

```

1  REM *****
2  REM *****
3  REM ***** MAZE RUN *****
4  REM ***** COLIN HARDS *****
5  REM ***** DEC 1981 *****
6  REM *****
7  REM *****
10 MODE 0:PRINT CHR$(12):COLORT 7 3 7 7:COLORG 8 1 3 5
20 CLEAR 1000:R%=41:DIM A%(R%),B%(R%),C%(R%),D%(R%)
30 ENVELOPE 0 15,5;11,5;9,5;7,5;5,5;3,5;0
40 POKE #BA2D,#5A:CURSOR 0,12:PRINT "* MAZE RUN *"
50 REM ..... Fill Array .....
60 FOR Q%=0 TO R%:READ A%(Q%),B%(Q%),C%(Q%),D%(Q%):NEXT Q%
70 WAIT TIME 120
80 PRINT CHR$(12):MODE 2A:COLORT 8 0 8 3
1000 REM ..... Draw maze .....
1010 FOR Q%=0 TO R%
1020 DRAW A%(Q%),B%(Q%) C%(Q%),D%(Q%) 1:NEXT Q%
1200 REM ..... SET UP .....
1210 DOT 68,50 5:DOT 2,0 3:X%=2:Y%=0:PRINT CHR$(12)
1230 FOR C%=15 TO 19:POKE (#BAE6-2*C%),#FF:NEXT C%
1240 REM FOR C%=28 TO 32:POKE (#B9DA-2*C%),#FF:NEXT C%
1250 CURSOR 3,3:PRINT "TIME ="
1260 CURSOR 24,3:PRINT "Best times so far are ";T1%
1270 CURSOR 46,2:PRINT T2%
1280 CURSOR 46,1:PRINT T3%
1320 Z%=GETC:IF Z%=0 THEN 1320
1330 IF Z%=9 GOTO 1320
2000 REM ..... RUN .....
2010 Z%=GETC
2020 IF Z%=16 THEN GOTO 3000
2030 IF Z%=17 THEN GOTO 3020
2040 IF Z%=18 THEN GOTO 3040
2050 IF Z%=19 THEN GOTO 3050
2090 T%=T%+1:CURSOR 10,3:PRINT T%:GOTO 2000
3000 DOT X%,Y% 8:Y%=Y%+1:GOSUB 5000:DOT X%,Y% 3:GOTO 2000
3020 DOT X%,Y% 8:Y%=Y%-1:GOSUB 5000:DOT X%,Y% 3:GOTO 2000
3040 DOT X%,Y% 8:X%=X%-1:GOSUB 5000:DOT X%,Y% 3:GOTO 2000
3050 DOT X%,Y% 8:X%=X%+1:GOSUB 5000:DOT X%,Y% 3:GOTO 2000
5000 REM ..... WALL HIT .....
5010 W%=SCRN(X%,Y%)
5020 IF W%=8 THEN RETURN
5030 IF W%=5 THEN 5200
5040 REM IF W%=3 THEN 5500
5050 SOUND 0 0 15 0 FREQ(50.0)
5060 NOISE 0 15
5070 WAIT TIME 20
5080 W%=0:T%=T%+50:SOUND OFF :NOISE OFF :DOT X%,Y% 1
5090 IF Z%=16 THEN Y%=Y%-1:GOTO 5130
5100 IF Z%=17 THEN Y%=Y%+1:GOTO 5130
5110 IF Z%=18 THEN X%=X%+1:GOTO 5130
5120 IF Z%=19 THEN X%=X%-1:GOTO 5130
5130 DOT X%,Y% 3:RETURN

```

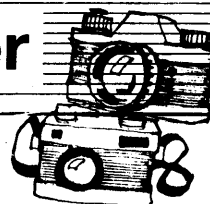
```

5200 REM ..... FINISH .....
5210 PRINT CHR$(12)
5220 IF T1%=0 THEN 5300
5230 IF T1%<T2% THEN 5290
5240 IF T2%=0 THEN 5320
5250 IF T2%<T3% THEN 5310
5260 IF T3%=0 THEN 5330
5270 IF T3%<T1% THEN 5330
5280 GOTO 5340
5290 T3%=T2%:T2%=T1%
5300 T1%=T%:GOTO 5340
5310 T3%=T2%

5320 T2%=T%:GOTO 5340
5330 T3%=T%
5340 REM FOR C%=33 TO 37:POKE (#BAE6-2*C%),#FF:NEXT C%
5350 REM FOR C%=28 TO 32:POKE (#B9DA-2*C%),#FF:NEXT C%
5360 CURSOR 3,3:PRINT "For another game press TAB           Best times ";T1%
5371 CURSOR 3,2:PRINT "Press E to end                       ";T2%
5380 CURSOR 3,1:PRINT "Your time was ";T%
5390 CURSOR 49,1:PRINT TT3%
5400 Z%=GETC:IF Z%=0 THEN 5400
5410 IF Z%=9 THEN T%=0:GOTO 1200
5420 IF Z%=69 OR Z%=101 THEN 5440
5430 GOTO 5400
5440 PRINT CHR$(12):COLORT 15 3 0 0
5450 PRINT " Best time was ";T1%;"           ":END
10000 DATA 0,0,0,52,0,52,71,52
10010 DATA 71,52,71,0,71,0,4,0
10020 DATA 4,0,4,25,0,29,8,29
10030 DATA 8,29,8,4,8,4,12,4
10040 DATA 16,0,16,8,16,8,12,8
10050 DATA 12,8,12,33,12,33,4,33
10060 DATA 4,33,4,48,4,48,8,44
10070 DATA 5,52,13,44,13,43,13,39
10080 DATA 8,39,16,39,16,38,16,12
10090 DATA 17,12,20,12,20,11,20,4
10100 DATA 21,4,67,4,71,8,24,8
10110 DATA 24,8,24,39,20,39,28,39
10120 DATA 20,40,20,47,21,47,60,47
10130 DATA 64,51,64,35,63,43,24,43
10140 DATA 36,42,29,35,29,34,29,12
10150 DATA 30,13,40,23,35,9,48,22
10160 DATA 48,23,48,28,47,28,32,28
10170 DATA 33,29,42,38,43,38,60,38
10180 DATA 64,34,43,34,52,33,52,18
10190 DATA 51,17,46,12,47,12,67,12
10200 DATA 71,16,64,16,64,17,64,30
59999 GOTO 59999

```

Screeendump zonder printer



Een screeendump zonder printer.

Bij het lezen van de titel van dit artikel dacht u waarschijnlijk: dat kan niet. Maar het kan wel, zelfs in 16 kleuren. Hoe? Door fotograferen van het beeldscherm. Moeilijk is dat niet, als u over de juiste apparatuur beschikt en het onderstaande in acht neemt. Als toepassingen noem ik het zelf maken van wenskaarten en titeldia's voor uw vakantieserie. U hebt nodig: een spiegelreflexcamera met een lens van goede kwaliteit (een macrolens is ideaal), statief een draad- of zelfontspanner en een zeer stabiel beeld. Goede resultaten bereikt u alleen als u met RGB werkt en liefst ook met een monitor. De weg naar goede opnamen is als volgt:

- * Zet de camera op statief. Zorg ervoor dat de achterwand van de camera evenwijdig is met de voorkant van de beeldbuis.
- * Zet de camera op een zodanige afstand van het scherm, dat u dit net niet helemaal in de zoeker hebt.
- * Haal een eventueel anti-reflectiescherm voor de beeldbuis weg.
- * Gebruik bij voorkeur een film met een gevoeligheid van ISO 100. Printmachines in ontwikkelcentrales verslikken zich wel eens in de DAI-kleuren en dan leveren ze veel te fletse foto's af. Laat uw film daarom afwerken onder de garantie niet goed-gratis opnieuw afdrucken. Diafilms kunnen perfecte resultaten leveren.
- * In de kamer moet het volkomen donker zijn.
- * Stel zeer nauwkeurig scherp.
- * De belichting: voor een gelijkmatig belichte opname moet de belichtingstijd minimaal 1/4 sec. zijn. Zelf werk ik als volgt: ik stel een geschikte combinatie van sluitertijd en diafragma in en regel dan de helderheid van mijn monitor totdat de belichtingsmeter van de camera aangeeft dat ik goed zit. Voor een ISO 100 film zijn er de volgende mogelijkheden:

1 sec. bij 11 voordeel: door de grote scherptediepte zijn ook de hoeken goed scherp,

nadeel : door de lange tijd last van beeldtrilling,

1/4 sec. bij 5.6 voordeel: door de korte tijd weinig tot geen last van beeldtrilling,

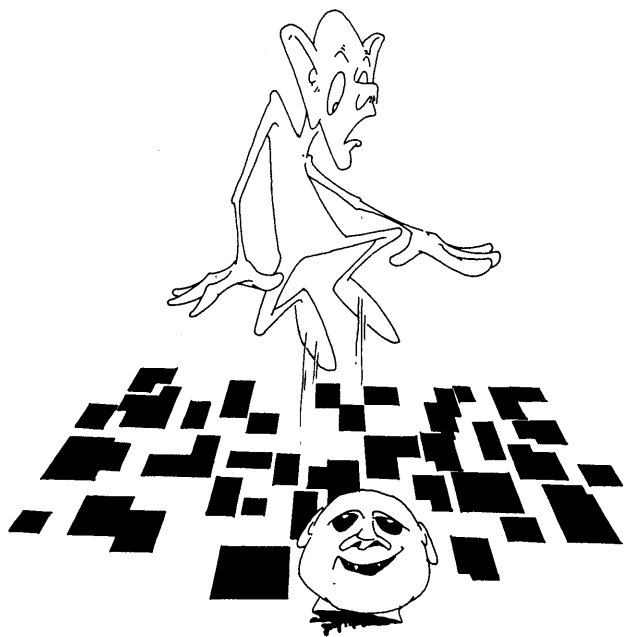
nadeel : de hoeken van het beeld zullen niet helemaal scherp zijn,

1/2 sec. bij 8 een compromis waar ik zelf meestal voor kies.

Waarschijnlijk zal de helderheid van het beeld iets groter moeten zijn dan normaal.

Als u zich aan deze punten houdt, dan zult u uitstekende resultaten bereiken. Ik wens u veel plezier en succes bij uw opnamen.

