

# Autorennen der Sonderklasse

**Nicht nur Beschleunigung, Höchstgeschwindigkeit und Bremswirkung des Rennwagens können per Menü eingestellt werden, es lassen sich mit Circle Circuit auch eigene Rennstrecken auf dem VC 20 entwerfen und speichern.**

Mit diesem Spiel können Sie den besten Rennfahrer in Ihrer Familie oder in Ihrem Freundeskreis ermitteln. Es geht darum, mit einem Rennwagen so schnell wie möglich eine bestimmte Anzahl von Runden in einem Rundkurs zu fahren. Das Fahrzeug wird mittels Joystick gelenkt. Drückt man auf den Feuerknopf, so beschleunigt das Auto, läßt man ihn wieder los, so wird es abgebremst. Nach jeder Runde wird die Zwischenzeit angezeigt. Nachdem jeder Spieler gefahren ist, wird die Rangliste mit den Zeiten auf dem Bildschirm ausgegeben.

Doch nun von Anfang an: Nach dem Start des Programms erscheint nach einer Weile das Titelbild mit den Wahlmöglichkeiten. Im Gegensatz zur normalen Bildschirmteilung wird bei diesem Programm der Bildschirmrahmen mitbenutzt, so daß jetzt 800 statt den bisherigen 506 Zeichen auf dem Bildschirm Platz haben. Sollte das Bild nicht genau in der Mitte stehen, kann es mit den Cursor-Tasten zentriert werden.

Zum Spielen muß man die Taste »S« drücken. Es folgt die Eingabe der Spieler- und der Rundenzahl. Der Rundkurs erscheint nun auf dem Bildschirm. Mit Betätigung des Joystickknopfs oder der Leertaste beginnt die Uhr zu laufen und man kann losfahren. Erreicht man das Ziel, so wird die Zwischenzeit angezeigt, und man ist in der nächsten Runde. Fährt man gegen eine Bande, so explodiert der Rennwagen, ist bald darauf aber wieder fahrtüchtig. Bei hoher Geschwindigkeit kann es vorkommen, daß das Auto einen Totalschaden hat. Der Spieler muß dann aufgeben, das heißt die Taste F1 drücken. Diese Taste bricht die Fahrt eines Spielers ab, egal in welcher Runde er sich befindet.

Aus der Rangliste kann man ablesen, welches die besten Spieler waren. Jeder kann sich dann seine persönliche Bestleistung merken und versuchen, sie zu verbessern, was bei einiger Übung auch oft gelingt. Wenn diese Zeit dazu noch Bestzeit ist, kommt es nicht selten vor, daß sich der bisherige Rekordhalter ein wenig ärgert und zugleich dazu angespornt wird, noch besser zu fahren. Circle Circuit ist damit eines der wenigen Computerspiele, die in Gesellschaft genauso spannend zu spielen sind wie alleine.

Doch jetzt wird es erst richtig interessant: Wählt man beim Menü die Funktion »Motorleistung«, so können die Werte für die Beschleunigung (0 bis 254), Bremsbeschleunigung (0 bis 254) und Höchstgeschwindigkeit (1 bis 254) durch Verändern der alten Werte bestimmt werden, wobei die Beschleunigung größer als die Bremsbeschleunigung sein muß. Dadurch kann man Rennwagen sehr unterschiedlicher Leistungsklassen simulieren. Vom Tourenwagen bis Formel 1 ist alles drin.

Um ganze Meisterschaftsserien fahren zu können, reicht es jedoch noch nicht aus, verschiedene Wagenklassen zur Verfügung zu haben. Es ist realistischer und macht auch

mehr Spaß, auf verschiedenen Rennstrecken zu fahren. Und auch das ist kein Problem.

Mit dem Editor kann der bestehende Rennkurs geändert oder ein vollkommen neuer Kurs erstellt werden. Auf diese Weise kann man beliebig viele Rennstrecken entwerfen und auf Massenspeicher übertragen. Die Palette reicht dabei von reinen Hochgeschwindigkeitskursen bis zu technisch anspruchsvolleren Strecken.

## Der Streckeneditor

Nach Betätigen der Taste »E« des Menüs erscheint der Kurs auf dem Bildschirm. Der Cursor kann nun mit dem Joystick bewegt werden. Wenn man den Feuerknopf betätigt, erscheint an der Cursorposition ein vorher gewähltes Zeichen.

Die Zeichenwahl trifft man durch Druck auf eine entsprechende Taste. Es sind fast alle nichtinversen Zeichen in allen Farben erlaubt. Folgende Tasten sind undefiniert worden:

T	inverses Blank
F	Ecke links oben
G	Ecke rechts oben
H	Ecke links unten
V	Ecke rechts unten
W	Abrundung links
X	Abrundung rechts
Y	Abrundung oben
Z	Abrundung unten

Die CLR-Taste löscht den Bildschirm. Mit den Tasten 2 bis 8 wählt man die laufende Farbe.

Die Funktionstasten F3 bis F7 dienen zur Positionierung des Rennwagens. Dabei bedeutet F3 in Richtung nach oben, F5 in Richtung nach links und F7 in Richtung nach rechts. Das Nummernzeichen »#« stellt die Ziellinie dar. Mit der Taste F1 gelangt man ins Menü zurück.

Der sich momentan im Speicher befindliche Rennkurs kann auf Diskette oder Kassette gespeichert werden. Dabei werden Floppy-Fehler abgefangen. Wenn man für den Filenamen nur einen Stern (»\*«) eingibt, so springt man zum Menü zurück. Dies kann von Nutzen sein, wenn man sich vertippt haben sollte. Wenn man das Programm mit STOP oder STOP/RESTORE abgebrochen hat, kann man mit »GOTO 400« einen Warmstart durchführen.

