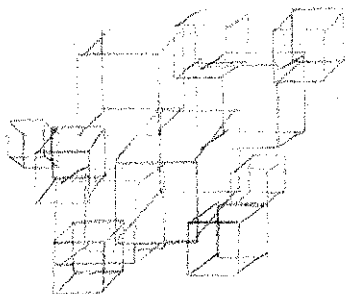


```

1      REM M          PROGRAMM : " Q U A D E R "
2      REM M          B.BROCK 7100 HEILBRONN
3      REM M          POSTFACH 1332
4      REM
5      REM
6      REM M  DAS PROGRAMM ZEICHNET ZUFAELLIGE QUADER.
7      REM M  MIT DER 'SPACE'-TASTE LAESST SICH DER BILD-
8      REM M  SCHIRM LOESCHEN !
9      REM M          V I E L  S P A S S  !
10     MODE 5
20     FILL 0,0 XMAX,YMAX 0
40     Q%=RND(15.0)+1.0
41     C%=RND(80.0)+15.0
42     S1%=SQR(C%^2.0/2.0)
43     S2%=S1%
47     X%=RND(XMAX-S1%-70.0)
48     Y%=RND(YMAX-S2%-70.0)
49     B%=RND(XMAX-S1%-X%-15.0)+15.0
50     A%=RND(YMAX-S2%-Y%-15.0)+15.0
100    IF GETC<>>0.0 THEN 10
200    GOSUB 10000
300    GOTO 40
10000  REM
10100  DRAW X%,Y% X%+B%,Y% Q%
10120  DRAW X%+B%,Y% X%+B%,Y%+A% Q%
10140  DRAW X%+B%,Y%+A% X%,Y%+A% Q%
10160  DRAW X%,Y%+A% X%,Y% Q%
10300  DRAW X%,Y%+A% X%+S1%,Y%+A%+S2% Q%
10320  DRAW X%+B%,Y%+A% X%+S1%+B%,Y%+A%+S2% Q%
10340  DRAW X%+B%,Y% X%+S1%+B%,Y%+S2% Q%
10400  DRAW X%+S1%,Y%+A%+S2% X%+S1%+B%,Y%+A%+S2% Q%
10420  DRAW X%+S1%+B%,Y%+S2% X%+S1%+B%,Y%+A%+S2% Q%
10500  FOR I%=X%+S1% TO X%+S1%+B% STEP 4.0
10520  DOT I%,Y%+S2% Q%
10540  NEXT
10560  FOR I%=Y%+S2% TO Y%+S2%+A% STEP 4.0
10580  DOT X%+S1%,I% Q%
10600  NEXT
10620  FOR I%=X% TO X%+S1% STEP 3.0
10640  DOT I%,(S2%/S1%)*(I%-X%)+Y% Q%
10660  NEXT
10900  RETURN

```



GPM

RECTANGLES

```
10 REM RECTANGLES / F.H. DRUIJFF 2/82
20 MODE 4:COLORG 0 14 1 3:GOTO 50
30 DRAW X,Y 159-X,Y C:DRAW 159-X,Y 159-X,129-Y C
40 DRAW 159-X,129-Y X,129-Y C:DRAW X,129-Y X,Y C:RETURN
50 FOR I=0 TO 20:X=80-3*I:Y=65-2*I:C=21:GOSUB 30
60 F=1-F:C=22+F:FOR J=0 TO 20:X=80-2*J:Y=65-3*J
70 GOSUB 30:NEXT:NEXT
80 FOR J=0 TO 20:X=80-2*J:Y=65-3*J:F=1-F:C=22+F:GOSUB 30:NEXT:GOTO 80
*
```

PADDLELINES

```
10 MODE 6:REM PADDLING WITH LINES F.H. DRUIJFF 2/82
20 P0=PDL(0)*XMAX/255:P1=PDL(1)*YMAX/255:P2=PDL(2)/16
30 P3=PDL(3)*XMAX/255:P4=PDL(4)*YMAX/255:P5=PDL(5)/16
40 DRAW P0,P1 P3,P4 23:COLORG P2 0 0 P5
50 IF PEEK(#FD00) IAND #30<>#30 GOTO 20:GOTO 10
*
```

PATCHWORK

```
10 REM PATCHWORK / F.H. DRUIJFF 2/82
20 MODE 1:DIM C(3):REM PROGRAM RUNS ALSO IN MODE 3 OR 5
30 FOR I=XMAX-7 TO 0 STEP -8:FOR J=0 TO YMAX-3 STEP 4
40 FOR K=0 TO 3:C(K)=RND(16):NEXT:FILL I,J I+7,J+3 RND(16)
50 FOR Y=J TO J+3:M=2-M:FOR X=I+7 TO I STEP -1:N=1-N
60 K=M+N:DOT X,Y C(K):NEXT:NEXT:NEXT:NEXT:GOTO 30
*
```

```

1  REM *****3-D SINUS PLOTT (M. RAHLFF)*****
5  MODE 0:PRINT CHR$(12)
10 ST1=0.1
15 COLORG 15 12 13 0:MODE 6
20 FOR X=0.0 TO 2.0*PI STEP ST1
30 Y=SIN(X)
40 X1=X*40.0+1.0:Y1=Y*40.0+55.0
50 DOT X1,Y1 0
60 DRAW X1,0 X1,Y1-1 13:DRAW X1,Y1 X1+20,Y1+20 12:DRAW X1
  +20,Y1+20 X1+40,Y1+40 0
65 DRAW X1+40,YMAX X1+40,Y1+40 13
70 NEXT
75 FOR I=Y1 TO 0.0 STEP -(ST1*40.0):DRAW X1,I X1+20,I+20
  12:DRAW X1+20,I+20 X1+40,I+40 0:NEXT I
76 T1=20.0
80 IF GETC<>0 THEN END
85 COLORG 15 0 12 13:WAIT TIME T1:COLORG 15 13 0 12:WAIT
  TIME T1:COLORG 15 12 13 0:WAIT TIME T1:GOTO 00

```

```

1  REM ***** PLOTT-DEMO 1 (M. RAHLFF) *****
3  FARBE=0.0:HINTERGRUND=12.0
5  COLORG HINTERGRUND FARBE 0 0
10 MODE 6
20 X0=168.0:Y0=128.0:X1=5.0:Y1=5.0
25 DIM X(4.0),Y(4.0)
30 FOR I=0.0 TO 10.4 STEP 0.2
35 FOR XY=1.0 TO 4.0
40 X(XY)=X0+X1*COS(I+(PI*XY/2.0)):Y(XY)=Y0+Y1*SIN(I+(PI*X
  Y/2.0))
50 NEXT XY
60 FOR XY=1.0 TO 3.0:DRAW X(XY),Y(XY) X(XY+1.0),Y(XY+1.0)
  FARBE:NEXT XY:DRAW X(4.0),Y(4.0) X(1.0),Y(1.0) FARBE
70 X1=X1+2.5:Y1=Y1+2.5:NEXT I
80 IF GETC=0.0 THEN 80:END