Aad de Bruijn
Heerenlaan 14
NL-3218 VL Heenvliet



PAGE 01 -- DOOLHOF

```

100 REM *** DWARS DOOR DE DOLLE
    DOOLHOF ***
110 REM *** DE BONT C *** 30-3-1985
    *****
120 GOTO 4000

200 REM *** [S] GETC-ROUTINE
-----
210 G=GETC:G=GETC:G=GETC:W=0:POKE
    #75,95
220 CC=RND(16):COLORT 8 0 8 CC:IF
    CC=8 THEN 220
230 SOUND 1 0 9 0 FREQ(RND(900)+50):
    WAIT TIME 3:SOUND OFF
240 G=GETC:IF G=0 THEN W=W+1:WAIT
    TIME 3:IF W<30 THEN 240
250 IF G=0 THEN W=0:GOTO 220
260 POKE #75,32:COLORT 8 0 8 14:
    RETURN

300 REM *** [S] CLOSE A DOOR (XD,
    YD) -----
310 P=1:FILL XD,YD XD+10,YD+10 20:
    PRINT CHR$(12);
320 FOR I=1 TO 5:DRAW XD+3,YD+I XD+
    8,YD+I 21
330 1 DRAW XD+3,YD+10-I XD+8,YD+10-I
    1 21:FOR Q=1 TO 100:NEXT
340 NEXT:LH=0:RETURN
    
```

```

400 REM *** [S] HINDERNIS
-----
410 HR=RND(9)+1:IF HR=1 AND
    RND(100)>2 THEN 410
420 PRINT CHR$(12);:BC=2:GOSUB 900:
    FOR I=1 TO 10
430 1 SOUND 1 0 15 3 FREQ(800):FOR Q=
    1 1 TO 100:NEXT
440 SOUND 1 0 15 3 FREQ(400):FOR Q=
    1 TO 100:NEXT:NEXT
450 SOUND OFF :IF HR=1 THEN 500
460 IF HR=2 THEN 600
470 IF HR=9 THEN 800
480 GOTO 700

500 REM *** [S] SLANGEKUIL
-----
510 PRINT K$;"U VALT IN DE
    SLANGEKUIL ! VAARWEL VRIEND !!"
520 PRINT K$;"HET SPEL IS VOOR U
    DEFINITIEF AFGELOPEN !!";
530 G$=CHR$(102):GOSUB 900:FOR Q=1
    TO 100:NEXT
540 G$=CHR$(103):TC=21:GOSUB 900:
    RETURN

600 REM *** [S] VALKUIL
-----
--
610 PRINT K$;"U BELANDT IN EEN
    VALKUIL.U KLIMT ERUIT,MAAR"
620 PRINT K$;"DAT KOST U";
630 G$=CHR$(101):GOSUB 900:SE=
    INT(100*RND(1))+20:SF=SF+SE
640 PRINT SE;" KOSTBARE
    STRAF-SECONDEN !!";:GOTO 840

700 REM *** [S] SPRINGPLANK
-----
710 PRINT K$;"U KOMT OP EEN
    SPRINGPLANK TERECHT !! VOOR U"
720 PRINT K$;"HET WEET BENT U OP
    EEN ANDERE PLEK BELAND!";
730 G$=CHR$(100):GOSUB 900:G$=" "
740 FILL XM,YM XM+10,YM+10 22:XI=XM:
    YI=YM
750 X=INT(28*RND(1))+1:Y=
    INT(18*RND(1))+1:XM=(X*10)+20
760 YM=(Y*10)+10:IF SCRNX(XM+5,YM+5)=
    0 THEN 750
770 XH=XM:YH=YM:BC=2:IF SCRNX(XM+5,
    YM+2)=8 THEN BC=0
780 GOSUB 900:GR=SCRNX(XM+5,YM+2):XG=
    XM:YG=YM:G$=CHR$(99)
790 GOSUB 900:FILL XI,YI XI+10,YI+
    10 22:GOTO 840
    
```

```

800 REM *** [S] VOEDSELPAKKET
-----
810 PRINT K$;"U VINDT 'N
VOEDSELPAKKET.NA 'N FLINKE
MAAL-"
820 PRINT K$;"TIJD GAAT U VERDER.U
WINT 100 SECONDEN !!!";
830 G$=CHR$(98):GOSUB 900:LIM=LIM+
100
840 CURSOR 10,1:PRINT "DRUK OP EEN
TOETS VOOR 'T VERVOLG";
850 LH=0:SH=0:G=GETC:G=GETC:G=GETC
860 G=GETC:IF G=0 THEN 860
870 G$=CHR$(99):GOSUB 900:PRINT
CHR$(12);:RETURN

900 REM *** [S] CALL OF FGT (XG,YG,
TH,TC,HS,VS,G$,BF) ----
910 SOUND 1 0 15 3 FREQ((XG+YG+
25)*2):POKE #2F0,BC*#40+TC
920 POKE #2F1,BF*#80+TH:POKE #2F2,
XG MOD 256
930 POKE #2F3,XG SHR 8:POKE #2F4,YG:
POKE #2F5,HS
940 POKE #2F6,VS:CALLM #300,G$:
SOUND OFF :RETURN

1000 REM *** DOOLHOF-GENERATOR
(GEWIJZIGDE VERSIE UIT
1010 REM *** NOS-HOBBYSLOOP
UITZENDING VAN 26-6-1983)
1020 CURSOR 20,0:PRINT "EVEN GEDULD
A.U.B!";:X=16:Y=10

PAGE 03 -- DOOLHOF

1030 D(X,Y)=9:FILL 11,0 319,210 21:
TH=0
1040 T=0:R1=0:R2=0:R3=0:R4=0:IF X+
2>28 THEN 1070
1050 IF D(X+2,Y)<>0 THEN 1070
1060 DX=2:DY=0:T=T+1:R1=1
1070 IF Y+2>18 THEN 1100
1080 IF D(X,Y+2)<>0 THEN 1100
1090 DX=0:DY=2:T=T+1:R2=1
1100 IF X<2 THEN 1130
1110 IF D(X-2,Y)<>0 THEN 1130
1120 DX=-2:DY=0:T=T+1:R3=1
1130 IF Y<2 THEN 1160
1140 IF D(X,Y-2)<>0 THEN 1160
1150 DX=0:DY=-2:T=T+1:R4=1
1160 IF T=0 THEN 1250
1170 IF T<>1 THEN 1330

1180 REM *** 1 AANSLUITEND HOKJE VRIJ
1190 D(X+0.5*DX,Y+0.5*DY)=1:XQ=X+
0.5*DX+2:YQ=Y+0.5*DY+1
1200 FILL XQ*10,YQ*10 (XQ*10)+10,
(YQ*10)+10 20
1210 XG=(X+2)*10:YG=(Y+1)*10
1220 TC=23:G$=CHR$(98):GOSUB 900:X=X+
DX:Y=Y+DY
1230 XG=(X+2)*10:YG=(Y+1)*10
1240 TC=22:G$=CHR$(97):GOSUB 900:D(X,
Y)=-10*DX-DY:GOTO 1040

```

```

1250 REM *** GEEN AANSLUITEND HOKJE
VRIJ DOE STAP TERUG
1260 FILL (X+2)*10,(Y+1)*10 ((X+
2)*10)+10,((Y+1)*10)+10 20
1270 IF D(X,Y)=9 THEN 1410
1280 BK=D(X,Y):IF ABS(BK)=2 THEN 1300
1290 X=X+BK/10:GOTO 1310
1300 Y=Y+BK
1310 FILL (X+2)*10,(Y+1)*10 ((X+
2)*10)+10,((Y+1)*10)+10 20
1320 GOTO 1040

1330 REM *** >1 AANSLUITEND HOKJE
VRIJ-KIES ER EEN UIT
1340 DR=INT(4*RND(1))+1
1350 ON DR GOTO 1360,1380,1390,1370
1360 IF R1=1 THEN DX=2:DY=0:GOTO 1190
1370 IF R4=1 THEN DX=0:DY=-2:GOTO
1190
1380 IF R2=1 THEN DX=0:DY=2:GOTO 1190
1390 IF R3=1 THEN DX=-2:DY=0:GOTO
1190
1400 GOTO 1340

1410 REM *** INGANG EN UITGANG
1420 COLORG 8 0 1 14:FILL 0,0 330,9
21
1430 Y=(INT(18*RND(1))+1)*10:IF
SCRN(25,Y+5)=0 THEN 1430
1440 FILL 0,201 330,210 21:FILL 10,Y
20,Y+10 20

1450 Y=(INT(18*RND(1))+1)*10:IF
SCRN(305,Y+5)=0 THEN 1450
1460 FILL 310,Y 320,Y+10 20

1500 REM *** HINDERNISSEN
1510 FOR I=1 TO 10
1520 1 X=INT(28*RND(1))+1:X=(X*10)+20:
1 Y=INT(18*RND(1))+1
1530 1 Y=(Y*10)+10:IF SCRN(X+5,Y+5)=0
1 THEN 1520
1540 FILL X,Y X+10,Y+10 22:NEXT:LH=0:
SH=0

2000 REM *** ZOEK UW WEG
-----
2010 SEC=0:SF=0:WAIT TIME 2:POKE
#1BF,#FF:POKE #1BE,#FF
2020 PRINT CHR$(12);:LIM=300:XM=0:YM=
100:P=0:GR=8:BC=0
2030 GR=SCRN(XM+5,YM+2):XG=XM:YG=YM:
TC=23:G$=CHR$(99)
2040 GOSUB 900:XH=XM:YH=YM:T$="":IF
GR=1 THEN GOSUB 400
2050 SEC=
(#FFFF-PEEK(#1BE)-(PEEK(#1BF)*25
6))/50:EF=SEC+SF
2060 IF LH<>LIM THEN CURSOR 50,3:
PRINT "LIM=";LIM;" ";
2070 IF SH<>EF THEN CURSOR 50,2:
PRINT "SEC=";EF;" ";

```

```

2080 IF HR=1 THEN 3140
2090 IF SEC+SF>LIM THEN 3100
2100 G=GETC:IF G=0 THEN 2050
2110 IF G=16 AND SCRNXM+5,YM+11<>0
THEN YM=YM+10
2120 IF G=17 AND SCRNXM+5,YM-1<>0
THEN YM=YM-10
2130 IF XM=0 THEN 2150
2140 IF G=18 AND SCRNXM-5,YM+5<>0
THEN XM=XM-10
2150 IF G=19 AND SCRNXM+15,YM+5<>0
THEN XM=XM+10
2160 IF XM=20 AND P=0 THEN XD=9:YD=
YM:GOSUB 300:GOTO 2030
2170 IF XM=320 THEN 3000
2180 FILL XH+2,YH+1 XH+8,YH+9 GR
2190 BC=0:LH=LIM:SH=SEC+SF:GOTO 2030

3000 REM *** U BENT GESLAAGD
-----
3010 G$=CHR$(99):SEC=
(#FFFF-PEEK(#1BE)-(PEEK(#1BF)*25
6))/50
3020 PRINT CHR$(12);:POKE #75,95:TC=
23:XG=320:YG=YM:ST=0
3030 XD=310:YD=YM:GOSUB 300:GOSUB
900:G=GETC:G=GETC:G=GETC
3040 FOR I=1 TO 5:IF GETC<>0 THEN ST=

PAGE 05 -- DOOLHOF

1
3050 WAIT TIME 3:NEXT:IF ST=1 THEN 3200
3200
3060 G$=CHR$(104):GOSUB 900
3070 FOR I=1 TO 5:IF GETC<>0 THEN ST=
1
3080 WAIT TIME 3:NEXT:IF ST=1 THEN

3090 G$=CHR$(99):GOSUB 900:GOTO 3040

3100 REM *** U BENT VERHONGERD
-----
3110 XG=XM:YG=YM:TC=21:G$=CHR$(103):
GOSUB 900
3120 PRINT CHR$(12);"U BENT VAN
HONGER OMGEKOMEN !!"
3130 PRINT "HET SPEL IS VOOR U
DEFINITIEF AFGELOPEN !!"
3140 MEL$="KKKKjUUKK9K..KKKKjUUKK9K":
TP=15:GOTO 3300

3200 REM *** HET EINDE
-----
3210 PRINT CHR$(12);"U BENT VEILIG
DOOR DE DOOLHOF GERAAKT"
3220 PRINT "EN WEL IN";SEC+SF;"
SECONDEN !!"
3230 MEL$=
"KKUjjUK9))9KK9..KKUjjUK9))9K9))
":TP=10

