI
N !"
7060 PRINT" L | OENN SIE UNSICHER SIND UND | ABELLENPLATZNICHT MEHR WISSEN,"
7070 PRINT"HABEN *IE JETZT | ELEGENHEIT, DIE | ABELLE ZU SEHEN."

```

# programme

```
7080 PRINT"—RUECKEN ♥IE 'J' ODER 'N'"
7090 INPUT AN$: IF AN$="J" THEN 6000: ELSE IF AN$="N" THEN 7100 : ELSE 7090
7100 INPUT "| ABELLENPLATZ"; PL
7110 PL=PL-1
7120 PRINT" "M$(PL)
7130 PRINT" "
7140 FOR J=0 TO 7
7150 LL(J)=35-LEN(BI$(J))
7160 PRINT BI$(J);
7170 FOR PP=1 TO LL(J)
7180 PRINT ". ";
7190 NEXT PP
7200 PRINT J
7210 NEXT J
7220 PRINT"—RUECKEN ♥IE NUN DIE ENTSPRECHENDE 'ENN-"
7230 PRINT"ZIFFER DES FALSCHEN | ABELLENWERTES !"
7240 INPUT KF
7250 IF KF<0 OR KF>7 THEN 7240
7260 PRINT" "M$(PL)
7270 PRINT" "
7280 PRINT BI$(KF); TAB(30); : INPUT S(PL,KF)
7290 PRINT" "DEITERE ♥ENDERUNGEN ? —RUECKEN ♥IE"
7300 PRINT"'J' ODER 'N'"
7310 INPUT AN$: IF AN$="J" THEN 7050: ELSE IF AN$="N" THEN 2000: ELSE 7310
10000 FOR X=0 TO 7
10010 Z=S(I,X): S(I,X)=S(J,X): S(J,X)=Z
10020 NEXT X
10030 Z$=M$(I): M$(I)=M$(J): M$(J)=Z$
10040 RETURN
20000 PRINT" "ASTE DRUECKEN !"
20010 GETKEY A1$
20020 RETURN
30000 DATA "—INGABE: | ABELLE", "1"
30010 DATA "| ABELLE SPEICHERN", "2", "LADEN DER | ABELLE", "3"
30020 DATA "| ABELLE AKTUALISIEREN", "4", "| ABELLEN AUSGABE AUF | ILDSCHIRM", "5"
30030 DATA "♥ENDERUNGEN", "6", "ARBEIT BEENDEN", "7"
30040 DATA "♥PIELANZAHL", "♥IEGE", "_EMIS", "IEDERLAGEN", "—RHALTENE | ORE"
30050 DATA "| ESCHOSSENE | ORE", "| ORDIFFERENZ", "—UNKTE"
40000 END
```

## SUPERDAT

— Dateiverwaltung für den C-16 mit Floppy

### Besondere Vorzüge dieses Programms:

- bis 30 Datensätze auf einmal im Speicher
- frei wählbare Eingabemaske
- Suchen nach Begriffen in beliebigen Feldern
- Ausdruck der Datensätze möglich
- Einfache, übersichtliche Bedienung
- Integriertes Diskettenhilfsprogramm

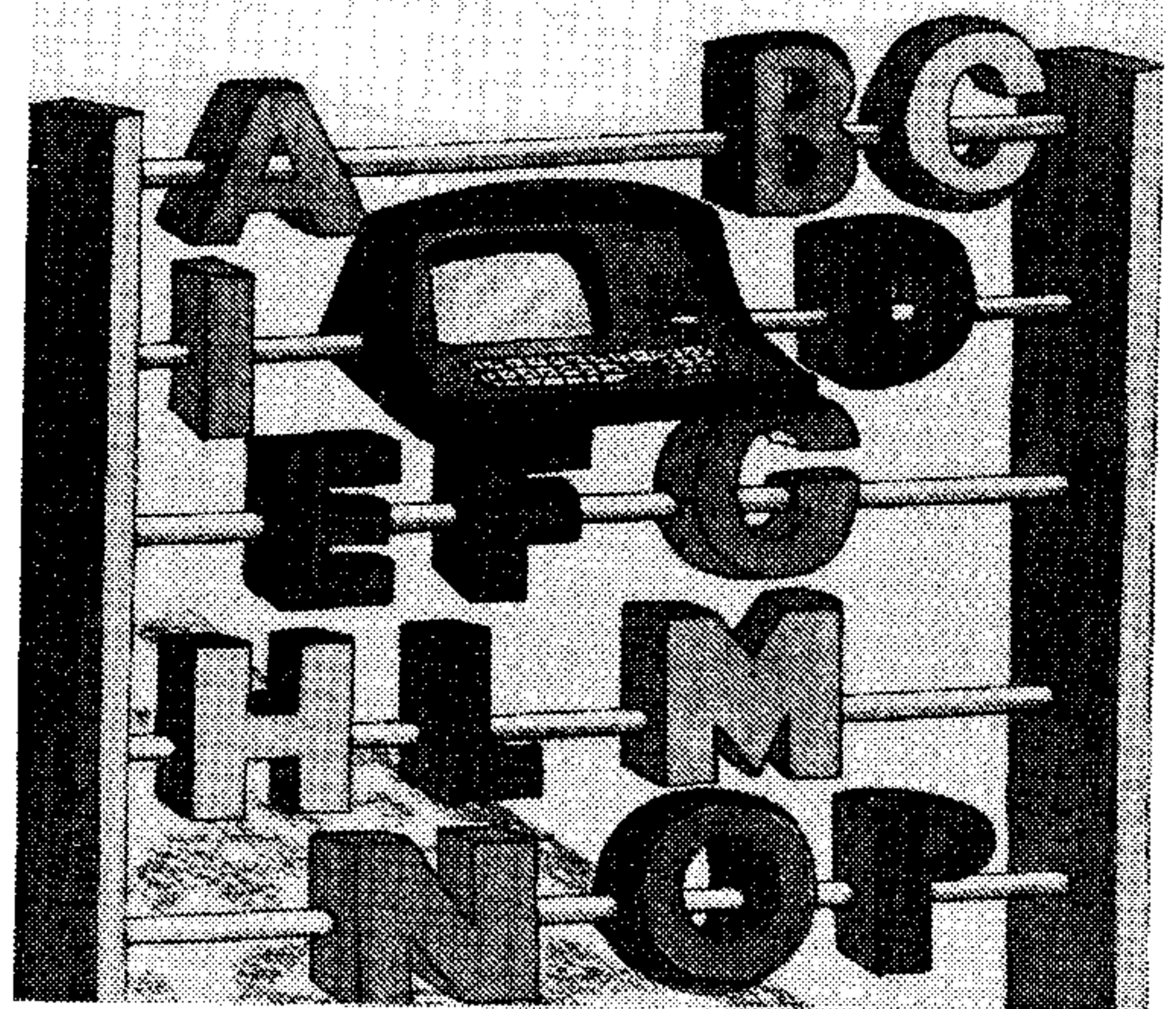
Mit diesem Programm wurde versucht, möglichst viele nützliche Funktionen so unterzubringen, daß sie einfach zu bedienen sind. Das Programm wird daher über 4 Parallelmenüs gesteuert, unter denen man mit den Funktionstasten F1-F3 und HELP beliebig hin- und herschalten kann.