```

```

1  REM ***** PLOTT-DEMO 2 (M. RAHLFF) *****
3  FARBE=0.0:HINTERGRUND=12.0
5  COLORG HINTERGRUND FARBE 0 0
10 MODE 6
20 X0=168.0:Y0=128.0:X1=5.0:Y1=5.0
25 DIM X(4.0),Y(4.0)
30 FOR I=0.0 TO 10.4 STEP 0.2
35 FOR XY=1.0 TO 4.0
40 X(XY)=X0+X1*COS(I+(PI*XY/2.0)):Y(XY)=Y0+Y1*SIN(I+(PI*X
  Y/2.0))
50 NEXT XY
60 DRAW XMAX,YMAX X(1.0),Y(1.0) FARBE:DRAW 0,YMAX X(2.0),
  Y(2.0) FARBE:DRAW 0,0 X(3.0),Y(3.0) FARBE:DRAW XMAX,0
  X(4.0),Y(4.0) FARBE
70 X1=X1+2.5:Y1=Y1+2.5:NEXT I
80 IF GETC=0.0 THEN 80:END

```

```

IMP INT
10 REM DEMOPROGRAMM BY BERND VON OSTEN  APRIL 83
20 REM
30 REM WICHTIG : Das Graphikprg. unter NP 29 muss geladen sein
50 COLORG 0 12 3 8:MODE 6:MODE 6
60 GOSUB 3010:REM Glas zeichnen
70 F=3:N!=0.0:H!=0.0:FOR I=1 TO 2:FOR X=43 TO 127
80 DOT X,100+H!*SIN(N!) F:N!=n!+0.45:NEXT
90 F=12:N!=1.5:H!=2.3:NEXT:F=3:FOR I=1 TO 2
100 FOR J=1 TO 22:X=RND(65.0)+53.0:Y=RND(80.0)+16.0:
110 DOT X,Y F:DOT X+1,Y+1 F:DOT X-1,Y+1 F
120 DOT X,Y+2 F:DOT X,Y+1 8:DOT X,Y-2 12
130 DOT X,Y-4 3:DOT X,Y-6 12:NEXT:F=12:NEXT
140 GOSUB 2000:REM "7up" zeichnen
150 GOSUB 2200:REM "DAI" zeichnen
155 REM Farben setzen
160 POKE $BFEE,$EF:POKE $BF94,$F5:POKE $BF3A,$D1
170 POKE $A04A,$FF
175 REM Farbwechsel
180 POKE $9FF0,$D5:POKE $9F3C,$ED:GOSUB 3300
190 WAIT TIME 23
200 POKE $9FF0,$DD:POKE $9F3C,$E5:GOSUB 3300
210 WAIT TIME 23
220 GOTO 180
1900 REM
1910 REM
2000 REM --- "7 UP" zeichnen ---
2010 REM "7" zeichnen und einfaerben
2020 GOSUB 3010:X=52:Y=116:C=8:CF=12:GOSUB 3000
2030 REM "u" zeichnen und einfaerben
2040 GOSUB 3010:X=73:CF=3:GOSUB 3000
2050 REM "p" zeichnen und einfaerben
2060 GOSUB 3010:GOSUB 3010:X=97:Y=109:GOSUB 3000
2070 X=113:Y=127:GOSUB 3000:RETURN
2090 REM
2100 REM
2200 REM --- "DAI" zeichnen ---
2210 FILL 175,194 293,250 12
2220 REM "D" zeichnen
2230 GOSUB 3010:X=205:Y=222
2240 FOR N!=0.0 TO PI/4.0 step 4.6E-2:H!=COS(N!):Z=SIN(N!)
2250 S=23.0*H!:J=X+16.0*Z!:GOSUB 2900
2260 S=23.0*Z!:J=X+16.0*H!:GOSUB 2900
2270 GOSUB 3010:X=204:Y=222
2280 FOR N!=0.0 TO PI/4.0 STEP 7.1E-2:H!=COS(N!):Z!=SIN(N!)
2290 S=16.0*H!:J=X+8.0*Z!:GOSUB 2900
2300 S=16.0*Z!:J=X+8.0*H!:GOSUB 2900
2310 REM "A" zeichnen
2320 GOSUB 3010:GOSUB 3010
2330 REM "I" zeichnen
2340 GOSUB 3010
2350 REM "D","A","I" einfaerben
2360 X=187:Y=200:C=3:CF=8:GOSUB 3000
2370 X=207:Y=206:GOSUB 3000
2380 X=245:Y=218:GOSUB 3000:X=227:Y=200:GOSUB 3000
2390 X=277:GOSUB 3000:GOSUB 3010:GOSUB 3010:GOSUB 3010:RETURN
2800 REM

```

```

2810 REM
2900 DOT J,Y+S 3:DOT J,Y-S 3:RETURN
3000 VAR=X+(Y SHL 16)+(C SHL 24)+(CF SHL 28):CALLM $400,VAR:RETURN
3010 READ F,X1,Y1
3020 READ X2:IF X2≠999 THEN READ Y2:DRAW X1,Y1 X2,Y2 F:X1=X2:Y1=Y2:GOTO 3020
3030 RETURN:REM DATEN
3040 DATA 8,40,120,50,10,120,10,130,120,129,120,119,9
3050 DATA 51,9,41,120,40,120,52,8,118,8,130,120,999,8
3060 DATA 67,160,35,160,35,150,55,150,47,115,57,115
3070 DATA 67,160,999,8,75,138,70,138,70,118,73,115,85
3080 DATA 115,88,118,88,138,83,123,81,121,77,121,75
3090 DATA 123,75,138,999,8,112,138,95,138,95,108,100
3100 DATA 108,100,118,112,118,115,121,115,135,112,138
3110 DATA 999,8,100,123,110,123,112,125,112,131,110,133
3120 DATA 100,133,100,123,999,3,205,245,180,245,185,202
3130 DATA 185,202,180,199,205,199,999,3,203,206,192,206
3140 DATA 192,238,203,238,999,3,235,199,225,199,228,202
3150 DATA 240 ,245,245,245,257,202,260,199,245,199,248
3160 DATA 202,245,214,235,214,232,202,235,199,999,3,242
3170 DATA 238,244,218,237,218,242,238,999,3,288,199,270
3180 DATA 199,275,204,275,240,270,245,288,245,283,240
3190 DATA 283,204,288,199,999
3200 DATA 3,174,193,174,251,294,251,294,193,174,193,999
3210 DATA 8,173,192,173,252,295,252,295,192,173,192,999
3220 DATA 3,172,191,172,253,296,253,296,193,172,193,999
3300 I=$BF94:IF PEEK(I)=$F7 THEN POKE I,$F0:RETURN
3310 POKE I,PEEK(I)+1:RETURN

```

\$ = HEX-Zeichen

Um einen Kreis zu zeichnen haben sich die herkömmlich benutzten Programme als zu langsam erwiesen. Hier möchte ich nun ein Programm vorstellen, daß die Wartezeiten achtelt.

```
10 INPUT "Radius X-Achse,Y-Achse ";RX%,RY%:PRINT
20 INPUT "Mittelpunkt X-Achse,Y-Achse ";MX%,MY%
30 MODE 5
100 FOR N=0.0 TO PI/4.0 STEP RX%/(RX%^1.06*RX%):X=COS(N):Y=SIN(N)
110 X%=X*RX%:Y%=Y*RY%:GOSUB 140:X%=Y*RX%:Y%=X*RY%:GOSUB 140
120 NEXT:END
140 X1%=MX%+X%:X2%=MX%-X%:Y1%=MY%+Y%:Y2%=MY%-Y%
150 DOT X1%,Y1% 14:DOT X1%,Y2% 14:DOT X2%,Y1% 14:DOT X2%,Y2% 14:RETURN
```

Bei vollständigem Ausmalen der Kreise :

```
150 DRAW X1%,Y1% X2%,Y1% 14:DRAW X1%,Y2% X2%,Y2% 14:RETURN
```

Hierbei wird jeweils eine SIN- und eine COS-Berechnung für ACHT Punkte benötigt. Einen Schwachpunkt stellt hierbei die Schrittweite dar, die noch verbessert werden muß. Dieses Programm müßte in ML in Verbindung mit der AMD 9511 erheblich schneller laufen als die bisher veröffentlichten Programme.