```

```

3300 REM *** UW KEUZE
-----
3310 FOR I=0 TO LEN(MEL$)-1:T$=
MID$(MEL$,I,1)
3320 1 F=ASC(T$)+90:IF TG$="." THEN F=
1 20000
3330 SOUND 1 0 15 0 FREQ(F):WAIT
TIME TP:SOUND OFF :NEXT
3340 CURSOR 10,0:PRINT "WILT U
OPNIEUW SPELEN (J/N) ";
3350 GOSUB 200:IF G=74 OR G=106 THEN
100
3360 POKE #75,95:PRINT CHR$(12);
TAB(22);"E I N D E"
3370 GOTO 5000

4000 REM *** TITELPAGINA
-----
4010 CLEAR 5000:DIM CL(3),D(28,18):
POKE #75,32
4020 XG=0:YG=0:G$="":TC=23:TH=1:HS=
11:VS=11:BC=0:BF=0
4030 ENVELOPE 0 15:PRINT CHR$(12);
4040 COLORG 8 0 1 14:MODE 6A:COLORT
8 0 8 9
4050 FOR X=10 TO 42:CURSOR X,1:POKE
#76,#FF:NEXT
4060 POKE #AA7C,#FF:POKE #A374,#FE:
POKE #9C6C,#FD
4070 POKE #9564,#FC:POKE #8E5C,#FB:
POKE #8754,#FA
4080 FOR I=0 TO 4:TC=21:TK=21:IF I=4
THEN TC=23:TK=22
4090 1 DRAW 40+I,40+I 210+I,40+I TK
4100 1 DRAW 210+I,40+I 280+I,160+I TK
4110 1 DRAW 280+I,160+I 100+I,160+I TK
4120 1 DRAW 100+I,160+I 40+I,40+I TK
4130 1 XG=100+I:YG=130+I:G$="DWARS":
1 GOSUB 900
4140 1 XG=90+I:YG=110+I:G$="DOOR":
1 GOSUB 900
4150 1 XG=80+I:YG=90+I:G$="DE":GOSUB
1 900
4160 1 XG=70+I:YG=70+I:G$="DOLLE":
1 GOSUB 900
4170 XG=60+I:YG=50+I:G$="DOOLHOF":
GOSUB 900:NEXT

4180 POKE #7556,#DE:POKE #74D0,#DF:
POKE #744A,#D9
4190 CURSOR 43,3:PRINT "GESCHREVEN
DOOR : "
4200 CURSOR 51,2:PRINT "DE BONT C"
4210 CURSOR 10,1:PRINT "DRUK OP EEN
TOETS OM TE BEGINNEN";
4220 TH=0:BF=1:GOSUB 200:MODE 2A

```

```

4500 REM *** INFO
-----
4510 MODE 0:PRINT CHR$(12);:K$="
* ":COLORT 8 0 8 14
4520 FOR X=5 TO 52:FOR Y=0 TO 23
STEP 23:CORSOR X,Y
4530 POKE #76,#FF:PRINT "*";:NEXT:
NEXT
4540 FOR X=5 TO 52 STEP 47:FOR Y=1
TO 22:CORSOR X,Y
4550 POKE #76,#FF:PRINT "*";:NEXT:
NEXT:CORSOR 0,22
4560 POKE #BF68,#D9:POKE #BEE2,#DA:
POKE #BDD6,#D0
4570 POKE #BAB2,#D7:POKE #B9A6,#DD:
POKE #B89A,#DE
4580 POKE #B708,#DF:POKE #B5FC,#D0:
POKE #B46A,#D6
4590 PRINT K$;"(>>>***====--DE DOLLE
DOOLHOF-----***<<<<)"
4600 PRINT K$;"(>>>>*****=====
-----*****<<<<<<)"
4610 PRINT K$
4620 PRINT K$;"DE BEDOELING VAN DIT
SPEL IS DAT U ZO SNEL"
4630 PRINT K$;"MOGELIJK DOORHEEN DE
DOOLHOF LOOPT! U MOET"
4640 PRINT K$;"SNEL ZIJN,WANT U
LOOPT HET RISICO VAN HON-"
4650 PRINT K$;"GER OM TE KOMEN !! U
KUNT BEWEGEN MET UW 4"
4660 PRINT K$;"CURSORTOETSEN. LET
WEL OP,ER ZIJN DIVERSE"
4670 PRINT K$;"HINDERNISSEN OP UW
WEG,EN WEL : "
4680 PRINT K$;"1) VOEDSELPAKETTEN:
ALS U HIER AANKOMT WORD"
4690 PRINT K$;" UW LOOPTIJD MET
100 SECONDEN VERLENGD!!"
4700 PRINT K$;"2) VALKUILEN:HIER
KUNT U TOT 100 SECONDEN"
4710 PRINT K$;" TIJD VERLIEZEN !!"
4720 PRINT K$;"3) SPRINGPLANKEN:DEZE
ZULLEN U WEGSLINGE-"
4730 PRINT K$;" REN,ZODAT U OP EEN
WILLEKEURIGE PLAATS"
4740 PRINT K$;" IN DE DOOLHOF KAN
TERECHT KOMEN !!"
4750 PRINT K$;"4) SLANGEKUIL:DE KANS
HIEROP IS ERG KLEIN,"

```

```

4760 PRINT K$;" MAAR DEZE KUIL IS
STEEDS DODELIJK !!"
4770 PRINT K$;"U KUNT NOOIT ZIEN
WELKE HINDERNISSEN OP UW"
4780 PRINT K$;"WEG LIGGEN, EN U ZULT
ZE OOK NOOIT KUNNEN"
4790 PRINT K$;"VERMIJDEN !!":COLORG
8 0 3 14
4800 PRINT K$;" WILT U DE DOOLHOF
ZIEN GROEIEN (J/N)";
4810 GOSUB 200:IF G=78 OR G=110 THEN
COLORG 0 0 0 14
4820 PRINT CHR$(12);:MODE 6A:K$="
":GOTO 1000
5000 REM ----- EINDE
-----

```

NEW SOFTWARE

Disk Based Dbasic

- For a description of the software see DAINAMIC newsletter 29
- This software package is yet available on the following floppy-drive systems :
 - Indata 2 * 80k drives with DOS 1.0 or CP/M slavedos eproms.
 - Indata 2 * 160k drives with MDOS3 slavedos eproms.
 - Indata 2 * 320k drives with MDOS3 slavedos eproms.
- The package contains :
 - a bootstrap.
 - the DBASIC V2.2 programming language interpreter.
 - the configuration file (CONFIG.CFG)
 - the file handler (DBFIL.SYS)
 - the basic dos (DBDOS.SYS)
 - one or more BIOS's (BIOSxxx.SYS)
 - the DCR driving extension (DCR.SYS)
 - the cross-reference utility (XREF.SYS)
 - the function keys extension (KEY.SYS)
 - the cassette driving extension (CAS.SYS)
 - a file copy utility (SYS0.SYS)
- Three manuals are included in the package.
- The END-USER price is **4000 Bfr.**
- Upgrade from DCR or CASSETTE version : **2000 Bfr.**
- BIOS's for other drive systems are in development

Chacun connaît des hauts et des bas.

Certains jours, tout va bien et d'autres, tout va mal.

Depuis longtemps, on suppose que ceci répond à des lois mathématiques.

Le docteur Wilhem FLIESS (1858-1928) et le docteur Hermann SWOBODA (1873-1963) ont posés les bases de la théorie du BIORYTHME.

Selon ces lois, notre vie serait régie par trois cycles:

- le cycle physique ou masculin qui a une période de 23 jours
- le cycle émotionnel ou féminin qui a une période de 28 jours
- le cycle intellectuel qui a une période de 33 jours.

Ces cycles sont sinusoidaux et définissent des alternances positives et négatives.

Les jours où les courbes passent par zéro sont critiques.

L'amplitude des courbes indique les influences positives ou négatives.

Les trois cycles débutent le jour de la naissance.

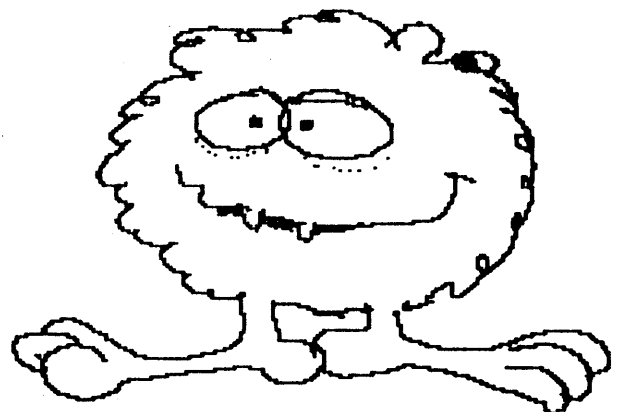
Pour établir un biorythme, il suffira de tracer ces trois sinusoides pour la période désirée.

Pour cela, il faut calculer le nombre de jours qui se sont écoulés depuis la naissance.

Le programme proposé est très complet. Il affiche au MENU:

- JOURS DE LA SEMAINE: connaissant une date, le programme calcule quel est le jour qui y correspond.
- ECART ENTRE DEUX DATES: connaissant deux dates, le programme calcule combien de jours séparent ces dates.
- SIGNES DU ZODIAQUE: en fonction de la date de la naissance, le programme calcule quel est le signe du Zodiaque qui y correspond.
- BIORYTHME: ce programme exploite les trois premiers afin d'établir les sinusoides du BIORYTHME. Les trois courbes apparaissent successivement sur l'écran.

Le programme est conversationnel




```

10  MODE 0:PRINT CHR$(12):COLORT 9 0 0 0:RESTORE:PRINT
20  PRINT "  MENU:";PRINT
30  PRINT "    JOURS DE LA SEMAINE                1"
40  PRINT "    ECART ENTRE DEUX DATES                2"
50  PRINT "    SIGNES DU ZODIAQUE                        3"
60  PRINT "    BIORYTHME                                4"
70  INPUT "      choisissez votre programme:";R:PRINT :PRINT
80  ON R GOTO 200,300,500,600
90  PRINT "CE PROGRAMME N'EXISTE PAS.RECOMMENCEZ.":PRINT :GOTO 70
200 PRINT CHR$(12):PRINT
210 PRINT "JOURS DE LA SEMAINE.":PRINT :PRINT
220 PRINT "  DONNEZ LA DATE: jour mois annee":PRINT
230 PRINT "EXEMPLE:9 (RETURN) 4 (RETURN) 83 (RETURN)"
240 PRINT " pour la date du 9 avril 1983.":PRINT
250 INPUT J%:INPUT M%:INPUT A%
260 GOSUB 2000
270 PRINT :PRINT "          CE JOUR EST UN ";J$(S%)
280 GOTO 2700
300 PRINT CHR$(12):PRINT
310 PRINT "ECART ENTRE DEUX DATES.":PRINT
320 PRINT "DONNEZ LA PREMIERE DATE jour mois annee":PRINT
330 PRINT "EXEMPLE: 9 (RETURN) 4 (RETURN) 83 (RETURN)"
340 PRINT " pour la date du 9 avril 1983.":PRINT
350 INPUT J%:INPUT M%:INPUT A%
360 GOSUB 2020:PRINT
370 C1%=C%
380 PRINT "DONNEZ LA DEUXIEME DATE jour mois annee":PRINT
390 INPUT J%:INPUT M%:INPUT A%
400 GOSUB 2020:PRINT :PRINT
410 PRINT "          L'ECART EST DE ";C%-C1%," JOURS."
420 GOTO 2700
500 PRINT CHR$(12):RESTORE:PRINT
510 PRINT "SIGNES DU ZODIAQUE.":PRINT
520 PRINT "DONNEZ LE JOUR ET LE MOIS DE VOTRE NAISSANCE.":PRINT
530 PRINT "PAR EXEMPLE:9 (RETURN) 4 (RETURN)"
540 PRINT "pour le 9 avril."
550 INPUT J%:INPUT M%
560 GOSUB 2500
570 PRINT :PRINT "  VOTRE SIGNE DU ZODIAQUE: ";Z$(I%)
580 GOTO 2700
600 PRINT CHR$(12):COLORT 12 0 12 12
610 POKE #BF69,#5A
620 CURSOR 3,22
630 PRINT "BIORYTHME":PRINT
640 FOR X=#BF67 TO #BF5D STEP -2.0
650 POKE X,42
660 POKE X-30,42
670 NEXT
680 POKE #BESC,#CF
690 PRINT
690 INPUT "QUEL EST VOTRE PRENOM:";P$
695 PRINT :PRINT
700 PRINT "QUELLE EST VOTRE DATE DE NAISSANCE:"
710 PRINT "  par exemple:9(RETURN) 4(RETURN) 83(RETURN)"
720 PRINT "  (pour 9 avril 1983)"
730 INPUT J%:INPUT M%:INPUT A%
731 GOSUB 2000:C1%=C%
735 PRINT :PRINT
740 PRINT "POUR QUEL MOIS DESIREZ-VOUS VOTRE BIORYTHME:"
750 PRINT "  par exemple 7(RETURN) 82(RETURN)"
760 PRINT "  pour juillet 1982"
770 INPUT MB%:INPUT AB%
780 WAIT TIME 50
790 PRINT CHR$(12):COLORT 12 0 12 12
800 POKE #BF69,#5A