### Beschreibung der einzelnen Menüs und Funktionen:

#### 1. Datei (F1)

##### a) Daten eingeben . . .

Den 8 Feldern der Eingabemaske können nun Begriffe zugeordnet werden (z. B. Name: Meier o. ä.)



## b) Daten ändern . . .

Der Computer fragt Sie, ob Sie nach einem Begriff im 1. Feld der Maske suchen wollen. Falls ja, geben Sie diesen ein, falls nein, drücken Sie RETURN. Hat der Computer den Datensatz (z. B. eine Adresse gibt er ihn Zeile für Zeile aus, und Sie können jeden Begriff editieren und/oder mit RETURN übernehmen.

## c) Daten ausgeben . . .

Wie unter b) sucht der Computer Ihren Datensatz und gibt ihn auf dem Bildschirm aus. Nun wird nach dem Ausdrucken gefragt. Nur bei Druck auf „J“ aktiviert der Computer den Drucker.

Als zusätzliche Option kann bei b) und c) auf die Frage nach dem Suchbegriff auch „\*“ eingegeben werden. Es werden dann alle Datensätze hintereinander ausgegeben. Auf die Frage „weiter?“ kann der Vorgang mit „N“ abgebrochen werden.

## d) Datei laden . . .

Eine abgespeicherte Datei wird mit der zugehörigen Maske geladen (keine Koordinationsprobleme Maske-Datensätze!). Vorher muß man den Filenamen angeben.

## e) Datei speichern . . .

Die im Speicher befindliche Datei wird mit der zugehörigen Maske nach Eingabe des Filenamens sequentiell abgespeichert.

## f) Datenspeicher löschen . . .

Nachdem die Sicherheitsabfrage mit „J“ beantwortet wurde, wird der Datenspeicher gelöscht. Die Maske bleibt erhalten.

## 2. Diskette (F2)

Dieses Menue ist das integrierte Diskettenhilfsprogramm.

### a) Inhaltsverzeichnis . . .

Der Computer holt das Directory der eingelegten Diskette

### b) Files löschen . . .

Nach Eingabe des entsprechenden Filenames wird ein File oder ein Programm gescratcht

### c) Files umbenennen . . .

Dazu müssen Sie den jetzigen und den neuen Filenamen eingeben

### d) Disk aufräumen . . .

Validieren der Diskette (schließen aller Files etc.)

### e) Disk formatieren . . .

Es kann mit oder ohne ID auf den einzugebenden Namen formatiert werden

## 3. Maske (F3)

Die z. Zt. aktivierte Maske wird angezeigt.