IMP INT

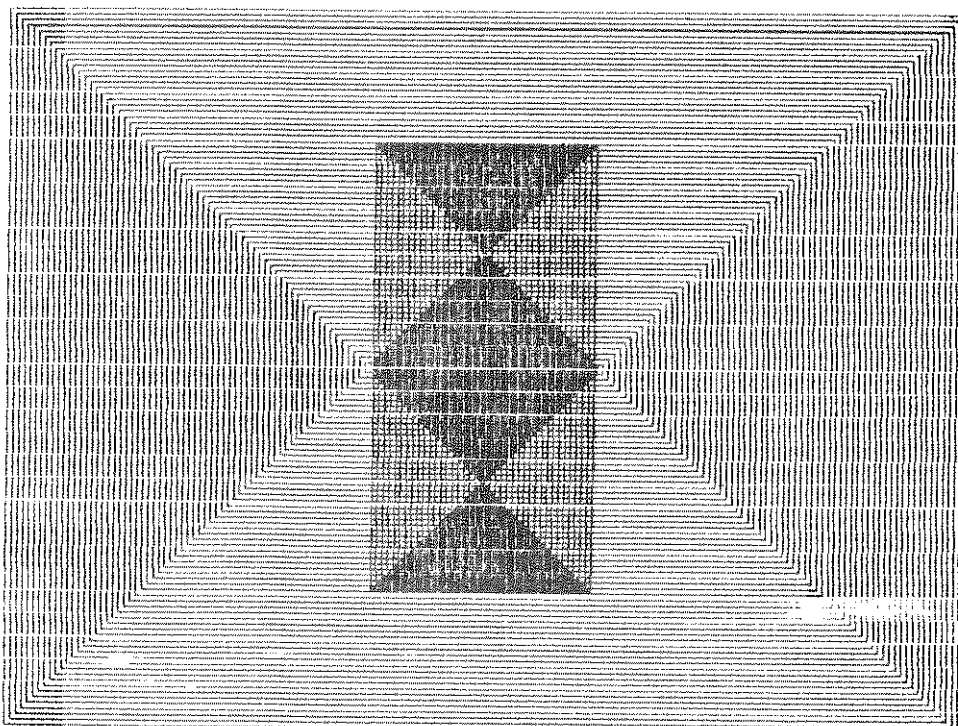
```
100 REM Ein kleines, aber feines Grafik-Demo
110 COLOR% 0 2 4 6
120 MODE 6
130 AX=0:AY=0:BX=XMAX:BY=YMAX
140 F=((AX/2 MOD 3)+21)
150 DRAW AX,AY BX,AY F:DRAW BX,AY BX,BY F
160 DRAW BX,BY AX,BY F:DRAW AX,BY AX,AY F
170 AX=AX+2:AY=AY+2:BX=BX-2:BY=BY-2
180 IF AX>335 THEN AX=AX-335
190 IF AY>255 THEN AY=AY-255
200 IF BX<0 THEN BX=BX+335
210 IF BY<0 THEN BY=BY+255
220 IF GETC<>32 THEN 140:END
```

Interessante Farbkontraste durch Andern der Off-Sets in Zeile 170 .

(z.B. AX=AX+3:AY=AY+2:BX=BX-3:BY=BY-2)

Viel Spaß beim Ausprobieren

Werner Schmitz



IMPINT

```

1   REM -- Hilfsprogramm KREIS/ELLIPSE von GERT DAHLHAUS
2   REM
8   PRINT CHR$(12);COLORG 8 0 14 4:F=14
10  INPUT "RADIUS X-ACHSE , RADIUS Y-ACHSE ";RX,RY:PRINT
20  INPUT "MITTELPUNKT X-ACHSE , Y-ACHSE ";MX,MY:PRINT
25  RX2:=RX#RX:RXY!:=RX2!/RY/RY
30  PRINT "KREIS oder VOLLKREIS (K/V) ";
40  Y!=GETC:IF Y!=0.0 GOTO 40
50  MODE 6,
60  IF MX-RX<0.0 OR MX+RX>XMAX OR MY-RY<0.0 OR MY+RY>YMAX THEN
PRINT "DER BILDSCHIRM IST LEIDER ZU KLEIN!":GOTO 10
70  IF CHR$(Y!)="V" THEN 100
80  IF CHR$(Y!)="K" THEN 200
90  GOTO 40

```

```

100 FOR Y=0 TO RY Vollkreis, - ellipse
120 X=SQR(RX2!-Y*Y#RXY!)
130 DRAW MX-X,MY-Y MX+X,MY-Y F
140 DRAW MX-X,MY+Y MX+X,MY+Y F
150 NEXT

```

```

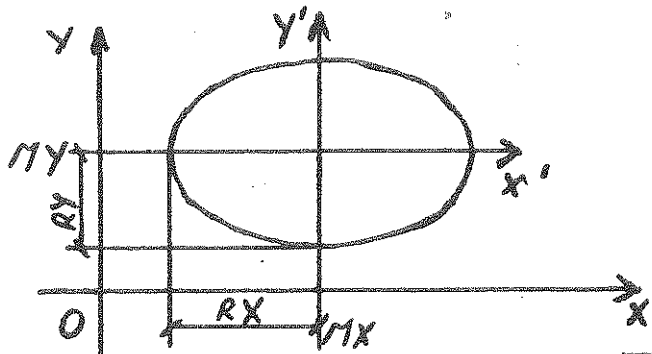
160 GOTO 300
200 X=RX Kreis , Ellipse
210 FOR Y=1 TO RY
220 XA=X:X=SQR(RX2!-Y*Y#RXY!)
230 DRAW MX-XA,MY+Y-1 MX-X,MY+Y F:DRAW MX+XA,MY+Y-1 MX+X,MY+Y F
250 DRAW MX-XA,MY-Y+1 MX-X,MY-Y F:DRAW MX+XA,MY-Y+1 MX+X,MY-Y F
280 NEXT

```

```

300 Y=GETC:IF Y=0 GOTO 300
310 MODE 0:GOTO 1
1000 REM *****
1010 REM DIESES PROGRAMM BENUTZT ZUM ZEICHNEN EINES KREISES,
1020 REM BZW. EINER ELLIPSE DIE FOLGENDE RELATION:
1030 REM
1040 REM : (X-MX)^2/RX^2 + (Y-MY)^2/RY^2 = 1
1050 REM
1060 REM ZUM ZEICHNEN WIRD DER Y-WERT JEWEILS UM 1 VOM
1070 REM MITTELPUNKT AUS NACH OBEN (140 und 230) UND
1080 REM NACH UNTEN (130 und 250) VERAENDERT.
1090 REM DER BERECHNETE X-WERT IST DER JEWEILIGE ABSTAND
1100 REM VOM X-MITTELPUNKT AUS.
1110 REM
1120 REM NACH DIESER METHODE SIND DURCH BERECHNUNG VON EINEM
1130 REM X-WERT 4 PUNKTE DES KREISES BESTIMMT.