```

```

810 CURSOR 2,22
820 PRINT "BIORYTHME DE":PRINT
830 POKE #BE5D,#5A
840 CURSOR 4,20:PRINT P$:PRINT
850 POKE #BD50,#CF
855 PRINT
860 GOSUB 2000
870 PRINT TAB(6);"VOUS ETES NE UN ";J$(S%)
880 PRINT
890 POKE #BBBE,#CE
900 GOSUB 2500
905 PRINT
910 PRINT TAB(6);"VOTRE SIGNE DU ZODIAQUE EST: ";Z$(I%)
920 POKE #BA2C,#CD
925 PRINT :PRINT
930 PRINT "(donnez une impulsion SPACE ";
931 PRINT "pour la suite du programme.)"
935 IF GETC(>)32 THEN 935
936 J%=1:M%=ME%:A%=AB%:GOSUB 2000
937 X=((C%-C1%)*8.0)-31.0
940 PRINT CHR$(12):MODE 6A:COLORT 14 0 0 0:COLORG 14 0 8 3
950 FOR N=1.0 TO 31.0
960 C=8.0
970 IF N=1.0 OR N=5.0 OR N=10.0 OR N=15.0 OR N=20.0 OR N=25.0 OR N=3
0.0 THEN C=0.0
980 DRAW (24+N*8),0 (24+N*8),230 C
990 NEXT
1000 DRAW (24+8),110 272,110 8
1010 CURSOR 0,3
1020 PRINT " 1 5 10 15 20 25 30"
1040 CURSOR 0,2:PRINT "PHYSIQUE . . . . ."
1050 FOR X1=31.0 TO 272.0 STEP 2.0:X2=X+X1
1060 Y=80.0*SIN(2.0*PI*X2/184.0)+110.0
1070 DOT X1,Y 0
1080 NEXT
1090 CURSOR 35,2:PRINT "(appuyez sur SPACE)"
1100 IF GETC(>)32 THEN 1100
1105 CURSOR 35,2:PRINT " "
1110 CURSOR 5,1:PRINT "EMOTIONNEL . . . . .";
1120 FOR X1=31.0 TO 272.0:X2=X+X1
1130 Y=80.0*SIN(2.0*PI*X2/224.0)+110.0
1140 DOT X1,Y 0
1150 NEXT
1160 CURSOR 35,2:PRINT "(appuyez sur SPACE)"
1170 IF GETC(>)32.0 THEN 1170
1180 CURSOR 35,2:PRINT " "
1190 CURSOR 10,0:PRINT "INTELLECTUEL-----";
1200 FOR X1=31.0 TO 272.0 STEP 0.5:X2=X+X1
1210 Y=80.0*SIN(2.0*PI*X2/264.0)+110.0
1220 DOT X1,Y 0
1230 NEXT
1240 CURSOR 33,2:PRINT "(appuyez sur SPACE"
1250 CURSOR 33,1:PRINT "pour revenir au MENU)"
1260 IF GETC(>)32.0 THEN 1260
1270 GOTO 10
1999 END
2000 DIM J$(7.0)
2010 J$(1.0)="LUNDI"
2011 J$(2.0)="MARDI"
2012 J$(3.0)="MERCREDI"
2013 J$(4.0)="JEUDI"
2014 J$(5.0)="VENDREDI"
2015 J$(6.0)="SAMEDI"
2016 J$(7.0)="DIMANCHE"
2020 A1%=INT(A%/100.0)
2030 A2%=A%-100*A1%

```

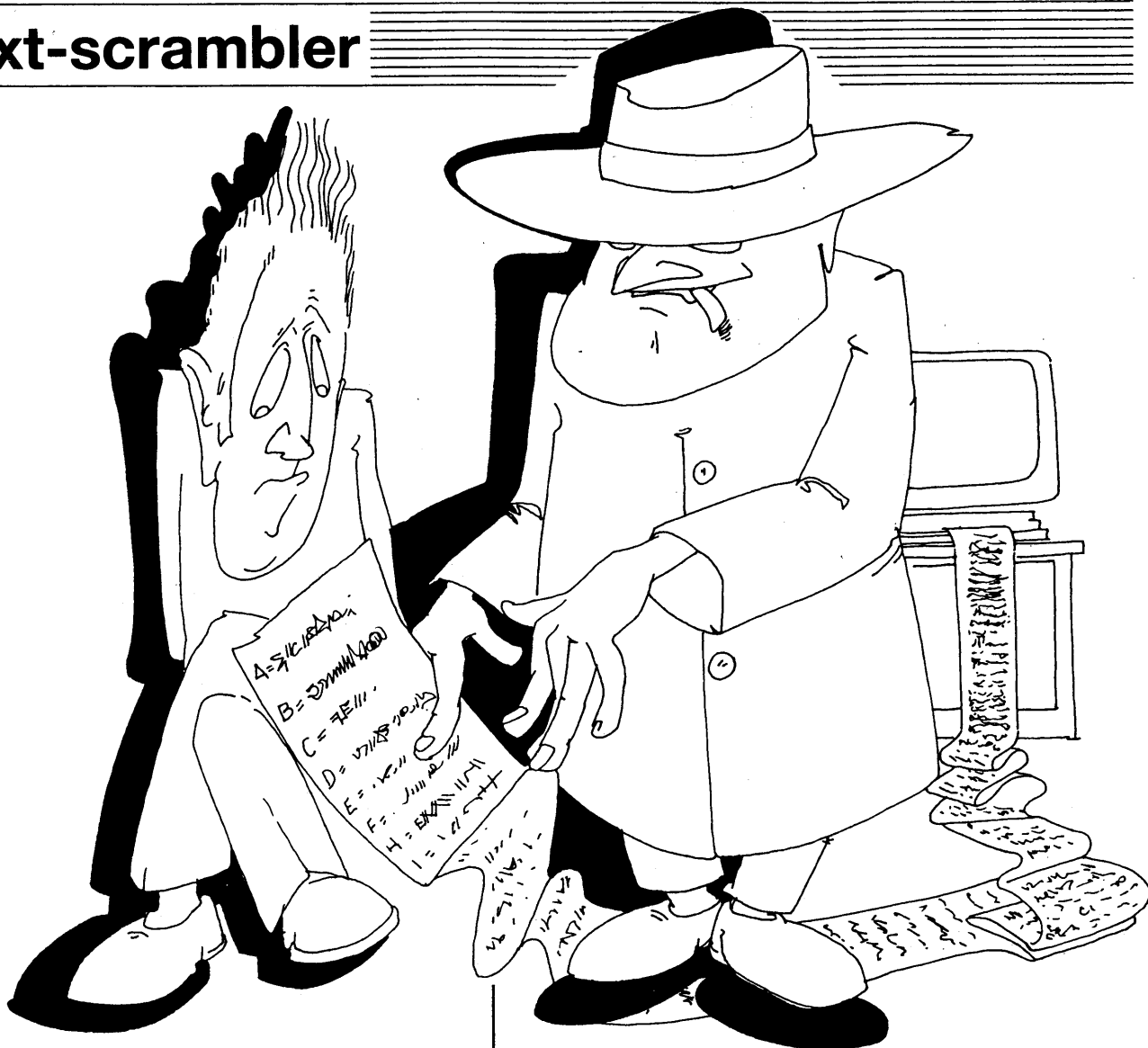
```

2040 N%=0
2050 IF M%>2 THEN 2140
2060 N%=2
2070 IF A2%=0 THEN 2120
2080 R%=A2%-4*INT(A2%/4.0)
2090 IF R%<>0.0 THEN 2140
2100 N%=1
2110 GOTO 2140
2120 R%=A1%-4*INT(A1%/4.0)
2130 IF R%=0.0 THEN N%=1
2140 C%=INT(365.25*A2%)+INT(30.56*M%)+N%+J%
2150 S%=3+C%-7*INT((C%+2.0)/7.0)
2160 RETURN
2500 DIM Z$(12.0)
2510 FOR I%=1 TO 12
2520 READ Z$(I%)
2530 NEXT
2560 I%=M%:L%=20
2570 ON M% GOTO 2610,2610,2600,2610,2600,2600,2590,2580,2590,2580,260
0,2580
2580 L%=L%+1
2590 L%=L%+1
2600 L%=L%+1
2610 IF J%<L% THEN 2630
2620 I%=I%+1
2630 IF I%<=12 THEN 2650
2640 I%=1
2650 RETURN
2700 CURSOR 0,5
2710 PRINT "donnez une impulsion SPACE pour revenir au MENU."
2720 IF GETC<>32.0 THEN GOTO 2720
2730 GOTO 10
10000 DATA "CAPRICORNE","VERSEAU","POISSON","BELIER"
10010 DATA "TAUREAU","GEMEAUX","CANCER","LION"
10020 DATA "VIERGE","BALANCE","SCORPION","SAGITTAIRE"

```



Text-scrambler



PAGE 01 -- TEXT-SCRAMBLER

```

100  REM *** TEXT SCRAMBLER-DECODER
    *****
110  REM *** WRITTEN BY : DE BONT
    CORNEEL / 18-2-1985 *****
120  REM
    *****
    *****
130  GOTO 1000

200  REM *** [S] PLACE STRING IN
    ARRAY *****
210  WW$(AR)=LINE$:AR=AR+1:LINE$=""
220  IF WW$(AR)="" THEN WW$(AR)=" "
230  RETURN

300  REM *** [S] INPUT S$ (KEY
    STRING) *****
310  G=GETC:G=GETC:G=GETC
320  G=GETC:IF G=0 THEN WAIT TIME 3:
    GOTO 320
330  IF G=9 THEN S$=W$:CURSOR 9,1:
    PRINT S$;:RETURN
340  IF G=8 AND LEN(S$)>1 THEN 390
350  IF G=32 THEN G=95:GOTO 380