### a) Maske ändern . . .

Die Belegung der 8 möglichen Felder wird Zeile für Zeile ausgegeben und kann editiert und/mit RETURN übernommen werden. 16 Zeichen pro Feld werden angenommen. Mit HELP setzen Sie das Ende der Maske (d. h. wenn Sie im 6. Feld z. B. HELP drücken, besteht die Maske aus den ersten 5 Feldern).

### b) Maske speichern . . .

Die derzeitige Maske wird z. B. für spätere Verwendungszwecke sequentiell abgespeichert. Vorher ist der Filename einzugeben.

### c) Maske laden . . .

Hiermit kann man nach Eingabe des Filenamens zum einen eine mit b) abgespeicherte Maske laden, oder aber die

Maske eines best. Datenfiles, getrennt von seinen Datensätzen, in den Speicher holen. Der Datenspeicher bleibt in beiden Fällen unverändert.

## 4. Farbwahl (HELP)

Die Hintergrund- und Rahmenfarbe kann nach pers. Geschmack über die Cursorarten verändert werden. Dabei bedeutet:

Cursor links . . . Rahmenfarbe verändern

Cursor rechts . . . Hintergrundfarbe verändern

Cursor up . . . Luminanz (bzw. Farbnummer) um 1 erhöhen

Cursor down . . . Luminanz (bzw. Farbnummer) um 1 erniedrigen

## Allgemeine Informationen:

- Hat der Computer eine erteilte Aufgabe erfüllt, wartet er stets auf einen Tastendruck. Bei Arbeiten mit der Peripherie meldet er OK.
- Mit der Stoptaste kann das Programm jederzeit (außer bei einem Input-Befehl+ unterbrochen werden. Das Programm kehrt dann an den Anfang zurück, wo man zwischen den 4 Menüs wählen kann. Alle Daten bleiben erhalten.
- Mit ESC/Q kann das Programm von jedem Menü aus beendet werden.
- Das Programm ist relativ absturzsicher, d. h. alle Fehler werden aufgefangen, und nach der Fehlermeldung wird das Programm fortgesetzt. Sollten Sie jedoch einmal von einer Funktionstaste auf eine andere abrutschen, kann es geschehen, daß der Computer mit READY aussteigt. Das Programm kann in diesem Falle aber mit GOTO 240 ohne Datenverlust fortgesetzt werden.
- Nach dem Start des Programms ist die Maske so ausgelegt, daß das Programm als Adressenverwaltung genutzt werden kann. Durch Änderung der Maske sind natürlich unendlich viele Einsatzmöglichkeiten, wie beispielsweise Platten-, Video-, Bücher-Programmverwaltungen etc. möglich.
- Aus Speicherplatzgründen dürfen die REMs nicht mit eingegeben werden.