```



$$\frac{(x-xM)^2}{RX^2} + \frac{(y-yM)^2}{RY^2} = 1$$

$$x = xM \pm \sqrt{\left(1 - \frac{(y-yM)^2}{RY^2}\right) RX^2}$$

Design und Programm von Andreas Pfeifle, Kirchheim-Teck.
Mit Modifikationen von Stefan Goller, Meckenheim.

```
10 MODE 0:PRINT CHR$(12)
20 PRINT " K A M E L - D E S I G N"
```

Die Wahlmöglichkeit für die drei Größen GR,XPO,YPO habe ich herausgenommen und durch fest vorgegebene Werte ersetzt.

```
30 REM PRINT :INPUT "ABSOLUTE GROESSE ";GR
40 REM PRINT :INPUT "X-POSITION, Y-POSITION ";XPO,YPO
45 GR=1.0;XPO%=0;YPO%=0

50 PRINT :INPUT "Laengsverzerrung 0< LZ <=1.8 ";LZ
60 PRINT :INPUT "Querverzerrung 0< QZ <=2.5 ";QZ
65 PRINT :PRINT :PRINT " Abbruch durch die Leertaste !"
70 PRINT :PRINT " 'S GEHT GLEICH LOS !!!"
80 WAIT TIME 200
90 MODE 6:COLORG 0 7 5 15
```

Schleife in der die zu zeichnenden Linien eingelesen werden.

```
100 READ X1%,Y1%,X2%,Y2%
110 IF X1%=0 OR X2%=0 OR Y1%=0 OR Y2%=0 THEN COLORG 7 0 0 0:GOTO 800
120 X1%=XPO%+X1%*GR*QZ
130 X2%=XPO%+X2%*GR*QZ
140 Y1%=YPO%+Y1%*GR*LZ
150 Y2%=YPO%+Y2%*GR*LZ
160 DRAW X1%,Y1% X2%,Y2% 7
170 GOTO 100
```

```
200 REM Vorderpfoten-Daten
210 DATA 33,10,98,10,33,10,37,14,37,14,45,15
220 DATA 45,15,57,14,57,14,56,27,49,15,52,17
230 DATA 52,17,57,17,65,14,63,19,63,19,62,29
240 DATA 64,17,67,19,67,19,63,21,73,15,77,19
250 DATA 77,19,77,24,77,24,71,23,71,23,62,27
260 REM Hinterpfoten-Daten
270 DATA 62,10,65,14,65,14,73,15,73,15,86,14
280 DATA 86,14,85,26,98,10,91,19,91,19,90,29
290 DATA 93,17,95,18,95,18,91,20,96,12,100,14
300 DATA 100,14,103,19,103,19,103,23,103,23,97,23
310 DATA 97,23,90,26
320 REM Vorderbeine-Daten
330 DATA 56,25,46,30,46,30,44,35,44,35,56,53
340 DATA 56,30,51,32,51,32,51,36,51,36,58,45
350 DATA 56,27,59,53,62,29,65,51
360 REM Hinterbeine-Daten
370 DATA 85,23,72,29,72,29,71,33,71,33,84,51
380 DATA 85,28,79,30,79,30,78,34,78,34,87,46
390 DATA 85,26,88,52,90,29,94,55
400 REM Rumpf-und Halsdaten
410 DATA 59,53,55,54,55,54,51,59,65,51,71,50
420 DATA 71,50,78,50,78,50,80,52,94,55,99,58
430 DATA 99,58,103,57,103,57,106,58,106,58,106,60
440 DATA 106,60,104,61,104,61,100,61,100,61,101,64
450 DATA 99,72,92,78,92,78,80,82,80,82,71,82
460 DATA 71,82,60,78,60,78,53,72,53,72,51,66
470 DATA 51,66,49,65,49,65,43,65,43,65,41,67
480 DATA 41,67,42,72,42,72,74,94,74,94,83,96
490 DATA 51,59,40,59,40,59,34,61,34,61,30,65
500 DATA 30,65,30,70,30,70,34,76,34,76,67,99
505 DATA 101,64,99,72
```

```

510 REM Kopf-Daten
520 DATA 83,96,99,96,99,96,109,99,109,99,119,107
530 DATA 119,107,120,112,120,112,118,118,118,118,113,123
540 DATA 113,123,104,127,104,127,92,127,92,127,84,126
550 DATA 77,126,68,122,68,122,71,120,71,120,74,120
560 DATA 81,123,74,119,74,119,74,116,74,116,79,115
570 DATA 67,122,50,132,50,132,39,135,39,135,28,135
580 DATA 28,135,16,132,16,132,8,127,8,127,6,120
590 DATA 6,120,10,116,14,124,10,120,10,120,10,114
600 DATA 10,114,16,109,16,109,24,100,24,100,32,108
610 DATA 32,100,37,110,37,110,39,107,39,107,54,102
620 DATA 54,102,67,99,67,99,84,100,82,102,84,100
630 DATA 83,98,84,100
640 REM Haar-Daten
650 DATA 97,124,93,133,93,133,86,139,86,139,82,139
660 DATA 82,139,75,135,103,122,100,130,100,130,117,138
670 DATA 117,138,120,138,120,138,121,132,110,121,128,127
680 DATA 128,127,131,125
690 REM Nasenloch-Daten
700 DATA 23,117,24,117,26,117,30,117
710 DATA 24,118,30,118,26,119,31,119,28,120,30,120
720 REM 'Hier ist endlich das Ende' - Daten
730 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

```

Zeichnet geschlossene Augen

```

800 FX=GR*QZ:FY=GR*LZ
820 DRAW 77*FX+XPO%,126*FY+YPO% 68*FX+XPO%,122*FY+YPO% 22
830 DRAW 68*FX+XPO%,122*FY+YPO% 71*FX+XPO%,120*FY+YPO% 22
840 DRAW 71*FX+XPO%,120*FY+YPO% 74*FX+XPO%,120*FY+YPO% 22
850 DRAW 81*FX+XPO%,123*FY+YPO% 74*FX+XPO%,119*FY+YPO% 22
860 DRAW 74*FX+XPO%,119*FY+YPO% 74*FX+XPO%,116*FY+YPO% 22
870 DRAW 74*FX+XPO%,116*FY+YPO% 79*FX+XPO%,115*FY+YPO% 22