## Das Eintippen

Um das Programm einzutippen, geht man so vor:

- (1) POKE44,32:POKE7936,0:NEW (diese Befehle verschieben den Basic-Anfang nach oben).
- (2) Eintippen und Abspeichern des Listings 1. Wenn man den Basic-Teil fehlerlos abtippt und die -1 am Ende eines DATA-Blocks nicht vergißt, dürfte wohl jeder Fehler vom Programm erkannt werden.
- (3) Löschen des Listings 1 mit NEW und Abtippen des Listings 2 (der Basic-Anfang muß immer noch verschoben sein!).

Wenn Sie die beiden Teile abgespeichert haben, können Sie das Programm nach gleichem Schema laden (vor dem Laden muß ebenfalls der Basic-Anfang verschoben werden).

Man kann das Programm auch als ein Ganzes abspeichern. Dazu sind folgende Schritte zu unternehmen:

- Computer einschalten
- POKE44,31:POKE7936,0:NEW
- Listing 1 laden und starten
- Listing 2 laden
- ?PEEK(45);PEEK(46) eingeben und sich die Werte aufschreiben
- POKE44,18:POKE4608,0:NEW
- Folgende Basic-Zeile eingeben: 1 POKE44,31:RUN

- POKE45,(1.Wert):POKE46,(2.Wert)
- SAVE"CIRCLE CIRCUIT";1 (,8 bei Floppy)

Auf diese Weise kann das Programm geladen und gestartet werden, ohne vorher etwas eingeben zu müssen. Und natürlich sind Sie auch jetzt noch völlig frei, sich eigene Rennstrecken zu erzeugen und abzuspeichern. Vom Hinderniskurs bis zum Nürburgring ist alles möglich. Also auf zur Formel I-Weltmeisterschaft im Wohnzimmer!

(Hannes Kaltenbach/ev)

\$1000	4096	Bildschirmspeicher
\$1320	4896	Maschinensprache
\$1400	5120	Zeichensatz
\$1C00	7168	Maschinensprache
\$1F00	7936	Basic-Beginn
\$3FFF	16383	Ende RAM-Speicher

F	zur Farbadressenberechnung
BE	Bildschirmadresse
AF	Autofarbe
R	Richtung des Autos
XA,YA	X,Y-Wert des Autos
SZ	Spiezlerzahl
RZ	Rundenzahl
J	Joystickadresse
T	Tonadresse
AC	Beschleunigung
AN	Bremsbeschleunigung
HG	Höchstgeschwindigkeit
HZ,VZ	zur Bildschirmzentrierung
T(9)	Zeiten der Spieler
Z(9)	sortierte Zeiten
F\$(6)	Farben
A\$(28)	enthält den Kurs

Speicheraufteilung bei »Circle Circuit«

```

1 REM * CIRCLE CIRCUIT * <125>
2 REM <064>
3 REM VON HANNES KALTENBACH <069>
4 REM PRIELMAYERSTR.16 <028>
5 REM 7990 FRIEDRICHSHAFEN <121>
6 REM <068>
7 REM MASCHINENSPRACHE UND SONDERZEICHEN <065>
8 REM <070>
9 REM VOR DEM EINTIPPEN: <203>
10 REM POKE44,31:POKE7936,0:NEW <014>
11 REM <073>
12 REM <074>
100 REM BLOCK 1 <007>
110 BL=1:Z=212:PS=21320:GOSUB 800 <116>
120 REM BLOCK 2 <028>
130 BL=2:Z=128:PS=11737:GOSUB 800 <097>
140 REM BLOCK 3 <049>
150 BL=3:Z=350:PS=35875:GOSUB 800 <115>
160 REM BLOCK 4 <070>
170 BL=4:Z=353:PS=36706:GOSUB 800 <149>
200 REM MASCHINENSPRACHE <169>
210 RESTORE:FOR I=4896 TO 5107:READ X:POKE <223>
I,X:NEXT:READ X <062>
220 REM SONDERZEICHEN
230 A=6144:SYS 4943:REM ZEICHENSATZ KOPIER <124>
EN <165>
240 FOR I=5280 TO 5287:POKE I,255:NEXT <122>
250 FOR I=5168 TO 5191:READ X:POKE I,X:NEX <117>
T
260 FOR I=5296 TO 5335:READ X:POKE I,X:NEX <049>
T <251>
270 FOR N=1 TO 8:FOR I=0 TO 7:READ X:POKE <246>
A+I,X:NEXT:A=A+40:NEXT:READ X <108>
280 REM MASCHINENSPRACHE <058>
290 FOR I=7168 TO 7517:READ X:POKE I,X:NEX <010>
T:READ X
300 FOR I=7518 TO 7870:READ X:POKE I,X:NEX <249>
T <037>
310 END <203>
600 IF N=Z THEN 640 <088>
610 PRINT"FALSCHER ANZAHL VON ZAHLEN IN BLO <040>
CK"BL <191>
620 PRINT"SOLL:"Z <253>
630 PRINT"IST ":"N:END
640 IF S=PS THEN PRINT"BLOCK"BL"OK":RETURN <227>
650 PRINT"PRUEFSUMMENFEHLER IN BLOCK"BL <165>
660 PRINT"SOLL:"PS <212>
670 PRINT"IST ":"S:END <234>
800 REM PRUEFSUMME UND ANZAHL VON BLOCK BE <128>
STIMMEN
810 N=0:S=0
820 READ X:IF X=-1 THEN GOSUB 600:RETURN <035>
830 N=N+1:S=S+X:GOTO 820
999 REM **** BLOCK 1 ****
1000 DATA 169,31,133,1,169,19,133,2,169,32 <011>
,160,0,145,1,170,165,2,72,24,105,132
1010 DATA 133,2,169,1,145,1,104,133,2,138, <212>
198,1,166,1,232,208,230,198,2,166,2
1020 DATA 224,15,208,222,96,169,20,133,2,1 <116>
60,0,132,1,165,2,72,24,105,108,133
1030 DATA 2,177,1,170,104,133,2,138,145,1, <212>
200,208,236,230,2,165,2,201,28,208
1040 DATA 228,96,160,0,132,2,162,3,10,38,2 <111>