```

10 REM  REM'S  N I C H T  E I N G E B E N  !
20 REM=====
30 REM=====
40 REM===      S U P E R D A T      ===
50 REM=====
60 REM===  (C) 1985 BY TIM SCHLUETER  ===
70 REM===      POLLENWEG 3          ===
80 REM===      4576 BERGE           ===
90 REM=====
100 REM=====
110 FOR I=1 TO 8: READ N: KEY I, CHR$(N): NEXT: DATA 133, 137, 134, 138, 135, 139, 136, 140: GOTO 180
120 REM===  MENUESCHALTUNG (F1-HELP)  ===

```

# programme

```
130 POKE1373,0:S=ASC(K$):IFS=133GOTO280
140 IFS=137GOTO670:ELSEIFS=134GOTO1210
150 IFS=140GOTO1460
160 IFS=27THENGETKEYK$:IFK$="Q"THEN1660
170 ONMGOTO330,730,1250,1500,250
180 DIMD$(8,50):TRAP1600:FH(4)=2:FH(0)=2:LH(0)=4:COLOR4,1:COLOR0,2,4:COLOR1,1:GO
SUB1070
190 REM===== ANFANG =====
200 PRINTCHR$(14)CHR$(8)"[CLS]"TAB(13)"[CLS] , 7 - _ - + |":POKE1344,0:M=5
210 PRINT" (C) 1985 BY | IM SCHLUETER":FORI=1TO40:PRINT"█";:NEXT
220 PRINT"█ [1] █-ATEI █ [3] █\ASKE █";
230 PRINT"█ [2] █-ISKETTE █ [4] █HELP █-ARBWAHL █";:FORI=1TO40:PRINT"█";:NE
XT
240 CLOSE1:PRINT"#####"CHR$(27)"T"
250 PRINT"#####|ITTE WAEHLEN█"
260 GETKEYK$:GOTO130
270 REM===== DATEI =====
280 M=1:K$="#####-":PRINT"█"K$"1-█ -ATEN EINGEBEN"
290 PRINTK$"2-█ -ATEN AENDERN":PRINTK$"3-█ -ATEN AUSGEBEN"
300 PRINTK$"4-█ -ATEI LADEN":PRINTK$"5-█ -ATEI SPEICHERN"
310 PRINTK$"6-█ -ATENSPEICHER LOESCHEN"
320 GETKEYK$:GOTO130
330 A=VAL(K$):IFA<1ORA>6GOTO320
340 ONAGOTO390,1110,1110,540,610,1040
350 REM==== DATEI IM RECHNER ? =====
360 IFAZ=0THENPRINT"##### /EINE -ATEN IM _ECHNER !█":GETKEYK$:GOTO280
370 RETURN
380 REM ===== EINGABE/AENDERUNG =====
390 AZ=AZ+1:Z=AZ:IFAZ>50THENPRINT"##### ZUVIELE -ATEN !":GOTO460
400 SCNCLR:IFA=2THENJ=Z+1:GOSUB360
410 PRINT:FORY=1TOAM:PRINTE$(Y);:IFA=1THEN430
420 PRINT"█";O$(Y,Z)CHR$(141)"Q";:FORX=1TOLEN(E$(Y))-1:PRINT"█";:NEXTX
430 O$(Y,Z)="" :PRINT"█";:INPUTO$(Y,Z):NEXTY
440 IFA=2THENRETURN
450 A=0:FORI=1TOAM:IFO$(I,Z)=""THENA=A+1:NEXTI:IFA=AMTHENAZ=AZ-1
460 GETKEYK$:GOTO280
470 REM ===== AUSGABE =====
480 GOSUB360:J=Z+1:PRINT"█"
490 FORX=1TOAM:PRINTE$(X)": [CLS]O$(X,Z):NEXT
500 PRINTTAB(11)"#####AUSDRUCK? (J/N)":GETKEYK$:IFK$<>"J"THENRETURN
510 OPEN1,4:PRINT#1:FORX=1TOAM:PRINT#1,CHR$(17):PRINT#1,E$(X)": ";O$(X,Z):NEXT:
CLOSE1
520 PRINT"█ [CLS] █.█":RETURN
530 REM ===== LADEN =====
540 GOSUB1560:PRINT"#####LADE -ATEI "CHR$(34)F$CHR$(34)
550 OPEN1,8,0,F$+",S,R":INPUT#1,AM:FORI=1TOAM:INPUT#1,N$(I):NEXTI
560 INPUT#1,AZ:FORX=1TOAZ
570 FORI=1TOAM:INPUT#1,O$(I,X):NEXTI,X
580 IFDS<>0THENPRINT"##### "+DS$+" ERROR !█":CLOSE1:GETKEYK$:GOTO280
590 PRINT"█ [CLS] █.█":CLOSE1:GETKEYK$:GOSUB1090:GOTO280
600 REM ===== SPEICHERN =====
610 GOSUB360:GOSUB1560:PRINT"##### SPEICHERE -ATEI ALS "CHR$(34)F$CHR$(34)
620 OPEN1,8,1,"e:"+F$+",S,W":PRINT#1,AM:FORX=1TOAM:PRINT#1,N$(X):NEXTX
630 PRINT#1,AZ:FORX=1TOAZ
640 FORY=1TOAM:PRINT#1,O$(Y,X):NEXTY,X
650 CLOSE1:PRINT"█ [CLS] █.█":GETKEYK$:GOTO280
660 REM ===== DISKETTE =====
670 M=2:K$="#####-
680 PRINT"█"K$"1-█ \NHALTSVERZEICHNIS":PRINTK$"2-█ -ILES LOESCHEN"
690 PRINTK$"3-█ -ILES UMBENENNEN"
```



# programme