```

```

360  IF G=13 THEN RETURN
370  IF G<65 OR G>90 THEN 310
380  S$=S$+CHR$(G):PRINT CHR$(G);:
    GOTO 310
390  S$=LEFT$(S$,LEN(S$)-1):PRINT
    CHR$(G);:GOTO 310

400  REM *** [S] GETC-ROUTINE
    *****
410  G=GETC:G=GETC:G=GETC
420  SOUND 1 0 9 0 FREQ(RND(950)+50):
    WAIT TIME 3:SOUND OFF
430  CC=RND(16):POKE #B46A,#DO+CC:IF
    CC=8 THEN 430
440  G=GETC:IF G=0 THEN W=W+1:WAIT
    TIME 3:IF W<30 THEN 440
450  IF G=0 THEN W=0:GOTO 420
460  POKE #B46A,#DO:RETURN

500  REM *** [S] INPUT L$ (LINE
    STRING) *****
510  G=GETC:G=GETC:G=GETC
520  G=GETC:IF G<8 OR G>90 THEN WAIT
    TIME 3:GOTO 520
530  IF G=8 AND LEN(L$)>0 THEN 590
540  IF G=13 THEN RETURN

```

```

550 IF G=34 THEN 520
560 IF G<32 OR G>90 THEN 510
570 L$=L$+CHR$(G)
580 PRINT CHR$(G);:GOTO 510
590 L$=LEFT$(L$,LEN(L$)-1):GOTO 580

600 REM *** [S] SCRAMBLING
*****
610 FOR I=0 TO LEN(L$)-1:T$=MID$(L$,
I,1)

PAGE 02 -- TEXT-SCRAMBLER

620 1 U$=CHR$(ASC(T$)+WW(NR)):LINE$=
1 LINE$+U$
630 1 NR=NR+1:IF NR=42 THEN NR=0
640 NEXT:GOTO 200

700 REM *** [S] PAGINA-TITLE (A$,B$,
C$) *****
710 MODE 0:PRINT CHR$(12);:COLORT 8
0 8 14:POKE #BFEE,#DD
720 POKE #BF68,#DA:POKE #BEE2,#DO:
POKE #BE5C,#F2:NR=0
730 POKE #BDD6,#DA:POKE #BD50,#DO:
FOR Y=22 TO 23
740 1 FOR X=14 TO 22:CURSOR X,Y:POKE
1 #76,#FF:CURSOR X+24,Y
750 POKE #76,#FF:NEXT:NEXT:CURSOR
30,19:POKE #76,#FF
760 POKE #8A,81:POKE #8B,189:CURSOR
14,23:K$=" *- *- "
770 PRINT ">";K$;A$;K$;"<":PRINT
TAB(14);">";K$;
780 PRINT "*****";K$;"<":
PRINT TAB(14);B$
790 PRINT TAB(14);C$:PRINT TAB(14);
795 PRINT ">>>>***=====O-----"
****<<<<":RETURN

800 REM *** [S] DEMO PASSWORD
*****
810 W$=W$+CHR$(68)+CHR$(69)+
CHR$(95)+CHR$(66)+CHR$(79)
820 W$=W$+CHR$(78)+CHR$(84)+
CHR$(95)+CHR$(67)+CHR$(79)
830 W$=W$+CHR$(82)+CHR$(78)+
CHR$(69)+CHR$(69)+CHR$(76)
840 W$=W$+CHR$(95)+CHR$(79)+
CHR$(79)+CHR$(83)+CHR$(84)
850 W$=W$+CHR$(69)+CHR$(73)+
CHR$(78)+CHR$(68)+CHR$(69)
860 W$=W$+CHR$(95)+CHR$(66)+
CHR$(65)+CHR$(65)+CHR$(82)
870 W$=W$+CHR$(76)+CHR$(69)+
CHR$(95)+CHR$(72)+CHR$(69)
880 W$=W$+CHR$(82)+CHR$(84)+
CHR$(79)+CHR$(71)+CHR$(95)
890 RETURN

900 REM *** [S] YELLOW SQUARE
*****
910 CURSOR X,Y:POKE #76,#FF:PRINT
L$;:RETURN

```

```

1000 REM *** INIT
*****
1010 CLEAR 20000:MODE 0:PRINT
CHR$(12);:COLORT 8 0 8 14
1020 DIM WW(42),WW$(255):GOSUB 800
1030 PW=0:ST=0:GOTO 4110

PAGE 03 -- TEXT-SCRAMBLER

1100 REM *** INFO
*****
1110 MODE 0:PRINT CHR$(12);:POKE
#BF68,#DD:L$=CHR$(1)
1120 FOR X=2 TO 58:Y=23:GOSUB 900:Y=
0:GOSUB 900:NEXT
1130 L$=CHR$(127):FOR Y=1 TO 23:X=2:
GOSUB 900:X=58
1140 GOSUB 900:NEXT:K$=" "+L$+"
":POKE #BEE2,#DB
1150 POKE #BE5C,#DO:POKE #BB38,#D6:
POKE #B9A6,#D7
1160 POKE #B814,#D9:POKE #B708,#DA:
POKE #B682,#DB
1170 POKE #B5FC,#DD:POKE #B4FO,#DF:
POKE #B46A,#DO:L$=" "
1180 FOR X=9 TO 10:FOR Y=4 TO 14:
GOSUB 900:NEXT:NEXT
1190 CURSOR 16,22:PRINT "TEXT
SCRAMBLER-DECODER"
1200 CURSOR 16,21:PRINT
"*****"
1210 PRINT K$;"THIS PROGRAM ALLOWS
YOU TO SCRAMBLE OR DE-"
1220 PRINT K$;"CODE ANY KIND OF
TEXT, BY USE OF A SECRET"
1230 PRINT K$;"KEYWORD KNOWN ONLY
BY YOU !! THE KEYWORD"
1240 PRINT K$;"YOU WILL GIVE ME
CAN BE ANY WORD OR LINE"
1250 PRINT K$;"AS LONG AS YOU USE
ONLY CAPITAL CHARAKTERS"
1260 PRINT K$;"AND SPACES (NO
NUMBERS OR SYMBOLS)"
1270 PRINT K$;" 1) THIS COMPUTER
WILL SCRAMBLE ANY TEXT"
1280 PRINT K$;" WICH YOU HAVE
TYPED IN !! THE SCRAMBLED"
1290 PRINT K$;" LINES WILL BE
STORED AWAY IN AN ARRAY !"
1300 PRINT K$;" 2) THIS COMPUTER
WILL DECODE ANY TEXT IT"
1310 PRINT K$;" CAN FIND IN HIS
ARRAY. THE DECODED TEXT"
1320 PRINT K$;" WILL BE SHOWN ON
YOUR SCREEN ONLY !"
1330 PRINT K$;" 3) YOU CHOOSE YOUR
RECORD-MEDIUM:"
1340 PRINT K$;" AUDIO-TAPE OR
DCR-TAPE"

```

```

1350 PRINT K$;" 4) YOU CAN SAVE YOUR
      ARRAY"
1360 PRINT K$;" 5) YOU CAN LOAD A
      NEW ARRAY"
1370 PRINT K$;" 6) YOU CAN STOP THE
      PROGRAM"
1380 PRINT K$;"" :IF PW=1 THEN 1450
1390 PRINT K$;" FIRST GIVE ME THE
      KEY-WORD (42 CHARAKTERS)"

```

PAGE 04 -- TEXT-SCRAMBLER

```

1400 PRINT K$;"
      -----
      -----"
1410 CURSOR 9,1:S$="":GOSUB 300:PW=1
1420 IF LEN(S$)>42 THEN S$=LEFT$(S$,
      42)
1430 IF LEN(S$)<42 THEN S$=S$+
      CHR$(95):GOTO 1430
1440 FOR I=0 TO LEN(S$)-1:WW(I)=
      ASC(MID$(S$,I,1))-60:NEXT
1450 CURSOR 9,1:PRINT S$+" ";:CURSOR
      9,2
1460 PRINT "MAKE YOUR CHOICE:GIVE A
      NUMBER BETWEEN 1-6";
1470 GOSUB 400:IF G<49 OR G>54 THEN
      1470
1480 ON G-48 GOTO 2000,3000,4000,
      5000,6000,7000
1490 GOTO 1470

2000 REM *** SCRAMBLE YOUR TEXT
      *****
2010 B$="PLEASE GIVE ME YOUR
      TEXTLINES NOW"
2020 C$="WHEN YOU SAY 'STOP',I
      WILL STOP"
2030 A$="SCRAMBLE TEXT":GOSUB 700:AR=
      0
2040 L$="":GOSUB 500:PRINT :POKE
      #BD50,#D4
2050 IF L$="" THEN L$=" "
2060 GOSUB 600:POKE #BD50,#D0
2070 IF L$="STOP" OR AR=256 THEN 1100
2080 GOTO 2040

3000 REM *** DECODE YOUR TEXT
      *****
3010 B$="I WILL DECODE EACH DATALINE
      I CAN"
3020 C$="FIND,WHEN YOU PRESS YOUR
      SPACEBAR"
3030 A$="DECODE A TEXT":GOSUB 700:AR=
      0
3040 L$=WW$(AR):IF L$="STOP" OR L$=
      "" THEN 1100
3050 FOR I=0 TO LEN(L$)-1:T$=MID$(L$,
      I,1)
3060 1 PRINT CHR$(ASC(T$)-WW(NR));:NR=
      NR+1:IF NR=42 THEN NR=0
3070 NEXT:PRINT :GOSUB 400:AR=AR+1:
      GOTO 3040

```

```

4000 REM *** CHOOSE BETWEEN AUDIO OR
      DCR *****
4010 B$="IF YOU HAVE TWO RECORDERS,
      YOU CAN"
4020 C$="CHOOSE ONE OF THEM : DCR OR
      AUDIO"
4030 A$="CHOOSE A TAPE":GOSUB 700
4040 CURSOR 10,16:PRINT "AUDIO OR

```

PAGE 05 -- TEXT-SCRAMBLER

```

      DCR (A/D) ";
4050 GOSUB 400:IF G=65 THEN 4080
4060 IF G=68 THEN ST=1:GOTO 4110
4070 GOTO 4050
4080 M1$="AUDIO":M2$="SET RECORD,
      START TAPE"
4090 CALLM #FOOO:REM CAS
4100 GOTO 4140
4110 M1$="DCR":M2$="INSERT
      WRITE-ENABLED CASSETTE"
4120 CALLM #FOOO:REM DCR
4130 IF ST=0 THEN 1100
4140 CURSOR 10,14:PRINT "YOU CHOSE
      FOR ";M1$
4150 CURSOR 10,12:PRINT "PRESS A KEY
      FOR MENU"
4160 GOSUB 400:GOTO 1100

5000 REM *** SAVE YOUR ARRAY
      *****
5010 B$="IF YOU WANT,I WILL SAVE
      YOUR TEXT"
5020 C$="IN ARRAY ON "+M1$+"-TAPE !!"
5030 A$="SAVE AN ARRAY":GOSUB 700
5040 CURSOR 10,16:PRINT "DO YOU WANT
      TO SAVE (Y/N) ";
5050 GOSUB 400:IF G=78 THEN 1100
5060 CURSOR 10,14:INPUT "GIVE ME A
      NAME ";NAME$:PRINT
5070 NAME$=" "+NAME$:CURSOR 10,12:
      PRINT M2$;":TYPE SPACE"
5080 GOSUB 400:IF G<>32 THEN 5080
5090 SAVEA WW$ NAME$
5100 GOTO 1100

6000 REM *** LOAD A NEW ARRAY
      *****
6010 B$="IF YOU WANT,I WILL LOAD
      YOUR TEXT"
6020 C$="IN ARRAY FROM "+M1$+"-TAPE
      !!"
6030 A$="LOAD AN ARRAY":GOSUB 700
6040 CURSOR 10,16:PRINT "DO YOU WANT
      TO LOAD (Y/N) ";
6050 GOSUB 400:IF G=78 THEN 1100
6060 CURSOR 10,14:INPUT "GIVE A NAME
      ";NAME$:PRINT
6070 IF NAME$="" THEN 6100
6080 LOADA WW$ NAME$
6090 GOTO 1100
6100 LOADA WW$
6110 GOTO 1100