```

Wichtige Variable

```

,202,208,250,133,1,165,2,24,105,20 <093>
1050 DATA 133,2,96,169,0,133,42,133,63,133 <046>
,64,162,8,70,41,144,13,165,63,24,101
1060 DATA 40,133,63,165,64,101,42,133,64,6 <110>
,40,38,42,202,208,232,96,177,251,170
1070 DATA 56,237,68,3,240,4,201,5,144,4,13 <142>
8,201,32,96,170,189,61,3,201,32,96
1080 DATA 208,8,169,0,141,114,3,76,235,29, <132>
169,35,205,62,3,240,16,205,63,3,240
1090 DATA 11,205,64,3,240,6,205,65,3,240,1 <174>
,96,173,114,3,240,3,76,235,29,169,1
1100 DATA 141,114,3,96,-1 <244>
1999 REM **** BLOCK 2 **** <128>
2000 DATA 254,252,252,248,240,224,128,0,12 <242>
7,63,63,31,15,3,1,0
2010 DATA 0,128,224,240,248,252,252,254,0, <218>
1,3,15,31,63,63,127
2020 DATA 3,7,15,15,15,15,7,3,192,224,240, <211>
240,240,240,224,192
2030 DATA 0,0,24,126,126,255,255,255,255,2 <171>
55,255,126,126,24,0,0
2040 DATA 8,62,20,20,93,93,73,0,16,96,82,2 <063>
01,60,28,76,32
2050 DATA 14,68,124,206,124,68,14,0 <173>
2060 DATA 16,38,14,30,172,73,114,16 <191>
2070 DATA 0,73,93,93,20,20,62,8,4,50,56,60 <015>
,147,74,6,8
2080 DATA 112,34,62,115,62,34,112,0 <032>
2090 DATA 8,78,146,53,120,112,100,8,-1 <126>
2999 REM **** BLOCK 3 **** <128>
3000 DATA 138,41,7,141,66,3,152,41,7,141,6 <088>
7,3,138,74,74,74,170,152,74,74,74,168
3010 DATA 134,251,169,25,133,40,132,41,32, <003>
138,19,165,64,24,105,16,133,252,165
3020 DATA 251,24,101,63,133,251,144,2,230, <202>
252,96,32,116,19,160,0,177,1,145,82
3030 DATA 200,192,8,208,247,96,160,0,140,6 <089>
0,3,162,2,134,40,169,3,133,83,169,69
3040 DATA 133,82,32,173,19,234,24,208,1,56 <237>
,46,60,3,132,42,32,52,28,164,42,165
3050 DATA 82,24,105,8,133,82,198,40,208,2, <047>
160,24,200,192,27,208,220,96,162,0
3060 DATA 134,41,134,42,174,66,3,240,6,74, <148>
102,42,202,208,250,133,41,96,32,116
3070 DATA 19,160,0,140,61,3,132,40,177,1,3 <223>
2,121,28,172,67,3,165,41,240,27,25
3080 DATA 69,3,133,63,185,69,3,24,101,41,1 <053>
76,4,197,63,240,5,169,1,141,61,3,165
3090 DATA 63,153,69,3,165,42,240,27,25,77, <110>
3,133,63,185,77,3,24,101,42,176,4,197
3100 DATA 63,240,5,169,1,141,61,3,165,63,1 <099>
53,77,3,230,40,164,40,192,8,240,18
3110 DATA 238,67,3,173,67,3,201,8,208,168, <051>
24,105,8,141,67,3,144,160,96,177,251
3120 DATA 170,56,229,242,240,4,201,5,144,4 <157>
,138,201,32,96,170,189,61,3,201,32
3130 DATA 96,160,0,173,62,3,145,189,200,17 <019>
3,63,3,145,189,160,25,173,64,3,145
3140 DATA 189,200,173,65,3,145,189,96,160, <243>
0,177,251,141,62,3,200,177,251,141

```

Listing 1. Maschinenspracheteil und Sonderzeichen zu »Circle Circuit«

```

3150 DATA 63,3,160,25,177,251,141,64,3,200
,177,251,141,65,3,174,68,3,232,138 <175>
3160 DATA 32,116,19,160,0,185,69,3,145,1,2
00,192,32,208,246,173,68,3,24,105,4,-
1 <025>
3169 REM **** BLOCK 4 **** <060>
3170 DATA 160,26,145,251,170,78,60,3,144,1
6,165,252,72,24,105,132,133,252,173 <131>
3180 DATA 101,3,145,251,104,133,252,136,19
2,24,208,2,160,1,202,138,205,68,3,208 <231>
3190 DATA 218,165,251,133,189,165,252,133,
190,173,68,3,141,102,3,96,32,0,28,32 <065>
3200 DATA 67,28,173,68,3,32,141,28,96,32,1
5,29,32,42,29,96,174,105,3,172,106 <142>
3210 DATA 3,173,104,3,208,1,136,201,1,208,
2,136,202,201,2,208,1,202,201,3,208 <132>
3220 DATA 2,202,200,201,4,208,1,200,201,5,
208,2,232,200,201,6,208,1,232,201,7 <222>
3230 DATA 208,2,232,136,142,105,3,140,106,
3,32,149,29,173,61,3,76,197,19,169 <246>
3240 DATA 160,141,11,144,32,162,29,173,104
,3,133,40,169,5,133,41,32,138,19,165 <223>