```
1260 ONAGOTO1280,1410,1360
1270 REM===== MASKE AENDERN =====
1280 SCNCLR:PRINT"MASKE AENDERN (|_L_ = _NDE)█":KEYB,"_" +CHR$(13)
1290 FORI=1TO8:K$=N$(I):PRINT"█(16)█"N$(I)CHR$(141)"█";:INPUTN$(I)
1300 IFLEN(N$(I))>16THENN$(I)=LEFT$(N$(I),16)
1310 IFLEFT$(N$(I),1)="_"GOTO1340
1320 NEXTI
1330 KEYB,CHR$(140):PRINT"█":AM=I-1:GOSUB1090:GOTO1210
1340 IFI=1THENN$(1)=K$:I=0:GOTO1280:ELSE1330
1350 REM===== MASKE SPEICHERN =====
1360 GOSUB1560
1370 PRINT"█SPEICHERE \ASKE ALS "CHR$(34)F$CHR$(34)"█"
1380 OPEN1,8,1,"@:"+F$+",S,W":PRINT#1,AM:FORI=1TOAM:PRINT#1,N$(I):NEXT:CLOSE1
1390 PRINT"█":GETKEYK$:GOTO1210
1400 REM===== MASKE LADEN =====
1410 GOSUB1560:PRINT"█LADE DIE \ASKE "CHR$(34)F$CHR$(34)"█"
1420 OPEN1,8,0,F$+",S,R":INPUT#1,AM:FORI=1TOAM:INPUT#1,N$(I):NEXT:CLOSE1
1430 IFDS<>0THENPRINT"█"DS$"-RROR !█":GETKEYK$:GOTO1210
1440 PRINT"█":GETKEYK$:GOSUB1090:GOTO1210
1450 REM===== FARBWahl =====
1460 M=4:PRINT"█-URSORTASTEN"
1470 PRINT"█-ARBE +/- => UP/DOWN"
1480 PRINT"█LAHMEN/INTERGRUND => LINKS/RECHTS"
1490 GETKEYK$:GOTO130
1500 IFK$="█"THENLH(F)=LH(F)-1:IFLH(F)<0THENLH(F)=7:FH(F)=FH(F)-1
1510 IFK$="█"THENLH(F)=LH(F)+1:IFLH(F)>7THENLH(F)=0:FH(F)=FH(F)+1
1520 IFK$="█"THENF=4:ELSEIFK$="█"THENF=0
1530 IFFH(F)<1THENFH(F)=16:ELSEIFFH(F)>16THENFH(F)=1
1540 COLORF,FH(F),LH(F):GOTO1490
1550 REM===== FILENAMEEINGABE =====
1560 F$="":INPUT"█ -ILENAME (16)":F$:IFLEN(F$)>16ORF$=""GOTO1560
1570 PRINT"█ -ISK EINLEGEN UND _|_/_/ DRUECKEN !█"
1580 GETK$:IFK$=CHR$(13)THENRETURN:ELSE1580
1590 REM===== ERROR'S =====
1600 TRAP1600:IFER<>5THEN1620
1610 PRINT"█ -LOPPY IST NICHT ANGESCHALTET !":RESUMENEXT
1620 IFER=23THENPRINT"█ /AME ZU LANG !":RESUME
1630 IFER<>30GOTO1650
1640 PRINT"█"TAB(15)"#ABBRUCH !":CLOSE1:FORI=1TO800:NEXT:GOTO250
1650 CLOSE1:PRINTCHR$(9)CHR$(142)"█"ERR$(ER)"-RROR":HELP:END
1660 POKE65287,8:COLOR0,2,7:COLOR4,15,6:PRINTCHR$(9)CHR$(142)"█LENDE":END
1670 FORZ=1TOAZ:IFA=2THENGOSUB400:ELSEGOSUB480
1680 IFA=3THENPRINT"█";
1690 PRINTTAB(11)"WEITER? (J/N) "
1700 POKE1373,0:GETKEYA$:IFA$="N"THENZ=0:GOTO280:ELSENEXT:Z=0:GOTO280
```

Ein Top-Zei-  
chengenerator

## Multigrafik

Das folgende Programm ist ein Top-Zeichengenerator der sowohl bei einer Diskettenstation als auch einer Datasette eingesetzt werden kann. Das Programm ermöglicht Ihnen einen eigenen Zeichensatz zu erstellen und diesen dann abzuspei-

chern, um ihn jederzeit wieder verwenden zu können.

Der Unterschied zwischen den beiden Programmen (Multigraph K + Multigraph D) besteht darin, daß in Multigraphic K (Kassette) die Zeilen