```

```

7000 REM *** STOP THE PROGRAM
*****
7010 B$="THE COMPUTER WILL END THE
PROGRAM"
7020 C$="AND YOUR ARRAY WILL BE
WIPED OUT!"
7030 A$="STOP PROGRAM!":GOSUB 700

PAGE 06 -- TEXT-SCRAMBLER

7040 CURSOR 10,16:PRINT "DO YOU WANT
TO STOP (Y/N) ";
7050 GOSUB 400:IF G=78 THEN 1100
7060 CURSOR 10,14:PRINT "T H E E N
D"

```

KENDOS commando's

Hierbij vindt U een zeer korte routine om vanuit BASIC alle commando's voor MDCR of KENDO via variabelen uit te voeren. (i.p.v. de slecht gekozen CALLM HF003, waarbij telkens de commando's op het scherm geprint worden)
De routine kan gelijk waar in het geheugen geplaatst worden.

```

UT : S3000 F5 C5 D5 E5 4E 23 46 C3 33 F0
      PUSH ALL                JMP F033

```

```

BASIC : A$= DISK OF DCR COMMANDO + CHR$(13)
        CALLM H3000,A$

```

VB:

```

10 FOR B = 0 TO 9: READ A : POKE H3000+B,A:NEXT
20 PRINT:INPUT "GEEF STARTADRES";ST$ (hexadecimaal)
30 PRINT:INPUT "GEEF EINDADRES ";EN$
40 PRINT:INPUT "GEEF FILENAAM";NAAM$
50 PRINT:D$="DSAVE"+CHR$(34)+NAAM$+" "+ST$+" "+EN$
  +CHR$(34)+CHR$(13)
60 CALLM H3000,D$ : GOTO 20
70 DATA 245,197,213,229,78,35,70,195,51,240

```

VB : MDCR

```

10 FOR B = 0 TO 9: POKE H3000+B,A:NEXT
20 FOR C=0 TO 9:D$="SKIP"+CHR$(48+C)+CHR$(13)
30 PRINT "DCR SPOELT NU "; C ; "PROGRAMMA'S VERDER"
40 CALLM H3000,D$
50 NEXT
60 DATA 245,197,213,229,78,35,70,195,51,240

```

Digitizer results



Digitizer results



POWER-ON INITIALISATION

(from DAInamic 5 p123)

This summary covers the initialisation of the DAI pc on switching on. All addresses are in hex-notation. The video RAM addresses are based on a 48K machine.

- Set the stack pointer to F900.
- Load FD06 with 30;ROM bank 0, cassette motors off.
- Load TICC interrupt mask register (FFF8) with 04; only external interrupts are allowed.
- Load the TICC command register with 0C and immediately afterwards with 0D
- TICC reset
- Select IN7 interrupt (20ms blanking pulse from TV logic)
- INTA signal from CPU will be accepted.
- Load addresses 01C0 and 01C1. These will be used as timers in the RST7 and RST6 restart procedures respectively.
- Load Timer 3 (FFFB) and Timer 4 (FFFC) of the TICC 5501.
- Load the addresses 0000-003F with the interrupt vector routines and the vector addresses. The latter are at 0062-0071.
- Check if the Maths Chip is present. In address 00D4 will be 00 or 7B, depending on whether the AMD9511 is absent or present. Addresses 00D1/D0 and 00D3/2 will be loaded with C7F2 and DDE0 respectively.
- Check the RAM capacity. This is done by means of the contents of addresses 1000, 2000, etc being read, inverted and loaded in again. Then those locations are read again to see what they contain. As long as RAM is available this procedure continues. Should there be no more (8K - 32K machines), or if the ROM is reached (48K), the sequence stops. Thus the highest RAM address for constructing the video-RAM is fixed.

Setting up the video RAM

- Each routine that has to use the video RAM makes use of the RST5 routine. That switches to ROM bank 2. The actual address will be specified by the data that follows the RST instruction:

```
eg:  xxx0  RST 5      There will be a jump to  
     xxx1  DATA 03   2E003.
```

To set up the video RAM RST5/00 will be used.

- Load 0081/80 with BFFF and 0083/82 with BFEF.
 - Set screen mode to mode 1 (009D=00).
 - Define cursor mode and ASCII value for cursor (0074,0075).
 - Specify the colours for the COLORT registers (007C-007F).
 - Select screen mode 0.
 - Set the colours for the COLORG registers (009E-00A1).
 - Load 00C5/C4 with CA01 and 00C7/C6 with CA25.
- Set screen mode at 10.
- Load B350 in 02A6/5.
 - Load 0084-0098 with addresses of the video RAM.
 - Load the colour registers BFF0-BFFF with data.
 - Now the whole video RAM is arranged, for an empty screen in mode 0. The line control bytes are made 7A, the colour bytes 40. The character data bytes are 20 (space) and their colour bytes 00.
 - In 0079/78 is stored the line control address of the first line (later, the current line) and in 007A the last usable address of this line.
 - Load the colour registers B350-B35F with data. They are used for the COLORT colours.
 - Select screen mode 0.

NOW THE VIDEO RAM IS BUILT-UP

- Load 0296 with 00.
- Select screen and RS232 as output (0131), and keyboard and screen as input (0135).
- Set the TICC communications rate register (FFF5) at 9600 Bauds and 1 stop bit.

Fill the video RAM with "DAI PERSONAL COMPUTER"

- For this RST/03 will be used. This routine is always used to put data in the video RAM.
- Jump to screen line 7, that is 6 carriage returns.
- On the 7th line from the top 'DAI PERSONAL COMPUTER' is placed in the video RAM. It is done in mode 0, in the highest resolution, with 14 spaces between 'PERSONAL' and 'COMPUTER'.
- Now line 7 is set for medium resolution (line control byte 5F). This gives the well-known larger letters, although 'COMPUTER' is now beyond the screen. Therefore between 'PERSONAL' and 'COMPUTER' a new line control byte is created (5F + 40). This results in 'COMPUTER' being placed neatly below the preceding words. The rest of the video RAM will now be reorganised.
- The video section is now ready.

Now various pointers and other memory locations must be set before there can be an entry from the keyboard.

- Initialise the sound generator (FC06). The volume will be set to 0 (FD04/05).
 - FF will be loaded into locations 01C2, 01D0, 01DE and 01EC.
 - The contents of ROM addresses D7A4-D7CA are transferred to RAM locations 02C5-02EB. This piece of program is used for cassette control. When a floppy disk is to be used this part will have to be changed (see the handbook).
 - The DCE bus (GIC) is initialised via FE01 and FE03; RST1/0C is used. There is a check to see if the DCE bus is active; at power-on reset this is assumed to be unnecessary.
 - The pointers for the BASIC program are loaded:
 - 029B/C : Start address of the HEAP
 - 029D/E : Size of HEAP
 - 029F/A0 : Start address of text buffer
 - 02A1/2 : Start address of symbol table
 - 02A3/4 : Beginning of video RAM.
 - Cassette port 1 is activated.
- The GETC routine is prepared:
- ROM pointer 02A7/8 is loaded with the start address of the table containing the character ASCII codes.
 - GETC is enabled by 00 in 02B9.
 - 02BE/F and 02C0/1 are loaded with 02BA. This is the address where the ASCII code of the first character will be stored.
 - 02C4 is loaded with FF (BREAK).
 - Locations 0275-028F are loaded with 00.

Now wait until a key is pressed

When this happens the GETC routine is performed. RST1/15 is used for it.

- As a result of the GETC routine the ASCII code for the key pressed is in the accumulator and in 02BA. 02B9 is again loaded with FF. 02BE and 02C0 are increased by 1 and 02C4 = 00.
- The colours of the COLORT registers are changed and the registers themselves altered via RST5/06.
- The screen is cleared and the cursor moved to BFE7 (top left).
- " BASIC V1.0 " is written on the first line by means of RST5/03. Then follows a carriage

return/.

Now the DAI PC can be made ready for BASIC.

- 00 is loaded into the following RAM locations: 0100, 0101, 0104, 0105, 0113, 0114, 0117, 0118, 0122. The stack-pointer is again set to F900; this value is also loaded in 0128/27.
- The TICC interrupt mask register (FFF8) is updated, first for interrupts from TIMER 4 and external interrupts, then for IN7 interrupts (TV page blanking signal).
- Through RST5/0C various CPU registers are adapted.
- Now the prompt (*) is placed at the beginning of the second line via RST5/03 and followed by the cursor.
- Then the program enters an endless loop in a section of the GETC routine. This loop can only be broken by interrupts.

INTERRUPTS

At this stage there are 2 possible interrupts (apart from the external ones which will be considered further, outside this article):

- Whenever timer 4 runs down RST6 is called.
- Via the TV page blanking signal (every 20ms) RST7 is used.
- RST7: - 01C0 is decremented. As long as the result is not = 0 there is a jump back to the main program. If it is zero then RST5/12 follows. This flashes the cursor each time RST7 is executed, writing alternately 5F or 20 (bar or space).
- RST6: - Timer 4 is reloaded. Whenever 01C0 after being decremented is not = 1 there is a jump back to the main program; if it is, the GETC routine is called and executed.

Now inputs can be made.

?CHR\$(12) - IN HARDWARE

(DAInamic 7 p209)

A design shortcoming in the DAI PC means that one must type in ?CHR\$(12) to clear the screen. This may be overcome in hardware with the following CMOS circuitry, so that the screen can be cleared by a single key operation, using an additional key. Extensions of the circuitry permit other functions to be called.

Fig 1 shows the wiring of the DAI keyboard in accordance with the matrix shown on page 34 of the Handbook. The horizontal lines from the adjacent 5501 are made active through software so that the software can read the vertical lines via the 5501. Imagine that the S-key is pressed; at the moment when pin 27 of IC93 (5501) goes low, pin 3 of IC90 and consequently pin 35 of IC93 go high. (See fig 4 for the whereabouts of ICs 89, 90 and 93 on the printed circuit board). In accordance with the code present at any specific instant on pins 33 to 39 inclusive of the 5501, the software decides which key is pressed. Other keys may be connected in parallel with the keyboard keys without affecting the operation of the keyboard.

In fig 2 the function RUN is realised with a single key. At reset the flip-flop (4013) is reset, pin 2 is high so the Johnson counter (4017) is also held reset. On the clock input (pin 14) of the 4017 are the pulses from IC89 pin 11, which are regularly generated by the software scanning the keyboard. The outputs of the analogue switches (4016) are connected in parallel with the four keyboard switches R, U, N and RETURN. Pressing the RUN key sets the flip-flop which in turn makes pin 15 of the 4017 low, allowing the 4017 to start counting. The next pulse on pin 14 makes pin 2 high and the first analogue switch closes. This is seen by the software via the 5501 and an R appears on the screen. The next pulse on pin 14 sets pin 2 low and pin 4 high, closing the second analogue switch and making a U appear on the screen, and so on... When pin 1 goes high the 4013 resets and blocks the 4017.