```

```

3250 DATA 63,9,128,141,68,3,32,162,29,174,
105,3,142,107,3,172,106,3,140,108,3 <213>
3260 DATA 76,145,30,169,127,141,34,145,173
,31,145,41,28,174,104,3,201,8,208,2 <123>
3270 DATA 162,1,201,12,208,2,162,2,201,4,2
08,2,162,3,72,173,32,145,41,128,168 <019>
3280 DATA 104,201,20,208,8,162,4,192,0,208
,2,162,5,201,28,208,6,192,0,208,2,162 <247>
3290 DATA 6,201,24,208,8,162,0,192,0,208,2
,162,7,142,104,3,173,31,145,41,32,208 <236>
3300 DATA 19,206,113,3,208,14,169,10,234,1
41,113,3,206,111,3,208,3,238,111,3 <187>
3310 DATA 169,255,141,34,145,206,112,3,208
,14,173,111,3,141,112,3,76,169,29,169 <117>
3320 DATA 0,141,11,144,206,115,3,208,13,16
9,40,141,115,3,238,111,3,208,3,206 <200>
3330 DATA 111,3,165,197,201,39,208,4,141,6
1,3,96 <197>
3340 DATA 173,111,3,201,25,208,3,238,111,3
,76,24,30,-1 <074>

```

Listing 1. Maschinenspracheteil und Sonderzeichen zu »Circle Circuit« (Schluß)

```

1 REM * CIRCLE CIRCUIT * <125>
2 REM <064>
3 REM VON HANNES KALTENBACH <069>
4 REM PRIELMAYERSTR.16 <028>
5 REM 7990 FRIEDRICHSHAFEN <121>
6 REM <068>
7 REM HAUPTPROGRAMM <052>
8 REM <070>
9 REM VOR DEM EINTIPPEN: <203>
10 REM POKE44,31:POKE7936,0:NEW <014>
11 REM <073>
15 GOTO 100 <199>
20 FOR I=1 TO 1500:NEXT:RETURN <174>
100 PRINT"BITTE WARTEN !" <035>
199 REM VARIABLE <246>
200 DIM T(9),Z(9):F=33792:BE=4147:J=37151:
T=36876:HZ=9:VZ=21 <218>
210 AF=0:R=0:XA=0:YA=0:N=0:AC=238:AN=220:H
G=230 <203>
220 X=0:Y=0:0=0:FA=0:A=0:BC=0:BO=0:BN=0 <057>
300 A$="" :DIM A$(28):GOSUB 5000 <013>
320 DIM F$(6):FOR I=0 TO 6:READ A:F$(I)=CH
R$(A):NEXT <226>
330 DATA 5,28,159,156,30,31,158 <253>
400 POKE T+3,10:I=36864:POKE I,HZ:POKE I+1
,VZ:POKE I+2,25:POKE I+3,64 <005>
410 POKE I+14,15:PRINT CHR$(142)CHR$(8) <249>
499 REM TITELBILD <145>
500 SYS 4896:POKE T+3,10:POKE T-7,192:GOSU
B 4500 <196>
510 GOSUB 4600:PRINT SPC(27)" {WHITE}TTTTTT
TTTTTTTTTTTTTTTT"SPC(4)"@* CIRCLE(2SPAC
E)CIRCUIT *"; <020>
515 PRINT SPC(4)"@*SPC(19)"@*SPC(4)"@ (2SPA
CE)---9/84---BY---(2SPACE)@*SPC(4)"@*S
PC(19)"@*"; <119>
520 PRINT SPC(4)"@ (3SPACE)H. KALTENBACH(3S
PACE)@*SPC(4)"L000000000000000000000000"; <053>
540 PRINT SPC(29)" {RVSON}S {RVOFF}PIEL {10SP
ACE,RVSON}E {RVOFF}DITOR"; <015>
545 PRINT SPC(28)" {SPACE,RVSON}A {RVOFF}BSP
EICHERN {4SPACE,RVSON}L {RVOFF}ADEN"; <062>
550 PRINT SPC(30)" {RVSON}M {RVOFF}OTORLEIST
UNG";:POKE 198,0 <202>
600 GET A$ <046>
610 IF A$="S"THEN 700 <056>
615 IF A$="E"THEN 800 <058>
620 IF A$="A"THEN 3300 <071>
625 IF A$="L"THEN 3500 <226>
630 IF A$="M"THEN 4200 <083>
640 IF A$=CHR$(29)THEN HZ=HZ+1:IF HZ>17 TH
EN HZ=1 <182>
650 IF A$=CHR$(17)THEN VZ=VZ+1:IF VZ>70 TH
EN VZ=0 <140>
655 POKE T-12,HZ:POKE T-11,VZ <148>
690 GOTO 600 <150>
700 PRINT SPC(37)"SPIELERZAHL (1-9):":POK
E 204,0 <199>
705 GET A$:SZ=VAL(A$):IF SZ<1 OR SZ>9 THEN
705 <249>
710 PRINT A$SPC(31)"RUNDENZAHL (1-98):":G
OSUB 4000 <048>
720 RZ=INT(VAL(B$)):IF RZ<1 THEN RZ=1 <155>