```

Zeichnet offene Augen

```

880 COLORG 7 0 7 0
890 DRAW 72*FX+XPO%,125*FY+YPO% 78*FX+XPO%,125*FY+YPO% 23
900 DRAW 78*FX+XPO%,125*FY+YPO% 80*FX+XPO%,123*FY+YPO% 23
920 DRAW 72*FX+XPO%,125*FY+YPO% 69*FX+XPO%,122*FY+YPO% 23
930 DRAW 69*FX+XPO%,122*FY+YPO% 71*FX+XPO%,119*FY+YPO% 23
940 DRAW 71*FX+XPO%,119*FY+YPO% 75*FX+XPO%,120*FY+YPO% 23
950 DRAW 77*FX+XPO%,121*FY+YPO% 75*FX+XPO%,118*FY+YPO% 23
960 DRAW 75*FX+XPO%,118*FY+YPO% 76*FX+XPO%,116*FY+YPO% 23
970 DRAW 76*FX+XPO%,116*FY+YPO% 81*FX+XPO%,116*FY+YPO% 23
980 DRAW 81*FX+XPO%,116*FY+YPO% 85*FX+XPO%,118*FY+YPO% 23
990 DRAW 85*FX+XPO%,118*FY+YPO% 85*FX+XPO%,121*FY+YPO% 23
1000 DRAW 85*FX+XPO%,121*FY+YPO% 83*FX+XPO%,123*FY+YPO% 23
1010 DRAW 83*FX+XPO%,123*FY+YPO% 77*FX+XPO%,121*FY+YPO% 23
1020 FILL 75*FX+XPO%,122*FY+YPO% 77*FX+XPO%,123*FY+YPO% 23
1030 FILL 80*FX+XPO%,118*FY+YPO% 82*FX+XPO%,119*FY+YPO% 23

```

Schleife zum öffnen und schließen der Augen. Test der Abbruchbedingung.

```
1035 IF GETC=32 THEN RESTORE:AX=0:GOTO 10
```

Es könnte noch eine Zeile, die eine weitere Taste überprüft, eingeschoben werden. Im Falle, das diese gedrückt wäre, würde dann ein Bildschirmausdruck auf dem Drucker erstellt werden.

Bsp.: 1037 IF GETC=ASC("D") THEN CALLM#400

Bei #400 wäre dann ein entsprechendes Maschinenprogramm zu finden.

```
1040 COLORG 7 0 7 0:WAIT TIME 55:COLORG 7 0 0 7:WAIT TIME 40:GOTO 1035
```

```

1      REM Peter Buettner
2      REM Offenbacher Str. 6
3      REM 1000 Berlin 33
4      REM Tel.: 8228108
10     PRINT CHR$(12)
15     T$="          ENTWICKLUNG EINER COMPUTERGRAFIK"
16     U$="          IN 7 STUFEN"
20     MODE 6A:COLORG 8 0 0 0
25     GOTO 120
27     MODE 6
30     X=24.0
40     Y=216.0
50     DRAW X,(Y-INT(RND(5.0)/4.0)*4.0) X+RND(7.0),Y+4+RND(12.0) 0
60     IF X>311.0 THEN Y=Y-26.0:X=24.0
65     A=RND(10.0)-1.0:IF A>8.2 THEN A=A+10.0
67     X=X+A
70     IF Y<18.0 GOTO 100
90     GOTO 50
100    GOTO 100
110    FILL 0,0 335,255 8
120    PRINT T$:PRINT U$
121    PRINT "      1. Striche"
125    X=24.0:Y=100.0
130    DRAW X,Y X,Y+10 0
140    IF X>311.0 GOTO 160
150    X=X+5.0:GOTO 130
160    WAIT TIME 200:PRINT CHR$(12):FILL 0,0 XMAX,YMAX 8
180    PRINT "      2. Geneigte Striche"
190    X=24.0:Y=100.0
200    DRAW X,Y X+3.5,Y+10 0
210    IF X>311 GOTO 230
220    X=X+5.0:GOTO 200
230    WAIT TIME 100:PRINT CHR$(12):FILL 0,0 XMAX,YMAX 8
260    PRINT "      3. Verschieden geneigte Striche"
270    X=24.0:Y=100.0
280    DRAW X,Y X+1+RND(8.0),Y+10 0
290    IF X>311 GOTO 310
300    X=X+5.0:GOTO 280
310    WAIT TIME 100:PRINT CHR$(12):FILL 0,0 335,255 8
330    PRINT "      4. Verschieden lange Striche"
340    X=24.0:Y=100.0
350    DRAW X,Y X+1+RND(8.0),Y+4+RND(12.0) 0
360    IF X>311 GOTO 380
370    X=X+5.0:GOTO 350
380    WAIT TIME 100:PRINT CHR$(12):FILL 0,0 335,255 8
410    PRINT "      5. Mit Unterlaengen"
420    X=24.0:Y=100.0
430    DRAW X,Y-INT(RND(5.0)/4.0)*4.0 X+1+RND(8.0),Y+4+RND(12.0) 0
440    IF X>311.0 GOTO 460
450    X=X+5.0:GOTO 430
460    WAIT TIME 100:PRINT CHR$(12):FILL 0,0 335,255 8
490    PRINT "      6. Verschiedene Abstaende"
500    X=24.0:Y=100.0
510    DRAW X,Y-INT(RND(5.0)/4.0)*4.0 X+1+RND(8.0),Y+4+RND(12.0) 0
520    IF X>311 GOTO 540
530    X=X+RND(10.0)-1.0:GOTO 510
540    WAIT TIME 100:PRINT CHR$(12):FILL 0,0 335,255 8
550    MODE 0
560    PRINT :PRINT :PRINT T$
570    PRINT :PRINT U$
580    PRINT :PRINT :PRINT "      Es folgt die letzte Stufe: 'Bild einer H
andschrift'"
585    PRINT :PRINT :PRINT :PRINT "      Hierbei noch Gruppenbildung: 'Wort'
und 'Zwischenraum'"
590    .WAIT TIME 255:WAIT TIME 255
600    GOTO 27

```

Vorhang runter !

```

1      GOSUB 30010:GOSUB 1010
2      COLORT 6 0 0 0:MODE 0
3      REM ***** ein kleines Spielchen von ***
4      REM ***** klaus frerichs ***
5      REM ***** graf-sponeckstr. 4 ***
6      REM ***** D-2800 bremen-41 ***
10     PRINT "          PLATZ FUER EINEN BRIEF"
20     PRINT "          BIS ZEILE 1000"
30     PRINT "          TASTE DRUECKEN ZUM UMBLAETTERN"
200    IF GETC=0 THEN 200
201    PRINT CHR$(12)
1000   END
1010   GOTO 10100
1020   POKE X,#42:POKE X+46,#24:POKE X+92,#42
1030   POKE X+138,#81
1040   POKE X+184,#FF:POKE X+230,#18
1050   POKE X+276,#18:POKE X+322,#3C
1060   POKE X+368,#42:POKE X+414,#99:POKE X+460,#99
1070   POKE X+506,#0:POKE X+552,#0:POKE X+598,#0
1080   RETURN
10100  COLORG 6 8 0 1:MODE 4:FILL 0,0 XMAX,YMAX 8
10105  A=YMAX:F=31:F1=33:F2=35
10110  X=#BDDF
10120  GOSUB 1020
10130  WAIT TIME 20
10140  FOR X=X TO #ABAB-506 STEP -46
10142  SOUND 0 0 15 3 FREQ(F):SOUND 1 0 15 2 FREQ(F1)
10143  SOUND 2 0 15 2 FREQ(F2)
10149  F=F+2:F1=F1+3:F2=F2+4
10150  GOSUB 1020
10152  B=A-1:IF B<0 THEN B=0
10155  DRAW 62,B 89,B 1
10160  DRAW 0,A XMAX,A 6
10170  A=A-1
10180  NEXT
10190  FOR F=F TO 31 STEP -2
10192  F1=F1-2:F2=F2-3
10193  SOUND 0 0 15 3 FREQ(F):SOUND 1 0 15 2 FREQ(F1)
10195  SOUND 2 0 15 2 FREQ(F2)
10196  NEXT
10198  SOUND OFF
10200  RETURN
30000  REM DINGDONG
30010  ENVELOPE 0 15,5;14,5;13,5;12,5;10,5;8,5;6,5;0
30020  SOUND 1 0 15 0 FREQ(1000)
30025  SOUND 0 0 15 0 FREQ(1000)
30030  SOUND 2 0 15 0 FREQ(996):WAIT TIME 20
30040  SOUND 1 0 15 0 FREQ(738):SOUND 0 0 15 0 FREQ(740)
30050  SOUND 2 0 15 0 FREQ(736):WAIT TIME 30
30060  SOUND OFF
30070  RETURN