```
1355 CHAR,20,"
1360 SYS 43115 N$,1,1
1380 CHAR,21,"
1390 CHAR,22,"
1591 CHAR,20,"
1595 SYS 43115 N$,1,1
```

1661 CHAR,0,21;  
1662 CHAR,0,22;  
eingesetzt oder ergänzt worden sind.  
Bei beiden Multigraphic-Programmen kann man die abgespeicherten Zeichensätze mit: **LOAD„Filename“,8,1** bzw. **LOAD„Filename“,1,1** wieder in den Computer einladen und diese dann mit:

**POKE65298,PEEK(65298)AND251:**  
**POKE65299,PEEK(65299)AND30R56:**  
**POKE 740,56**

aktivieren. Damit der Zeichensatz nicht von einem Programm überschrieben wird, muß der hintere Teil des Speichers mit:

**POKE55,255:POKE56,55:CLR**

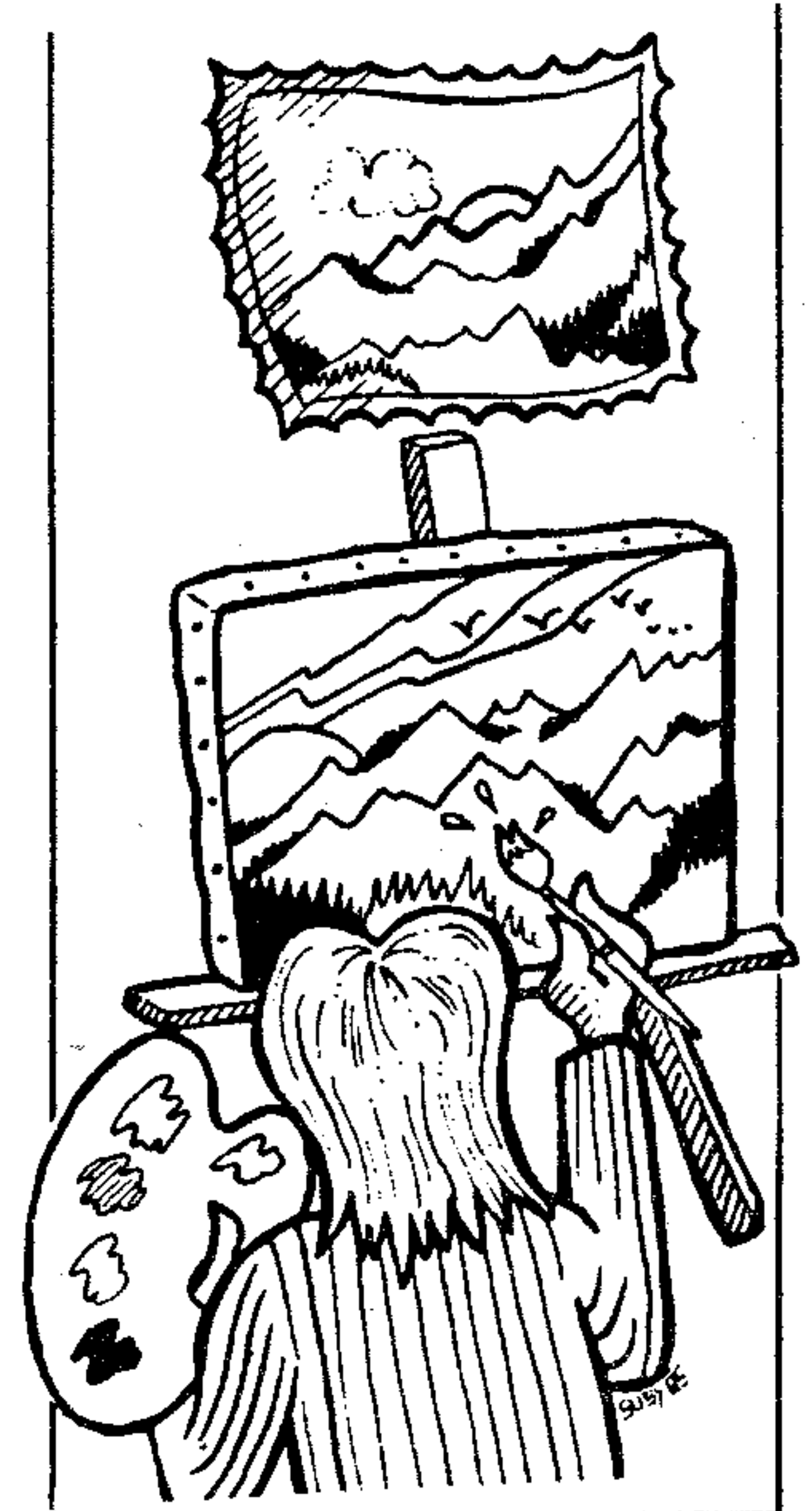
für den Zeichensatz reserviert werden. Sollte während dem Gebrauch des neuen Zeichensatzes ein schreckliches Flimmern auf dem Bildschirm auftreten, ist dieses mit:

**POKE65298,PEEK(65298)AND251**

zu beheben (das Flimmern tritt durch einen Fehler auf).