```

```

730 IF RZ>98 THEN RZ=98 <198>
780 SP=1:A=1:0=0 <007>
799 REM KURS ZEICHNEN <162>
800 SYS 4896:POKE T+3,8:POKE T-7,205 <053>
805 PRINT" {HOME,WHITE}SP 1 RUNDE 01{3SPACE
}00:00.00" <162>
810 BN=9016:FOR I=4121 TO 4145:POKE I+F,2:
POKE I,20 <033>
815 POKE BN-I+F,2:POKE BN-I,20:NEXT <226>
820 FOR I=4146 TO 4846 STEP 25:POKE I+F,2:
POKE I,20 <060>
825 POKE BN-I+F,2:POKE BN-I,20:NEXT <236>
850 PRINT" {HOME}"SPC(49):FOR I=0 TO 17:PR
INT SPC(2)A$(I):NEXT <207>
860 POKE 648,17:PRINT" {HOME}"SPC(245)LEFT$
(A$(18),22); <252>
870 POKE 648,18:PRINT" {HOME}"MID$(A$(18),2
3); <181>
880 FOR I=19 TO 28:PRINT SPC(2)A$(I):NEXT
:POKE 648,16 <073>
890 BN=BE+XA+25*YA:IF A$="E"THEN 2000 <158>
899 REM INITIALIZE <251>
900 X=(XA+1)*8:Y=(YA+2)*8 <129>
905 POKE 190,BN/256:POKE 189,BN-256*PEEK(1
90) <028>
910 POKE 830,PEEK(BN):POKE 831,PEEK(BN+1) <239>
915 POKE 832,PEEK(BN+25):POKE 833,PEEK(BN+
26) <102>
930 POKE 873,X:POKE 874,Y:POKE 882,0 <084>
940 POKE 836,R*5+128:POKE 869,AF:POKE 872,
R:POKE 879,255 <007>
950 POKE 7793,255-AC:POKE 7836,255-AN:POKE
7862,255-HG <059>
980 POKE 781,X:POKE 782,Y:SYS 7573:SYS 758
6 <196>
990 N=14:GOTO 1430 <131>
995 TI$="000000" <212>
1000 SYS 7704 <167>
1010 IF PEEK(829)=39 THEN T(SP)=0:GOTO 131
0 <254>
1020 IF PEEK(882)THEN 1250 <028>
1050 IF PEEK(879)>200 THEN 1140 <156>
1080 X=PEEK(873):Y=PEEK(874):POKE T+1,134 <068>
1100 FOR N=1 TO 255-PEEK(879)STEP 15 <157>
1110 POKE 781,X:POKE 782,Y:POKE 836,INT(RN
D(1)*8)*5+128 <238>
1120 SYS 7573:SYS 7586:NEXT <168>
1140 POKE T+1,0:POKE 879,255:POKE 873,PEEK
(875):POKE 874,PEEK(876) <178>
1160 GOTO 1000 <084>
1250 C$=RIGHT$(STR$(INT(TI/6*10)),2):B$=TI
$ <228>
1255 A=A+1:A$=MID$(STR$(A),2):IF LEN(A$)<2
THEN A$="0"+A$ <087>
1260 PRINT" {HOME}"SPC(11)A$SPC(3)MID$(B$,3
,2)SPC(1)MID$(B$,5)SPC(1)C$ <086>
1280 IF A=RZ+1 THEN 1300 <136>
1290 SYS 7664:GOTO 1010 <209>
1300 T(SP)=VAL(B$)*100+VAL(C$):GOTO 1400 <151>
1310 SP=SP+1:IF SP>SZ THEN 1500 <255>
1320 PRINT" {HOME}"SPC(11)"01{3SPACE}00:00.