```

Eschers Quader, enthält in 20-39 das Color-Fill (NP29)-MP.
 Da das NP29 nicht das Füllen mit drei Farben hintereinander erlaubt, werden die Kanten des Quaders 2-fach neu gezeichnet, was Zeit kostet.

```

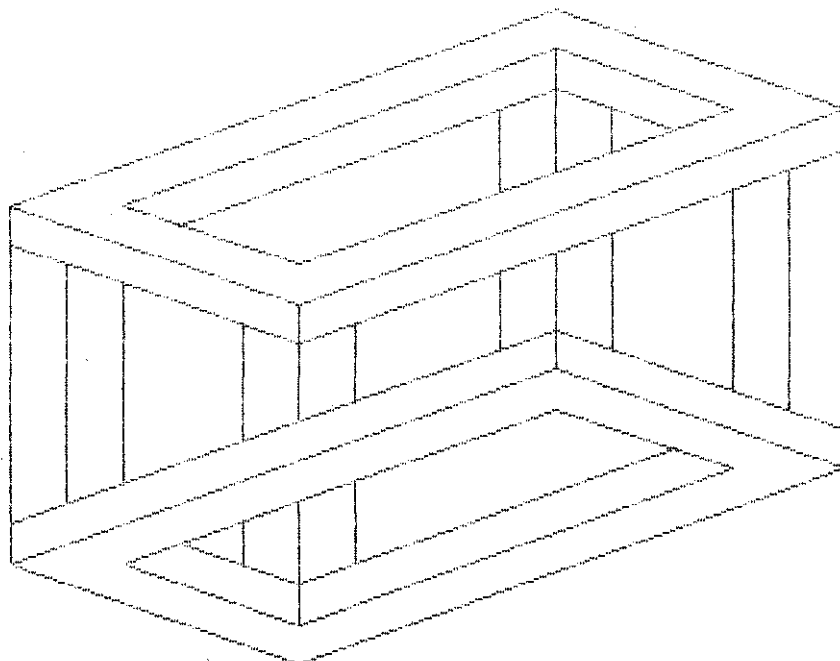
1  REM *****
2  REM *          MAURITS ESCHER ZAUBERWUERFEL          *
3  REM *                      VON                      *
4  REM *          RALF DEGENHARDT    JANUAR 84          *
5  REM *****
6  REM *          NP 29 MUSS AUF #400 GELADEN SEIN !    *
7  REM *Mit Cursor hoch-tief wird das Licht beeinflusst*
8  REM *                      IMP INT                  *
9  REM *****
10 CLEAR 900: DIM A!(150.0),A(13.0),B(37.0)
20 FOR N=#400 TO #52F: READ A: POKE N,A: NEXT
21 DATA #F5,#E5,#C5,#D5,#22,#17,#5,#7E,#E6,#F,#32,#22,#5,#7E,#E6,#F0
22 DATA #F,#F,#F,#F,#32,#23,#5,#23,#7E,#32,#1B,#5,#23,#7E,#32,#1A
23 DATA #5,#23,#7E,#32,#19,#5,#2A,#19,#5,#23,#22,#19,#5,#2A,#19,#5
24 DATA #2B,#22,#19,#5,#3A,#1B,#5,#4F,#CD,#7,#5,#C2,#2D,#4,#2A,#19
25 DATA #5,#23,#22,#19,#5,#22,#1C,#5,#3A,#1B,#5,#32,#1E,#5,#3E,#0
26 DATA #32,#1F,#5,#2A,#19,#5,#2B,#22,#20,#5,#23,#EB,#2A,#1C,#5,#7D
27 DATA #BB,#C2,#81,#4,#7C,#BA,#C2,#81,#4,#3A,#1E,#5,#5F,#3A,#1B,#5
28 DATA #BB,#C2,#81,#4,#3A,#1F,#5,#FE,#1,#CA,#12,#5,#3E,#1,#32,#1F
29 DATA #5,#2A,#20,#5,#23,#22,#20,#5,#3A,#1B,#5,#4F,#CD,#7,#5,#C2
30 DATA #81,#4,#2A,#20,#5,#2B,#EB,#3A,#1B,#5,#47,#4F,#2A,#19,#5,#3A
31 DATA #23,#5,#EF,#21,#DA,#2,#E6,#3A,#1B,#5,#3C,#4F,#2A,#19,#5,#CD
32 DATA #7,#5,#C2,#E9,#4,#3A,#1B,#5,#4F,#2A,#19,#5,#23,#CD,#7,#5
33 DATA #C2,#F3,#4,#3A,#1B,#5,#3D,#4F,#2A,#19,#5,#CD,#7,#5,#C2,#FD
34 DATA #4,#3A,#1B,#5,#4F,#2A,#19,#5,#2B,#CD,#7,#5,#CA,#53,#4,#2A
35 DATA #19,#5,#2B,#22,#19,#5,#C3,#C3,#4,#3A,#1B,#5,#3C,#32,#1B,#5
36 DATA #C3,#D1,#4,#2A,#19,#5,#23,#22,#19,#5,#C3,#A7,#4,#3A,#1B,#5
37 DATA #3D,#32,#1B,#5,#C3,#B5,#4,#EF,#27,#DA,#2,#E6,#4F,#3A,#22,#5
38 DATA #B9,#C9,#D1,#C1,#E1,#F1,#C9,#0,#0,#0,#0,#0,#0,#0,#0,#0
39 DATA #0,#0,#0,#0,#FF,#FF,#FF,#FF,#FF,#FF,#FF,#FF,#FF,#FF,#FF,#FF
40 COLORG 5 0 10 15: MODE 6: MODE 6
50 Q=0: GOSUB 160
60 C=0: CF=15
70 FOR N=1 TO 10
71 READ I,J: GOSUB 220
72 NEXT N
80 CF=10
81 FOR N=1 TO 10
82 READ I,J: GOSUB 220
83 NEXT N
90 COLORG 0 5 10 15
100 Q=15
101 RESTORE: GOSUB 180
110 READ A: IF A<>999 THEN 110
111 C=15: CF=5
112 FOR N=1 TO 4
113 READ I,J: GOSUB 220
114 NEXT N
120 Q=0
121 RESTORE: GOSUB 180
130 A!=0.5*PI: B!=PI: C!=4E-2
131 K=GETC
132 IF K=16 THEN C!=C!*2.0: IF C!>1.0 THEN C!=1.0
133 IF K=17.0 THEN C!=C!*0.5: IF C!<1E-2 THEN C!=1E-2
134 FOR N!=0.0 TO 2.0*PI STEP C!
135 COLORG 0 9.0+6.1*SIN(N!+A!) 9.0+6.1*SIN(N!) 9.0+6.1*SIN(N!+B!)
136 NEXT N!
137 GOTO 131
160 A=0: B=0
161 FOR N=0 TO 6
162 A(N)=A:A(13.0-N)=312-A: IF N=3 THEN A=A+3