```

100 REM *****
110 REM *
120 REM * C-16 MULTIGRAPHIC *
130 REM * ----- *
140 REM *
160 REM * SOFTWARE SPRENGER *
170 REM *
180 REM *
190 REM * 29.12.1985 *
200 REM *
210 REM *
220 REM *****
240 POKE1344,64:POKE55,255:POKE56,55:CLR
260 COLOR1,1:COLOR0,1:COLOR4,1
270 REM -----
280 REM ZEICHENSATZ-COPY
290 REM -----
300 PRINT"MONITOR00000"
310 PRINT"T D000 D7FF 3800"
320 PRINT"X":PRINT"RUN380"
330 POKE1319,19:FORA=1320TO1323:POKEA,13:NEXT
340 POKE239,5:END
350 REM -----
360 REM NEUE ZEICHEN
370 REM -----
380 SCNCLR:POKE65286,PEEK(65286)AND239:COLOR1,2
390 FORA=15240TO15279:READD:POKE A,D:NEXTA
400 DATA255,0,255,0,255,0,255,0
410 DATA74,10,120,72,72,196,4,12
420 DATA233,9,24,40,47,33,32,96
430 DATA28,34,65,65,34,28,28,34
440 DATA60,66,153,161,161,153,66,60
450 FORA=14576TO14591:READD:POKEA,D:NEXT
460 DATA 0,0,0,24,24,0,0,0
470 DATA 0,60,126,126,126,126,60,0
550 REM -----
560 REM DATAEN UND VARIABLEN
570 REM -----
580 VOL 8:CO=117:L=6:R=11:VO=1
585 DIMK(7,7):TRAP2900
590 POKE65298,PEEK(65298)AND251:POKE65299,PEEK(65299)AND30R56:POKE740,56
600 KEY 8,"POKE 65298,192:SCNCLR"+CHR$(13)
610 PRINTCHR$(8):PRINTCHR$(27)+"C"
620 REM -----
630 REM TITEL
640 REM -----
670 GOSUB2290
700 CHAR,2,11,"SIND INFORMATIONEN ERWUENSCHT (J/N)"
710 POKE239,0:GETKEYA$
    
```



***Zeichnen Sie Ihre eigenen  
Gemälde mit Multigrafik***

# programme

```

720 IFA$="J"THEN GOSUB 2450
740 SOUND1,550,10:SCNCLR
745 POKE65286,PEEK(65286)AND239:COLOR4,7,6
770 GOSUB990:GOSUB 1160
775 POKE65286,PEEK(65286)OR16

831 IFPEEK(1347)=1ANDPEEK(3195)=112THENSOUND1,900,5:GOSUB1180
832 IFPEEK(1347)=2ANDPEEK(3195)=65THENSOUND1,900,5:GOSUB1211
833 IF TA$=""THEN830
834 SOUND1,900,5
835 POKEAB,PEEK(AB)-128
840 IF TA$="N"THEN GOSUB 990
850 IF TA$="L"THEN 1260
860 IF TA$="S"THEN 1450
870 IF TA$="C"THEN 1720
880 IF TA$="E"THEN 1890
890 IF TA$="P"THEN 2070

895 IF TA$="B"THEN SCNCLR:END
896 IF TA$="T"ANDVO=1THENVO=0:VOL0:GOTO970
897 IF TA$="T"ANDVO=0THENVO=1:VOL8:SOUND1,900,3
900 AB=3072+L+R*40:POKEAB,K(L-6,R-11)
910 IF TA$="□"THENR=R-1:IFR<11THENR=18
920 IF TA$="■"THENL=L+1:IFL>13THENL=6
930 IF TA$="◻"THENR=R+1:IFR>18THENR=11
940 IF TA$="◼"THENL=L-1:IFL<6THENL=13

950 IF TA$=" "AND PEEK(AB)=30THEN K(L-6,R-11)=31
960 IF TA$=" "AND PEEK(AB)=31THEN K(L-6,R-11)=30
970 AB=3072+L+R*40:POKEAB,K(L-6,R-11)+128
980 GOTO 830
990 REM -----
1000 REM RASTER ZEICHNEN
1010 REM -----
1030 FORR=11TO18
1040 FORL=6TO13
1050 K(L-6,R-11)=30