```

Listing 2. »Circle Circuit«, Hauptprogramm. Verwenden Sie bei der Eingabe den Checksummer VC 20 in diesem Heft.

```

00" <061>
1330 O=0:A=1 <050>
1350 SYS 7439:GOTO 900 <211>
1400 FOR N=1 TO 8:FOR I=200 TO 240:POKE T,
I <235>
1410 NEXT I,N:POKE T,0:GOSUB 20:GOTO 1310 <139>
1430 IF (PEEK(J)AND 32)=0 OR PEEK(197)=32 T
HEN GOSUB 1470:GOTO 995 <175>
1440 N=N+1:IF N=15 THEN GOSUB 1460 <054>
1445 IF N=30 THEN N=0:GOSUB 1470 <217>
1450 GOTO 1430 <024>
1460 PRINT "{HOME,4SPACE}":RETURN <127>
1470 PRINT "{HOME,WHITE}SP"SP:RETURN <206>
1500 SYS 4896:POKE T-7,192:FOR I=6400 TO 6
407:POKE I,255:NEXT:GOSUB 4500:GOSUB
4600 <049>
1510 PRINT "{2SPACE}RANGLISTE"SPC(16)" {RED}
TTTTTTTTTTT{WHITE}"SPC(11); <146>
1515 O=230:IF SZ=1 THEN SP=1:Z(1)=T(1):FOR
N=1 TO 1:GOTO 1600 <214>
1520 FOR I=1 TO SZ:Z(I)=T(I):NEXT <203>
1530 FOR I=1 TO SZ-1:FOR N=I+1 TO SZ:IF Z(
I)<Z(N)THEN 1550 <118>
1540 A=Z(I):Z(I)=Z(N):Z(N)=A <045>
1550 NEXT N,I <129>
1570 FOR N=1 TO SZ:FOR I=1 TO SZ:IF T(I)=Z
(N)THEN SP=I <188>
1580 NEXT:T(SP)=0:IF Z(N)=0 THEN 1640 <053>
1600 POKE T,0:POKE T-1,0+15:O=O-10:A#=MID$(
STR$(Z(N)),2) <113>
1610 IF LEN(A#)<6 THEN A#="0"+A#:GOTO 1610 <081>
1620 PRINT SPC(30)"SPIELER"SP "{2SPACE}"LEF
T$(A#,2)"":MID$(A#,3,2)".MID$(A#,5); <066>
1630 FOR D=15 TO 0 STEP-.5:POKE T+2,D:NEXT <160>
1640 POKE T,0:POKE T-1,0:POKE T+2,15:NEXT
N <102>
1650 PRINT "{HOME}"SPC(15)"(TASTE)":POKE 19
8,0:WAIT 198,1:POKE T+3,10:GOTO 510 <169>
1999 REM EDITOR <192>
2000 POKE BN,R*5+128:POKE BN+F,AF <073>
2010 X=0:Y=0:O=PEEK(BE):FA=5:BN=BE:BO=BE:B
C=20:C#="T" <250>
2020 POKE BN,102:POKE 198,0 <121>
2100 FOR I=1 TO 40:NEXT <214>
2105 I=PEEK(J):IF (I AND .16)=0 THEN IF X>0
THEN X=X-1 <010>
2110 POKE J+3,127:IF (PEEK(J+1)AND 128)=0 T
HEN IF X<22 THEN X=X+1 <037>
2120 POKE J+3,255:IF (I AND 4)=0 THEN IF Y>
0 THEN Y=Y-1 <142>
2130 IF (I AND 8)=0 THEN IF Y<28 THEN Y=Y+1 <120>
2140 BN=BE+X+25*Y:IF BN=BO THEN 2160 <194>
2150 POKE BO,0:O=PEEK(BN):POKE BN,102:BO=B
N <186>
2160 IF (I AND 32)THEN 2500 <079>
2400 O=BC:POKE BN+F,FA <189>
2450 A$(Y)=LEFT$(A$(Y),X*2)+F$(FA-1)+C#+MI
D$(A$(Y),2*X+3) <246>
2500 GET A$:IF A#=""THEN 2100 <231>
2510 A=ASC(A#) <118>
2520 IF A>49 THEN IF A<57 THEN FA=A-49:GOT
O 2500 <045>
2530 IF A=133 THEN 500 <052>
2540 IF A=147 THEN 2700 <154>
2550 IF A=134 THEN R=0:O=128:AF=FA:XA=X:YA
=Y:POKE BN+F,FA:GOTO 2500 <101>
2560 IF A=135 THEN R=2:O=138:AF=FA:XA=X:YA
=Y:POKE BN+F,FA:GOTO 2500 <002>
2570 IF A=136 THEN R=6:O=158:AF=FA:XA=X:YA
=Y:POKE BN+F,FA:GOTO 2500 <175>
2580 IF A<32 THEN 500 <061>
2585 IF A>127 THEN IF A<161 THEN 2500 <002>
2590 IF A=44 OR A=58 OR A=34 THEN 2500 <048>
2600 C#=A#:BC=A AND 63:IF A AND 128 THEN B
C=BC OR 64 <089>
2650 POKE BN,BC:GOTO 2100 <031>
2700 A#="" :FOR I=1 TO 23:A#=A#+ "{WHITE,SPA
CE}":NEXT:FOR I=0 TO 28:A$(I)=A#:NEXT <211>
2710 A#="E":GOTO 800 <082>
2999 REM CASS,DISK <010>
3000 PRINT SPC(37)"CASS,DISK (C,D) ? " :PO
KE 204,0 <106>
3005 GET A$:IF A#<"C"AND A#<"D"THEN 3005 <163>
3010 PRINT A# :RETURN <198>
3100 PRINT SPC(31)"FILENAME: " :GOSUB 4000:
RETURN <082>
3300 GOSUB 3000:GOSUB 3100:IF B#=""THEN 5