```

```

163 A=A+21:NEXT N
170 FOR N=0 TO 18
171 B(N)=B:B(37.0-N)=255-B:READ C:B=B+C
172 NEXT N
180 READ A:IF A<>888 THEN 180
181 X1=A(5.0):Z=12:GOSUB 200
182 X1=A(8.0):Z=10:GOSUB 200
183 X1=A(0.0):Z=2:GOSUB 200
184 X1=A(13.0):GOSUB 200
185 X1=A(1.0):Z=1:GOSUB 210
186 X1=A(2.0):GOSUB 210
187 X1=A(11.0):GOSUB 210
188 X1=A(12.0):GOSUB 210
190 Z=2
191 FOR M=1 TO 6
192 X1=A(M+3.0):GOSUB 210
193 NEXT M
194 RETURN
200 FOR N=1 TO Z
201 READ A,B,C:DRAW X1,B(A) A(B),B(C) Q
202 NEXT N
203 RETURN
210 FOR N=1 TO Z
211 READ A,B:DRAW X1,B(A) X1,B(B) Q
212 NEXT N
213 RETURN
220 VAR=A(I)+1+(B(J)+2 SHL 16.0)+(C SHL 24.0)+(CF SHL 28.0)
221 CALLM #400,VAR
222 RETURN
300 DATA 15,16,8,8,8,8,8,5,8,3,4,4,4,3,4,9,7,3,-125
310 DATA 888
320 DATA 0,0,3,0,13,8,1,2,3,1,11,8,2,3,4,2,10,9
321 DATA 18,0,26,18,13,32,21,0,29,21,13,34,24,2,29,24,11,34
322 DATA 37,0,29,37,13,34,36,2,29,36,11,34,35,3,28,35,10,33
323 DATA 19,0,5,19,13,11,13,2,3,13,11,8
324 DATA 3,0,29,3,8,16,8,13,34,8,8,16
325 DATA 6,24,7,22,15,30,13,31
326 DATA 3,5,10,20,1,6,12,21,3,7,14,20,17,23,30,34,16,25,31,36,17,27,32,
34
330 DATA 5,1,4,3,4,10,4,20,0,5,0,3,7,17,7,30,10,33,11,15
331 DATA 4,2,5,2,5,12,5,18,12,13,12,10,8,19,8,31,3,28,1,6
340 DATA 999
350 DATA 5,0,6,1,5,21,6,24

```



Dies Programm zeichnet eine 'Kugel' dreidimensional als durchsichtiges Gitter.
 Die einzugebenden Werte bestimmen die Anzahl der Gitterflächen horizontal und vertikal; gebräuchliche Werte sind z.B. 40,20 oder 60,30 (50,3 ist auch nicht schlecht! (Red.)).

Die Zeichengeschwindigkeit ist mit AMD 9511 schneller.

Rotationskörper enthält in den Zeilen 31-54 das MODE 8 (NP26)-MP, das Programm selbst läßt sich auch in MODE 6 gut verwenden, dazu ist zu ändern:

	MODE 6	MODE 8	Bedeutung
Zeile 151	120	110	Y >
Zeile 161	30	40	Z (Tiefe) > des Körpers
Zeile 162	120	250	X >
Zeile 162	167.5	255.5	X-Mittelpunkt
Zeile 163	127.5	119.5	Y-Mittelpunkt

```

1  REM *****
2  REM *           ROTATIONSKOERPER (KUGEL)           *
3  REM *                   VON                       *
4  REM *           RALF DEGENHARDT   JANUAR 84       *
5  REM *****
6  REM * NP 26 (MODE 8A) MUSS AUF #300 GELADEN SEIN *
7  REM *
8  REM * MITTELPUNKTSKOORDINATEN IN MODE 8:         *
9  REM * X:255.5  Y:119.5                          *
10 REM * HOEHE:110      BREITE:250      TIEFE:40    *
11 REM * BREAK NICHT BETAETIGEN,SONDERN ASCII-TASTE *
12 REM *****
13 REM
14 REM
15 REM
16 REM
20 CLEAR 5000:DIM Z(99.0)
21 REM
22 REM
23 REM
24 REM
31 FOR NZ=#300 TO #46F:READ AZ:POKE NZ,AZ:NEXT
32 DATA #C5,#3A,#40,#0,#F5,#3E,#BF,#32,#40,#0,#32,#6,#FD,#3A,#FF,#2
33 DATA #FE,#F,#CA,#4C,#3,#F5,#3A,#9D,#0,#FE,#F,#CA,#B5,#3,#21,#0
34 DATA #4,#CD,#46,#E5,#CD,#F3,#3,#F1,#6,#6,#11,#9E,#0,#CD,#FC,#E5
35 DATA #21,#4F,#42,#6,#F,#CD,#FC,#E5,#32,#9D,#0,#3E,#42,#1,#F0,#30
36 DATA #11,#0,#0,#2A,#82,#0,#CD,#A4,#3,#C3,#97,#3,#F5,#3A,#9D,#0
37 DATA #FE,#E,#CA,#D3,#3,#21,#20,#4,#CD,#46,#E5,#21,#E7,#50,#CD,#87
38 DATA #E6,#11,#CF,#4E,#CD,#FD,#E1,#2A,#80,#0,#F1,#6,#6,#11,#9E,#0
39 DATA #CD,#FC,#E5,#21,#CF,#4E,#6,#F,#CD,#FC,#E5,#CD,#A0,#3,#2A,#82
40 DATA #0,#1,#D4,#30,#11,#0,#0,#32,#9D,#0,#3E,#42,#CD,#A4,#3,#E
41 DATA #1C,#2A,#BE,#0,#CD,#A4,#3,#F1,#32,#40,#0,#32,#6,#FD,#C1,#C9
42 DATA #F5,#C3,#2B,#E5,#F5,#C5,#D5,#E5,#C3,#E6,#E5,#F5,#C3,#8B,#E4,#F5
43 DATA #F5,#D5,#C3,#D2,#E4,#F1,#32,#9D,#0,#21,#0,#4,#CD,#46,#E5,#CD
44 DATA #F3,#3,#CD,#AB,#3,#21,#4F,#42,#11,#9E,#0,#6,#F,#CD,#FC,#E5
45 DATA #C3,#97,#3,#21,#20,#4,#CD,#46,#E5,#F1,#AF,#CD,#AF,#3,#21,#CF
46 DATA #4E,#11,#7C,#0,#6,#F,#CD,#FC,#E5,#3E,#F,#32,#9D,#0,#C3,#97
47 DATA #3,#0,#0,#0,#0,#0,#0,#2A,#90,#0,#CD,#87,#E6,#2A,#80,#0,#C9
48 DATA #C0,#7D,#B8,#E,#B0,#7D,#B0,#7D,#E8,#7F,#C0,#7D,#58,#8C,#40,#71