Fig 3 shows an extension of this circuit which caters for the functions ?CHR\$(12), RUN, LIST and EDIT. An example will illustrate the numbering used in the schematic: pin 3 of IC210 (AND) goes to pin 3 of IC215 (OR) and to pin 5 of IC204 (ANALOGUE SWITCH).

DISADVANTAGES :

- 1) Sometimes when the DAI is switched on the counter starts running and a SYNTAX ERROR is given.
- 2) A relatively large amount of hardware is needed (17 ICs for the fig 3 circuit). It would be interesting to see if a PROM could be used when even more functions are needed.
- 3) The keyboard must be in Upper Case mode.

The circuit is connected with ribbon-cable. One can carefully solder the wires on the underside of the printed-circuit board to the pins of IC89 and 90 and to the pins of the blue resistor-block on the right, next to the IC.

PROGRAMMING TECHNIQUES

(from DAInamic 18, page 285)

In mid-August I received a tape from Koert van Espen which I found very interesting. It was conveniently provided with test arrays to facilitate input adjustments and the program itself was recorded more than once. The accompanying letter explained that he had invented STARBUILDER himself. It appeared to be a simple but pleasing action game and indeed original. Possible problems had been foreseen and solutions found. I asked Koert if I could use his program as a basis for an article and he kindly agreed, knowing full well that it would be criticised. Here is his original program:-

```
10 REM Program by Koert Van Espen Date: 1983 08
20 GOSUB 10000
30 PRINT "Do you want to play - without red bricks (type '1')?"
40 PRINT " - with red bricks (type '2')?"
50 G!=GETC: IF G!>49.0 AND G!>50.0 GOTO 50
100 FOR B=1 TO 3
110 PRINT CHR$(12): MODE 2A
120 CURSOR 0,2: PRINT "SCORE: ";S: CURSOR 30,2: FOR W=1 TO 3-B: IF B<>3 THEN
PRINT CHR$(255);" ";
130 NEXT
140 DRAW 0,0 XMAX,0 3: DRAW 0,YMAX-14 XMAX,YMAX 3: DRAW 0,0 0,YMAX-14 3:
DRAW XMAX,0 XMAX,YMAX-14 3
150 X=XMAX/2: Y=YMAX/2
160 DOT X,Y 5
170 E=30: R=15: FOR K=1 TO G!-48
180 FOR O=1 TO E: XX=INT(RND(2.0*X-2.0)+1.0): YY=INT(RND(2.0*Y-16.0)+1.0): IF SCRN
(XX,YY)=15 OR SCRN (XX,YY)=3 THEN O=O-1: NEXT
190 DOT XX,YY R: NEXT: E=15: R=3: NEXT
200 ENVELOPE 0 15,200;0,200;: IF T=0 GOTO 1000
300 H=SCRN(X,Y): IF H=3 GOTO 500
310 IF H=15 THEN S=S+100: T=T+1:SOUND 0 0 15 0 4545
320 IF T=30 GOTO 900: DOT X,Y 3: CURSOR 8,2: PRINT S;" "; SOUND 0 0 15 1 36363
330 GG=GETC: IF GG=0 THEN 1100: G=GG: ON GG-15 GOTO 2000,3000,4000,5000
500 SOUND 0 1 15 0 64516: ENVELOPE 1 15,20;10,20;5,10;: WAIT TIME 100: SOUND OFF
510 G=0: GG=0: T=0: NEXT
520 PRINT "Again?(Y/N)"
530 A=GETC: IF A<>89 AND A<>78 THEN 530: IF A=78 THEN END
800 G=0: GG=0: T=0: S=0: GOTO 100
900 G=0: GG=0: T=0: S=S+5000: GOTO 110
1000 G=GETC: IF G=0 GOTO 1000
1100 S=S-1: ON G-15 GOTO 2000,3000,4000,5000
2000 Y=Y+1: GOTO 300
3000 Y=Y-1: GOTO 300
4000 X=X-1: GOTO 300
5000 X=X+1: GOTO 300
10000 PRINT CHR$(12): COLORT 7 0 0 0: COLORG 0 3 5 15: MODE 0
10010 CURSOR 19,18: PRINT "S T A R B U I L D E R "
10020 CURSOR 19,17: PRINT " - - - - - "
10030 CURSOR 19,15: PRINT " KOERT VAN ESPEN "
10040 PRINT: PRINT: PRINT: PRINT "DO YOU WANT INSRUCTIONS? (Y/N)"
10050 G!=GETC: IF G!>89.0 AND G!>78.0, THEN 10050: IF G!=78.0 THEN RETURN
```

```
10060 PRINT CHR$(12); PRINT "Try to catch the white stars as fast as possible,"; PRINT  
"but avoid anything coloured red,"  
10070 PRINT "Use the cursor keys to move your ship.":RETURN
```

As you can see, it is a fairly short program and if you type it in you will find that it almost always works well. The only fault in the working rarely appears (have you found it?). You will have noted from the listing that line numbering is in multiples of 10, an excellent custom. The program appears to have been carefully thought out and thoroughly rechecked before being sent in; something sadly lacking with many submissions. The professional programmer will put a service advice or something similar after the last line, even though the program itself gives instructions. As I have said before, I object to asking the user if he needs instructions when they are as short as here. The introduction/explanation is correctly placed at the end of the listing, with logical line numbers, but this sub-routine has two RETURNS and that is not structurally good. It could be done just as well with GOTOS. We only go from line 20 to line 10000 and always return to line 100. If later we want to include a CLEAR would that be trouble free? For that matter would the program be any better with CLEAR? A CLEAR 4 is no good but CLEAR 256 is all right. I consider lines 30, 40 and 50 belong to the explanations; there the choice would only be made once in the program. Regarding line 50, do you realise that:

```
50 G=GETC: IF G=49 OR G=50 GOTO 100: GOTO 50
```

is the same? But watch out, Koert's solution leaves a free line for you between 50 and 100 but the second method does not do so directly although it could be made to. In both examples extra time is required for the AND/OR; probably not so important here, but the keyboard will not react so well. Here is my proposal:

```
50 G=GETC: IF G<49 GOTO 50: IF G>50 GOTO 50
```

The same recipe can be applied to line 530. With an IF AND or an IF OR combination the DAI looks at both possibilities and then combines before taking any action, so it would be better to stop as soon as one is found to be wrong. This happens with an OR if the first condition is satisfied and with an AND if the first condition is not satisfied. Structurally AND and OR are preferred because others can more easily understand what is being tested, but for the sake of speed (with the DAI), simple IFs are faster. The score is updated at line 320. A couple of spaces are printed after S to ensure that any figures remaining from the previous line are wiped out. After the " " I would like to see a semicolon to restrict the winking cursor, or better still, make it invisible with POKE #75,32. In line 180 X+X would be quicker than 2.0*X. It is correct to use floating point numbers 1.0 and 2.0 because RND works with them and it is faster than with integers. The INT is superfluous because there is an assignment to an integer variable.

Now for the error! Koert rightly foresaw that there was a chance of a 'star' being placed where one already existed, resulting in 29 (or less), and making it impossible to catch 30. He solved this by testing if the spot chosen was empty; if it was not he lowered the loop counter by 1. This excellent solution looks logical but it does not work with the DAI. At the start of each FOR-NEXT loop DAI calculates the number of times the loop has to be run through; intervening changes to the loop variable O therefore have no effect on the number of loops. I do not like to see a FOR associated with two NEXTs. The test itself could also have been better, not only with the previously mentioned OR construction, but also to avoid wasting time by twice calculating the SCRN(XX,YY). Better would be: HELP=SCRN(XX,YY): IF HELP=15 OR HELP=3 THEN . . . or even better: IF SCRN(XX,YY) <>0 THEN . . . , but this is like carrying coals to Newcastle! the fault is nicer and it is done faster. Reducing the loop variable is meaningless, so how do we solve it? Several methods are available and I will give two:-

- 1) Keep however many stars are placed. If for example only 28 are there, then we need only catch 28 to obtain a new field. Instead of testing if SCRN(XX,YY)=0 we can update the counter by COUNTER = COUNTER +1 -SGN(SCRN(XX,YY)).
- 2) If the chosen point is occupied we choose another but to gain time, only choose another YY value.

REAL CIRCLES

(from DAInamic 18, page 319)

Remarks on the article "INCREMENTAL CIRCLE GENERATION" by F.VAN AMERONGEN in DAInamic 14.

This small program is really quite clever. The fast drawing speed is achieved by avoiding the conventional sines and cosines. The algorithm used does not originate from the equation for a circle; it is really for ELLIPSES with almost similar axis lying obliquely with an angle of 45 degrees to the X axis. It can easily be seen if K1 in line 30 is replaced by a higher value (eg = 30). (If a "number out of range" is given the original value of R should be reduced). The only points on these ellipses which are proper to a circle are (0,R), (R,0), (0,-R) and (-R,0).

To achieve a true circle one uses the equations

$$x = R \cos \theta \qquad y = R \sin \theta$$

which after differentiation give

$$dx = -y d\theta \qquad dy = x d\theta$$

This results in

$$x_2 \approx x_1 - y_1 \Delta \theta \qquad y_2 \approx y_1 + x_1 \Delta \theta$$

where $\Delta \theta$ is a small angular difference and \approx means approximately equal to.

The starting point (x_1, y_1) satisfies the equation

$$x_1^2 + y_1^2 = R^2$$

however one sees that the point (x_2, y_2) does not lie on the circle

$$x_2^2 + y_2^2 = R^2 (1 + \Delta \theta^2)$$

One sees that each point determined has co-ordinates that are too great by a factor

$$\sqrt{(1 + \Delta \theta^2)}$$

so that the iterative process will produce an accumulated factor of

$$(\sqrt{(1 + \Delta \theta^2)})^{n-1}$$

where n represents the number of sides of the approximate polygon

$$n = \frac{2\pi}{\Delta \theta}$$

and a spiral will be obtained instead of a circle. Finally, to obtain a circle each new co-ordinate must be divided by a factor

$$\sqrt{(1 + \Delta \theta^2)}$$

A program herewith clearly demonstrates the difference between F. van Amerongen's "pseudo circles" and true mathematical circles.

G. Doumont.

CIRCLES which are not real circles.