```

```

10 <015>
3310 IF A#="C"THEN PRINT "{CLR}":OPEN 2,1,
1,B#:GOTO 3400 <075>
3320 OPEN 1,8,15:OPEN 2,8,2,B#+",S,W":INPU
T#1,I,A#:IF I THEN 3700 <065>
3400 FOR I=0 TO 28:PRINT#2,A$(I):NEXT <098>
3410 C#=CHR$(13):PRINT#2,AF;C#;R;C#;XA;C#;
YA <127>
3450 CLOSE 2:CLOSE 1:GOTO 510 <255>
3500 GOSUB 3000:GOSUB 3100:IF B#=""THEN 5
10 <217>
3510 IF A#="C"THEN PRINT "{CLR}":OPEN 2,1,
0,B#:PRINT "{3DOWN}GEFUNDEN !":GOTO 36
00 <227>
3520 OPEN 1,8,15:OPEN 2,8,2,B#+",S,R":INPU
T#1,I,A#:IF I THEN 3700 <006>
3600 FOR I=0 TO 28:INPUT#2,A$(I):NEXT <220>
3610 INPUT#2,AF,R,XA,YA <188>
3620 CLOSE 1:CLOSE 2:GOTO 510 <227>
3700 PRINT "{HOME}"SPC(250)SPC(150)"FEHLER"
I;A# <000>
3710 CLOSE 1:CLOSE 2:GOSUB 20:GOSUB 20:GOS
UB 20:GOTO 510 <242>
4000 B#="" <223>
4005 POKE 204,0 <094>
4010 GET C#:IF C#=""THEN 4010 <106>
4020 IF C#=CHR$(13)THEN POKE 204,1:RETURN <254>
4030 IF C#=CHR$(20)THEN IF B#<">THEN B#=L
EFT$(B#,LEN(B#)-1):GOTO 4070 <037>
4040 IF LEN(B#)=16 THEN 4010 <111>
4050 IF ASC(C#)<32 OR ASC(C#)>95 THEN 4010 <129>
4060 B#=B#+C# <128>
4070 PRINT C# :GOTO 4010 <088>
4100 IF LEN(B#)>4 THEN A=-1:GOTO 4120 <234>
4110 A=INT(VAL(B#)):IF A>255 THEN A=255 <210>
4120 RETURN <114>
4199 REM MOTORLEISTUNG <099>
4200 B#=STR$(AC):PRINT SPC(37)"BESCHLEUNIG
UNG: "B#; <075>
4210 GOSUB 4005:GOSUB 4100:IF A<0 THEN 510
<085>
4220 AC=A:IF AC=255 THEN AC=254 <097>
4300 PRINT SPC(10-LEN(B#))"BREMSBESCHL. {2S
PACE}":B#=STR$(AN):PRINT B#; <106>
4310 GOSUB 4005:GOSUB 4100:IF A<0 THEN 510 <185>
4320 AN=A:IF AN=255 THEN AN=254 <062>
4350 PRINT SPC(10-LEN(B#))"HOECHSTGESCHW. :
";B#=STR$(HG):PRINT B#; <125>
4360 GOSUB 4005:GOSUB 4100:IF A<0 THEN 510 <237>
4370 HG=A:IF HG=0 THEN HG=1 <052>
4380 GOTO 510 <038>
4499 REM 3D-STRASSE <130>
4500 BN=4607:FOR I=1 TO 12:POKE BN,78:POKE
BN+2*I,77:BN=BN+24:NEXT <159>
4510 BN=4608:FOR I=2 TO 5:POKE BN,46:BN=BN
+25*I:NEXT:RETURN <094>
4600 PRINT "{CLR}":BN=4246:FOR I=12 TO 1 S
TEP-1:POKE BN,77:POKE BN+2*I,78 <138>
4620 BN=BN+26:NEXT:POKE 4558,160:POKE 4583
,160:RETURN <119>
5000 A#="{10SPACE}" <203>
5100 A$(0)="{RED}F{7SPACE,WHITE}#"+A#+"{3S
PACE,RED}G" <087>
5110 A$(1)="{8SPACE,WHITE}#{4SPACE}"+A# <049>
5120 A$(2)=A$(1) <253>
5130 A$(3)=A$(1) <039>
5140 A$(4)="{4SPACE,CYAN}WTTTTTTTTTTTTTTT{4
SPACE}" <004>
5150 A$(5)=A#+ "GTTTTT{2SPACE}GH{3SPACE}" <025>
5160 A$(6)=A#+ "{9SPACE}T{3SPACE}" <046>
5170 A$(7)="{2SPACE,GREEN}VH"+A#+"{5SPACE,
CYAN}T{3SPACE}" <141>
5180 A$(8)="{2SPACE,GREEN}GF"+A#+"{5SPACE,
CYAN}T{3SPACE}" <108>
5190 A$(9)=A#+ "{9SPACE}T{3SPACE}" <173>
5200 A$(10)=A$(9) <163>
5210 A$(11)=A$(9) <237>
5220 A$(12)="{4SPACE,RED}VTTTTTTTTTX{5SPACE
,CYAN}T{3SPACE}" <045>
5230 A$(13)="{RED}H{2SPACE}VF"+A#+"{4SPACE
,CYAN}T{3SPACE}" <197>
5240 A$(14)="{RED}TTTT"+A#+"{5SPACE,CYAN}T
{3SPACE}" <212>
5250 A$(15)="{RED}F"+A#+"{7SPACE,CYAN}VT{3
SPACE}" <113>

```

Listing 2. »Circle Circuit« (Fortsetzung)

```

5260 A$(16)=A$+" {7SPACE}VTF {3SPACE}" <002>
5270 A$(17)=A$+" {6SPACE}VTF {4SPACE}" <076>
5280 A$(18)=" {7SPACE}VTTTTTTTTT {5SPACE}" <206>
5290 A$(19)=" {6SPACE}VF {5SPACE}" +A$ <220>
5300 A$(20)=" {5SPACE}VF" +A$+" {GREEN}V {5SPACE}" <254>
5310 A$(21)=" {4SPACE, CYAN}VF" +A$+" {SPACE, GREEN}GH {4SPACE}" <194>
5320 A$(22)=" {4SPACE, CYAN}Z {5SPACE, RED}H {2SPACE}" +A$ <116>
5330 A$(23)=A$+"G {2SPACE}" +A$ <209>
5340 A$(24)=A$+" {3SPACE, GREEN}V {9SPACE}" <055>
5350 A$(25)=A$+" {2SPACE}VF {5SPACE, RED}V {2SPACE}W" <013>
5360 A$(26)=" {7SPACE, GREEN}GH {9SPACE, RED}V

```

```

F {3SPACE}" <176>
5370 A$(27)=" {8SPACE, GREEN}F" +A$+" {3SPACE, RED}V" <028>
5380 A$(28)="H" +A$+" {9SPACE}VTT" <249>
5390 AF=1:R=2:XA=6:YA=1 <121>
5399 REM KURS FORMATIEREN <106>
5400 FOR N=0 TO 28:A$="" <246>
5410 FOR I=1 TO LEN(A$(N)):B$=MID$(A$(N), I, 1) <031>
5420 A=ASC(B$): IF A<32 OR A>143 AND A<160 THEN C$=B$:GOTO 5450 <050>
5430 A$=A$+C$+B$ <192>
5450 NEXT I:A$(N)=A$:NEXT N:RETURN <203>

```

Listing 2. »Circle Circuit«, Hauptprogramm (Schluß)

# Gold-Fieber

**Sammeln Sie Goldklumpen! Die lukrative Tätigkeit wird allerdings durch viele Gefahren sehr erschwert.**

»Stone Rescue«, so der Name des Spiels, handelt von dem nie aufgebenden Goldsucher Freddy Stone. Freddy durchwühlt unterirdische Gänge und Räume nach Goldklumpen. Er ist dermaßen goldgierig, daß er einen Raum erst dann wieder verläßt, wenn er den letzten Rest des gelben Metalls »eingesackt« hat. Selbst Gefahren wie bissige Giftschlangen, Pilze mit tödlicher Wirkung, magische Fallen und spitze Dornen, dämpfen Freddy's Goldgier nicht im mindesten.