```

```

49 DATA #0,#2,#F0,#1C,#86,#0,#0,#0,#0,#0,#0,#0,#0,#0,#0,#0
50 DATA #E8,#7F,#B8,#E,#8,#6F,#18,#6F,#30,#71,#40,#71,#58,#8C,#B0,#7D
51 DATA #0,#2,#F0,#1C,#86,#0,#0,#0,#0,#0,#0,#0,#0,#0,#0,#0
52 DATA #E1,#E3,#F5,#7E,#FE,#3,#CA,#4F,#4,#F1,#E3,#E5,#C3,#FD,#C6,#3A
53 DATA #9D,#0,#FE,#C,#DA,#49,#4,#1F,#DA,#49,#4,#17,#3C,#32,#FF,#2
54 DATA #E5,#D5,#CD,#0,#3,#D1,#E1,#C3,#49,#4,#0,#0,#0,#0,#0,#0
55 REM
56 REM
57 REM
58 REM
100 COLORG 0 5 10 15
110 POKE #2FF,#F:CALLM #300
111 PRINT "NACH DEM BILDAUFBAU MIT 'SPACE' NEUE KOORDINATEN EINGEBEN"
112 INPUT "ELEMENTE:  HORIZ.: ( 0>X<256 ) , VERT.: ( 0>Y<121 )";Z1%,Z2%
113 IF Z1%=0.0 OR Z2%=0.0 THEN 111
120 DIM A%(1.0,Z1%),B%(1.0,Z1%)
130 C=Z1%/2.0/PI:D=Z2%/PI
140 POKE #2FF,#E:CALLM #300:CALLM #300
150 FOR J%=0 TO Z2%
151 R=SIN(J%/D):S=110.0*COS(J%/D)
160 FOR I%=0 TO Z1%
161 T=40.0*R*SIN(I%/C)
162 A%(E%,I%)=255.5+250*R*COS(I%/C)+T
163 B%(E%,I%)=119.5+S+T
165 IF I%=0 OR J%=0 THEN 180
170 DRAW A%(E%,I%-1.0),B%(E%,I%-1.0) A%(E%,I%),B%(E%,I%) 23
171 DRAW A%(E%,I%),B%(E%,I%) A%(F%,I%),B%(F%,I%) 23
180 NEXT I%
181 E%=1-E%
182 F%=1-E%
190 NEXT J%
200 K%=GETC:IF K%<>32 THEN 200
201 GOTO 110

```


IMP INT

```
1      REM *****
2      REM *           H O R I Z O N T           *
3      REM * VON           RALF DEGENHARDT           1/84 *
4      REM *****
5      REM * CURSOR HOCH   = SCHNELLER           *
6      REM * CURSOR TIEF   = LANGSAMER           *
7      REM * CURSOR RECHTS = VORWAERTS           *
8      REM * CURSOR LINKS  = RUECKWAERTS         *
9      REM *****
10     COLRG 0 13 14 15:MODE 6:MODE 6
11     A=7
12     FOR N=6 TO 334 STEP 24
13     DRAW 205,165 N,A 21:NEXT N
14     A=253:B=B+1:IF B=2 THEN 70
15     GOTO 30
16     A=6
17     FOR N=7 TO 247 STEP 24
18     DRAW 205,165 A,N 21:NEXT N
19     A=334:B=B+1:IF B=4 THEN 120
20     GOTO 80
21     Q=24:P=5:A!=165.0:A=A!:R!=1.0
22     Q=Q-1:IF Q=20 THEN Q=23
23     GOSUB 300
24     R!=R!+0.3:A!=A!-20.0/R!:A=A!:IF A!<1.0 THEN R=1:GOTO 170
25     GOTO 130
26     Z=GETC
27     IF Z=16 THEN P=P-1:IF P=1 THEN P=2
28     IF Z=17 THEN P=P+1
29     IF Z=18 THEN R=2
30     IF Z=19 THEN R=1
31     ON R GOTO 180,190
32     COLRG 0 0 15 15:WAIT TIME P
33     COLRG 0 15 0 15:WAIT TIME P
34     COLRG 0 15 15 0:WAIT TIME P
35     GOTO 170
36     COLRG 0 15 15 0:WAIT TIME P
37     COLRG 0 15 0 15:WAIT TIME P
38     COLRG 0 0 15 15:WAIT TIME P
39     GOTO 170
40     B=INT(3.0/4.0*A+0.5)
41     C=INT(XMAX/2.0+3.0*R!)
42     D=INT(YMAX/2.0+3.0*R!)
43     E=C+A:F=C-A:G=D+B:H=D-B
44     DRAW F,H E,H 0
45     DRAW E,H E,G 0
46     DRAW E,G F,G 0
47     DRAW F,G F,H 0
48     RETURN
```

IMP INT

```

1  REM *****
2  REM *          DEMO FUER          *
3  REM *          P A I N T          *
4  REM *          VON                *
5  REM *          RALF DEGENHARDT    3/84 *
6  REM *****
7  REM * PAINT (NF35) MUSS AUF #2000 GELADEN SEIN *
8  REM *****
10 MODE 0:MODE 6
15 DRAW 0,0 XMAX,0 21:DRAW XMAX,0 XMAX,YMAX 21
16 DRAW XMAX,YMAX 0,YMAX 21:DRAW 0,YMAX 0,0 21
18 FOR N=1 TO 250
20 COLOR# RND(16.0) RND(16.0) RND(16.0) RND(16.0)
30 A=RND(300.0):B=RND(220.0):C=26-B/11:D=C*0.5:H=C+D
31 IF A+H>335.0 THEN 30
32 IF B+H>255.0 THEN 30
40 E=A+C:F=A+D:G=A+H:I=B+D:J=B+C:K=B+H
50 DRAW A,B E,B 21:DRAW A,B A,J 21:DRAW E,B E,J 21
51 DRAW A,J E,J 21:DRAW A,J F,K 21:DRAW E,J G,K 21
52 DRAW E,B G,I 21:DRAW G,I G,K 21:DRAW G,K F,K 21
60 CALLM #2003
61 R=E+1:S=B+2:T=INT(RND(4.0)):GOSUB 100:CALLM #2006
62 R=G-1:S=K-2:GOSUB 100:CALLM #2006
63 R=A+1:S=B+1:T=INT(RND(4.0)):GOSUB 100:CALLM #2006
64 R=E-1:S=J-1:GOSUB 100:CALLM #2006
65 R=A+2:S=J+1:T=INT(RND(4.0)):GOSUB 100:CALLM #2006
66 R=G-2:S=K-1:GOSUB 100:CALLM #2006
70 CALLM #2009
80 NEXT
81 WAIT TIME 250:GOTO 10
100 IF R>255 THEN POKE #200C,R-256:POKE #200D,1:GOTO 102
101 POKE #200C,R:POKE #200D,0
102 POKE #200E,S
103 POKE #200F,T
104 POKE #2010,1
105 RETURN

```