1060 NEXTL:CHAR,6,R,"^^^^^^^^":NEXTR
1070 R=11:L=6
1110 RETURN
1120 REM -----
1130 REM BILDSCHIRM ZEICHNEN
1140 REM -----

1160 CHAR,3,1,"A B C D E F G H I J K L M N O P Q R"
1170 CHAR,3,5,"S T U V W X Y Z"
1180 CHAR,3,3,"* | - - - - | | | | | | | | | | | | | | | |"
1190 CHAR,3,7,"* | / X O * | *"
1200 GOTO1220
1211 CHAR,3,3,"_ . . . + . | | = | | | | | | | | | | | | | | | |"
1212 CHAR,3,7,"_ - = . + _ - L"
1220 CHAR,25,7,"LADEN":CHAR,25,9,"SPEICHERN"
1230 CHAR,25,11,"COPIEREN":CHAR,25,13,"EDITIEREN"
1240 CHAR,25,15,"NEUES FELD":CHAR,25,17,"PAGES DRUCKEN":CHAR,25,19,"TUN EI
N/AUS"
1245 CHAR,25,21,"BEEENDEN"
1250 RETURN
1260 REM -----
1270 REM LADEN
1280 REM -----

1320 N$="":GOSUB 2230
1330 CHAR,25,9,"LADEN":CHAR,25,10,"-----"
1340 CHAR,23,12,"":INPUT"NAME ";N$
1350 IF N$=""THEN 1660
1360 SYS43115 N$,8,1

1370 POKE2034,PEEK(10):POKE2035,PEEK(43):POKE2036,PEEK(44):SYS65493:FE$=DS$
1440 K$="A":GOSUB2230:GOSUB 1220:GOTO1966
1450 REM -----
1460 REM SPEICHERN
1470 REM -----
1510 N$="":GOSUB 2230
1520 CHAR,25,9,"SPEICHERN":CHAR,25,10,"-----"
1580 CHAR,23,12,"":INPUT"NAME ";N$
1590 IF N$=""THEN1660
1595 SYS43115 N$,8,1
1600 POKE217,56:POKE216,0:POKE2036,64:POKE2035,0
1610 POKE2034,216:SYS65496
1660 POKE2022,8:POKE2023,15:PRINTCHR$(147)
1665 FE$=DS$:GOSUB2230:GOSUB1220:GOTO970

```







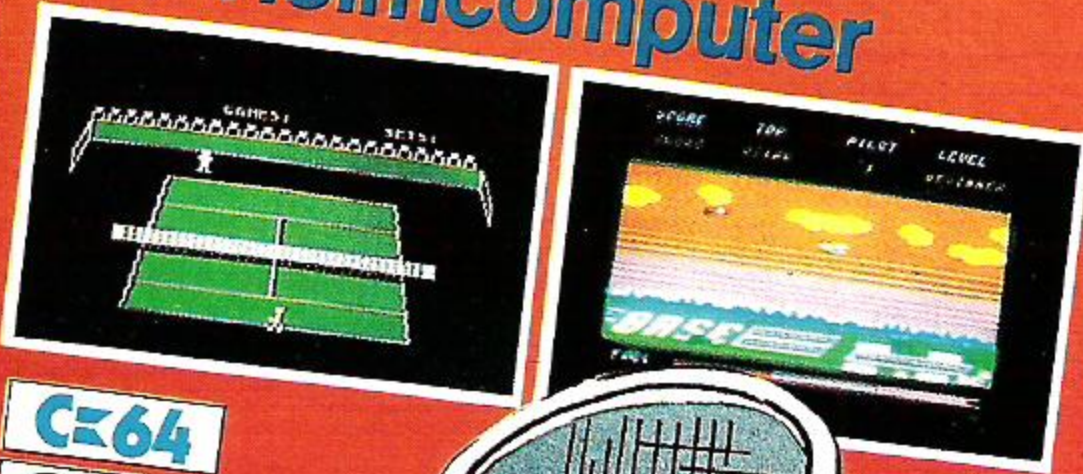
Jetzt auch regelmäßig C-16/116-Programme in

# Computronik <sup>2</sup>

Doppelausgabe März/April 86 3. Jahrgang

Software für Heimcomputer

ÖS 55 Sfr 6,50 DM 6,50



- Commodore 64
- C-16
- VIC-20
- ATARI
- Schneider
- TI-99/4A
- Apple
- Sinclair Spectrum

1986

Riesen-Chance für alle Software-Autoren

Im Heft

Checksummer

- für
- Commodore 64
- Schneider CPC 464
- MSX-Systeme
- VC-20

Hits aus dem Inhalt



Spaß mit dem Computer zum Eingeben!

# Superspiele