Hat Freddy alle Nuggets eingesteckt, krabbelt er tiefer in die Höhle. Er steigt dazu über eine Leiter in eine tiefer liegende Etage. Innerhalb eines Höhlenraums kann sich Freddy auch an Lianen nach oben oder unten hangeln.

Tödlich sind die Höhlenwände, wenn Freddy dagegen springt. Ebenso verträgt Freddy keinen Sturz über vier Meter Höhe. Wie soll er auch! Sollte Freddy einen Raum nicht mehr lebend verlassen, werden alle Goldklumpen, wie durch eine magische Hand, wieder am ursprünglichen Ort plaziert. In diesem Fall vertreten Freddy seine beiden Zwillingbrüder, die auch vom Goldfieber gepackt werden.

## Helfen Sie dem Goldsucher Freddy!

Führen Sie Freddy durch das unterirdische Reich und zeigen Sie ihm, wo die Goldklumpen liegen. Schließen Sie dazu einen Joystick an den Control-Port 1 an. Haben Sie sich durch die zehn Bilder »geschlagen«, können Sie Ihre eigenen Höhlen konstruieren. Mit dem Spielgenerator von »Stone Rescue« sind bis zu 89 Bilder möglich.

## Tips zur Eingabe und zur Bedienung

»Stone Rescue« besteht aus drei Programmen: zwei Maschinenroutinen und einem Basic-Hauptprogramm. Der »Bildgenerator« wird nur ein einziges Mal zur Erzeugung von zehn Bildern benötigt. Er enthält die zehn Bilder in stark komprimierter Form.

Das Hauptprogramm lädt die Maschinenroutinen von Diskette und steuert den Spielverlauf. Es druckt das Titelbild, übergibt Parameter an die Maschinenprogramme und übernimmt die beiden Funktionen »Play« (spielen) und »Generate« (Spielfeld generieren).

Das Maschinenprogramm »STONE 1« ist das Herzstück des Programms. Es erfüllt folgende Aufgaben: Joystickabfrage, Kollisionsüberprüfung, Scoreangabe, Sound, Animation der

Schlangen und Veränderung der Grafikzeichen, Veränderung des Zeichensatzes, Bildgenerator und Laden von Bildern.

Das Programm »STONE 2« enthält die Daten der zwölf Multi-color-Sprites der Männchen und für die beiden Schlangen-Sprites. Benennen Sie bitte beide Maschinenprogramme mit dem angegebenen Namen. Die Maschinenroutinen werden nämlich vom Basic-Programm nachgeladen.

Der »Bildgenerator« dient nur zum Erstellen der ersten zehn Grafikbilder. Laden Sie dazu »STONE 1« und »STONE 2« absolut von Diskette (,8,1) und geben nach jedem Ladevorgang NEW ein. Danach können Sie den »Bildgenerator« laden und mit RUN starten. Nach etwa 25 Minuten sind die ersten zehn Level generiert und Sie können »STONE RESCUE 1« laden und starten. Das Spiel beginnt.

Wenn Sie »Stone Rescue« zum erstenmal eingeben und starten, halten Sie bitte die untenstehende Reihenfolge ein.

1. »STONE RESCUE 1« abtippen und speichern
2. NEW eingeben
3. »STONE 1« mit dem MSE eintippen und speichern
4. NEW
5. »STONE 2« wieder eingeben (MSE) und speichern
6. NEW
7. LOAD "STONE 1",8,1
8. NEW
9. LOAD "STONE 2",8,1
10. NEW
11. LOAD "BILDGENERATOR",8
12. RUN (Bilder werden auf Diskette erzeugt)
13. LOAD "STONE RESCUE 1",8
14. RUN (Spiel und Generator werden gestartet)

Haben Sie diese Prozedur einmal hinter sich gebracht, müssen Sie nur noch LOAD "STONE RESCUE 1",8 und danach RUN eingeben. Alles andere geschieht dann automatisch.

Nun zum interessantesten Teil von »Stone Rescue«, dem Bildgenerator. Im Hauptprogramm müssen Sie dazu, bei der Frage »Play or Generate«, »G« eingeben und den Joystick, wie beim Spiel, am Control-Port 1 anschließen. Sie sehen dann das Spielfeld. Mit dem Joystick fahren Sie den Cursor nun an die Stelle, an der Sie mit Ihrer Konstruktion beginnen möchten. Zum Setzen eines Zeichens drücken Sie einfach den Feuerknopf, zur Auswahl eines Zeichens die F1- (vorwärts im Zeichensatz) oder die F3-Taste (rückwärts). Das Zeichen wird rechts unten am Bildschirm und an der Cursorstelle dargestellt. Zum Abspeichern eines Bildes auf Diskette drücken Sie F4. Mit der Plus- oder Minus-Taste wird die gewünschte Bildnummer eingestellt. Mit RETURN wird mit dem Speichern begonnen. Sollte die Nummer schon vorhanden sein, fragt das Programm, ob der alte Level überschrieben werden soll. Das Laden funktioniert ähnlich dem Speichern eines Levels, nur müssen Sie F6 drücken. Sollte ein Bild nicht vorhanden sein, wird die Fehlermeldung der Diskettenstation angezeigt.