(DAInamic 18, page 320)

```

5      GOTO 2000
10     PRINT CHR$(12); MODE 0
15     CLEAR 2000; DIM U(190,0),V(190,0)
20     COLORG 0 5 10 15; R=90,0
35     MODE 6; XC=XMAX/2.0; YC=YMAX/2.0
40     DRAW XC,0 XC,YMAX 5; DRAW 0,YC XMAX,YC 5
50     X1=R; Y1=0,0; K1=3,0
55     FOR K1=3,0 TO 31,0 STEP 7,0
60     K=K1/X1; A=SQR(1,0+K*K); B=1,0/A; C=5,0
70     FOR I=0,0 TO (2,0*PI)/K; X2=(X1+K*Y1)*B; Y2=(Y1-K*X1)*B
80     GOSUB 1000; NEXT; WAIT TIME 50
110    FOR I=0,0 TO (2,0*PI)/K; X2=X1+K*Y1; Y2=Y1-K*X2; C=10,0; U(I)=X2; V(I)=Y2
120    GOSUB 1000; NEXT; WAIT TIME 100
122    IF K1=31 GOTO 130
124    C=0,0; DRAW R+XC,YC U(0,0)+XC,V(0,0)+YC C
126    FOR J=1,0 TO (2,0*PI)/K; DRAW U(J)+XC,V(J)+YC U(J-1,0)+XC,V(J-1,0)+YC C
128    NEXT J
130    NEXT K1; WAIT TIME 250; GOTO 10
1000   P=X1; Q=Y1; M=X2; N=Y2; DRAW P+XC,Q+YC M+XC,N+YC C
1100   X1=X2; Y1=Y2; RETURN
2000   PRINT " The circle in green is the true mathematical circle."
2010   PRINT " The orange one is the 'Van Amerongen version.'"
2020   PRINT " For each value of K1 between 3 and 31 in steps of"
2030   PRINT " 7 both circles are presented on the screen, after"
2040   PRINT " which the orange circle is deleted. By this means"
2050   PRINT " the 'ovalisation' of the orange polygon should be"
2060   PRINT " very clear to see."
2070   WAIT TIME 750; GOTO 10
    
```

HIGH SPEED DATA LOADER (HSDL)

(from DAInamic 18, page 330)

In order to overcome the annoyance of long loading times for large programs on tape we came up with the idea of a High Speed Data Loader. It can read-in at a rate rather more than 10 kilobytes per second. The hardware and software is so designed to avoid problems where DCRs are in use, but as we do not own disc drives we cannot say how the HSDL will work with them. The HSDL is able to read-in BASIC, MLP and combined programs in accordance with whatever is in its EPROMs. We started with a card for 4 EPROMs of types 2716, 2732 or 2764 which give a maximum program size of 8, 16 or 32 kilobytes respectively. This EPROM card comes with a connector in the HSDL which in turn is connected via a ribbon cable to the DAI's DCE connector.

When all is connected the rest is simple; switch on the DAI and in Basic, type in, RDL1...4, Return. When the cursor jumps to the next line the program is in. If the program is a Basic one, or an MLP and Basic combination, a RUN command is all that is needed now to set it going. It is a bit more difficult with MLP; go first to UTility, type Z3 and then Gxxxx, the start address of the program, and then it will run. Since all programs will not be 8K or 16K we have made it possible to put additional programs on one card. Each EPROM is addressed by the appropriate command, RDL1 for the first, RDL4 for the 4th. When a program takes up more

than one EPROM, calling the overflow EPROM will produce the message "NOT AVAILABLE". The space unused by a program in an EPROM is unfortunately wasted as it is only possible to address whole EPROMs.

The high speed data loader consists of the following parts:-

- 1) EPROM card on the X bus on which is an EPROM containing the operating software.
- 2) The data loader containing the hardware, with a socket for the EPROM card.
- 3) The EPROM card with the required program(s).

Two prototypes have been constructed and we are now drafting the definitive printed circuit. If there is sufficient interest we would like to start a small production run and discuss with Dainamic the possibility of putting programs from their library into PROM or EPROM, possibly on a circuit board with a connector.

Once one has the basic unit the only thing then required is the EPROM board containing the program. The important factor is the cost. We have made a provisional estimate and arrived at a cost of about 275 Guilders for a basic unit, consisting of:-

EPROM with operating software;
High speed data loader;

EPROM card with a demo program in an EPROM.

We must still find prices for the extras like xbus printed circuit, ribbon cable with connectors, EPROM cards etc.

After a talk with Dainamic we have a couple of comments and for you, a few questions:

- 1) If you use your DAI mainly for a specific purpose like SPL, DNA, Word Processing, VIDITEL (Dutch Prestel service), or anything similar where only 1 or 2 programs are needed, then HSDL is certainly what you need, with your program(s) in EPROM.
- 2) DAInamic usually supplies on one cassette, several programs for a comparatively low price. This would not be possible with EPROMs, so the relative cost would be more.

Now for our questions:

- a) Which programs would you wish to see in EPROM ?
- b) Would you prefer EPROMs to be loose or fully mounted ?
- 3) How much would you be prepared to pay ? Beyond the fixed cost of the EPROMs, construction etc, there is the cost of the programs themselves (fixed cost of hardware about 80 Guilders for 16 kilobytes). Please give a realistic indication of price. Consider how much games computer programs cost !

We would like to know if you are interested and what your ideas, wishes and queries are. Drop us a post card and we can then go into the subject further in a future issue of DAInamic.

H.Kop,
Burg. Smeetsstraat 39,
6151 GM MUNSTERGELEEN
Netherlands.

H.Rison,
Luxemburgstr. 17,
6164 BS GELEEN,
Netherlands.

Jeroen demo 9

PAGE 01 -- JEROEN DEMO 9

```
90 PRINT CHR$(12):MODE 5A:MODE 5A
100 COLORG 0 13 0 0:COLORT 0 13 0 0:GOSUB 2010
110 A$=" Hartelijk welkom bij jeroentel ":GOSUB 1110
120 A$=" ":GOSUB 1120
130 A$=" Met deze thuiscomputer is het mogelijk ":GOSUB 1130
140 A$=" toegang te krijgen tot VIDITEL ":GOSUB 1140
150 A$=" de gegevensbank van de PTT via de tele- ":GOSUB 1155
160 A$=" foon.Het is dan ook mogelijk zelf infor- ":GOSUB 1160
170 A$=" matie aan te bieden ":GOSUB 1170
180 Y=204:X=15:B=0
200 READ A:A=A*4:X1=X+A:IF A=2000 THEN Y=Y-6:X=15:B=0:GOTO 200
205 IF A=4000 THEN GOSUB 1180:WAIT TIME 300:LOAD "JEROEN DEMO 10"
210 FILL X+1,Y X1,Y-5 B:IF B=0 THEN B=3:X=X1:GOTO 200
230 B=0:X=X1:GOTO 200
1110 POKE #9943,#6A:I=#9941:M=#98E9:GOSUB 3000:RETURN
1120 POKE #98E9,#6A:I=#98E7:M=#9890:GOSUB 3000:RETURN
1130 POKE #988F,#6A:I=#988D:M=#9835:GOSUB 3000:RETURN
1140 POKE #9835,#6A:I=#9833:M=#97DB:GOSUB 3000:RETURN
1150 POKE #9349,#6A:I=#9347:M=#92EF:GOSUB 3000:RETURN
1155 POKE #8E5D,#6A:I=#8E5B:M=#8E03:GOSUB 3000:RETURN
1160 POKE #8E03,#6A:I=#8E01:M=#8DA9:GOSUB 3000:RETURN
1170 POKE #8DA9,#6A:I=#8DA7:M=#8D4F:GOSUB 3000:RETURN
1180 POKE #8D4F,#6A:POKE #8D4B,3:POKE #8D4D,32:FOR I=#8D49 TO #8CFB STEP -2:
POKE I,1:NEXT
1190 POKE #8CF5,2:POKE #8CF7,32:POKE #8CF9,2:RETURN
2010 PRINT CHR$(12):POKE #75,32:COLORT 0 13 0 0:COLORG 0 13 0 0:MODE 5A:MODE
5A:Y=204:X=0
2020 POKE #BFEF,#6A:FOR I=#BFED TO #BF96 STEP -2:POKE I,32:NEXT
2030 POKE #BF95,#68:FOR I=#BF93 TO #BF3C STEP -2:READ C:POKE I,C:NEXT
2040 POKE #BF3B,#6A:POKE #BF39,32:POKE #BF37,3:FOR I=#BF35 TO #BEE7 STEP -2:
POKE I,1:NEXT
2050 POKE #BEE5,2:POKE #BEE3,32:RETURN
3000 Q=LEN(A$):FOR P=1 TO Q STEP 1:L$=MID$(A$,P,1)
3010 1 W=ASC(L$):POKE I,W:I=I-2:IF I>M THEN NEXT P
3020 1 RETURN
10000 1 DATA 32,32,74,101,114,111,101,110,116,101,108,32,32,32,32,32,32,32,
1 32,32,32,32,32,32,32,32,50,49,79,52,50,54,52,97
10010 1 DATA 32,32,32,32,52,53,99,32,32
10020 1 REM DATA 32,32,74,101,114,111,101,110,116,101,108,32,32,32,32,32,32,32,
1 32,32,32,32,32,32,32,32,50,49,79,52,50,54,53,97
10030 1 REM DATA 32,32,32,32,52,53,99,32,32
10040 1 REM DATA 32,32,74,101,114,111,101,110,116,101,108,32,32,32,32,32,32,32,
1 32,32,32,32,32,32,32,32,50,49,79,52,50,54,54,97
10050 1 REM DATA 32,32,32,32,52,53,99,32,32
61000 1 DATA 0,10,9,3,8,6,500,1,10,8,3,9,4,500,2,2,5,3,6,2,1,2,9,2,500
61010 1 DATA 2,2,6,2,6,2,1,2,9,2,500,2,2,6,2,6,2,1,2,9,2,500,2,2,6,2,5,3,1,3,8,
1 2,500
61020 1 DATA 2,2,6,2,5,2,3,2,8,2,500,2,2,6,2,5,2,3,2,8,2,500,2,2,6,2,4,9,7,2,
1 500
61030 1 DATA 2,2,6,2,4,9,7,2,500,2,2,6,2,4,2,5,2,7,2,500,2,2,6,2,3,3,5,3,6,2,
1 500
61040 1 DATA 2,2,5,3,3,2,7,2,6,2,500,1,10,3,2,9,2,4,4,500,0,10,2,6,5,6,1,6,500,
1 1000
```

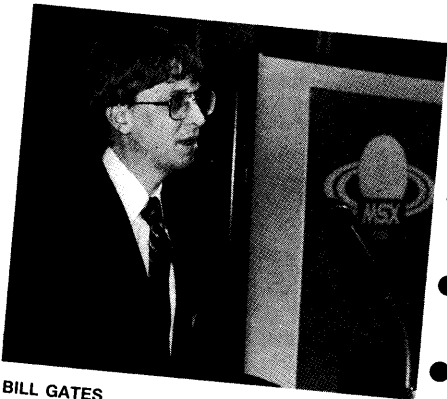
MSX

CLUB magazine

september-oktober '85



KAY NISHI



BILL GATES

- Programmeertechnieken
- Muziek potpourri
- Goniometrische functies
- Schematechnieken
- Para
- Drijfzand
- Music box
- Test SPECTRAVIDEO 728
- Kleurpotloden
- BASIC tekstverwerker
- Time-programma
- Test PUDDLES software
- Cursus Z80 deel 3

SOON AVAILABLE : SUPERFONT 150 STYLES !

DIGITAL DIGITAL DIGITAL

SOON AVAILABLE : SUPERFONT 150 STYLES !

ORANGE ORANGE ORANGE

SOON AVAILABLE : SUPERFONT 150 styles !

STANDARD STANDARD STANDARD

SOON AVAILABLE : SUPERFONT 150 styles !

LARGEST LARGEST LARGEST

SOON AVAILABLE : SUPERFONT 150 styles !

SCRIPT SCRIPT SCRIPT

SOON AVAILABLE : SUPERFONT 150 styles !

DOUBLE DOUBLE DOUBLE

SOON AVAILABLE : SUPERFONT 150 styles !

THICK THICK THICK