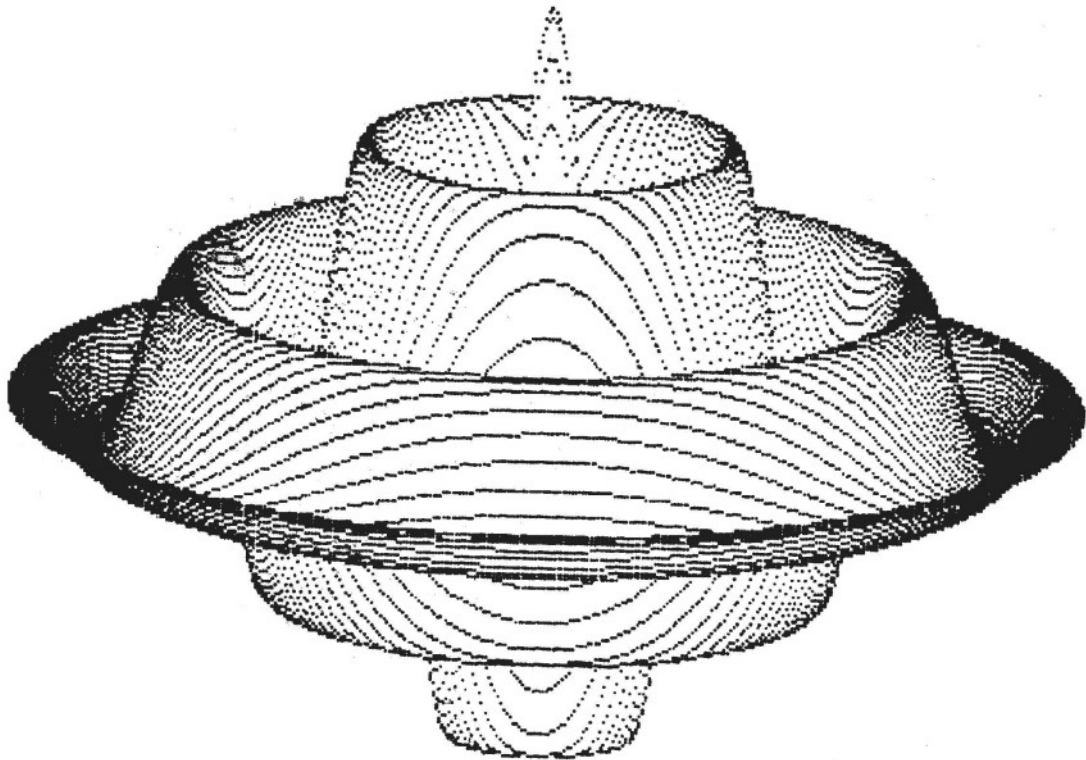


# DAI

## NAMIC

NUMMER 0/1 \*\*\* 1980



GEDRUKTE PERIODIEK verschijnt tweemaandelijks

Verantw. Uitgever : W. HERMANS HEIDE 98 3171 WESTMEERBEEK

COLOFON

DAInamic verschijnt tweemaandelijks.  
 abonnementsprijs is inbegrepen in de  
 jaarlijkse contributie:

750 Bfr 50 Gld 50 Dm

Bij toetreding worden de verschenen  
 nummers van de jaargang toegezonden.

DAInamic redactie:

- Dirk Bonn 
- Freddy De Raedt
- Wilfried Hermans
- Jules Meulenbergs
- Jos Schepens
- Roger Theeuws
- Bruno Van Rompaey
- Jef Verwimp

vormgeving :Ludo van Mechelen

U wordt lid door storting van de  
 contributie op nr406-3016141-33 van  
 KREDIETBANK WESTMEERBEEK, via bank-  
 instelling of POSTGIRO.  
 Abonnement loopt van januari tot  
 december.

U kan telefonisch contact nemen op  
 nr 016/698623.

correspondentieadres:

DAInamic  
 Heide 98  
 3171 WESTMEERBEEK BELGIE

DAInamic verschijnt de eerste week van  
 de pare maanden.

Bijdragen zijn steeds welkom.

4		3		2		1	
HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC
1	4096	1	256	1	16	1	1
2	8192	2	512	2	32	2	2
3	12288	3	768	3	48	3	3
4	16384	4	1024	4	64	4	4
5	20480	5	1280	5	80	5	5
6	24576	6	1536	6	96	6	6
7	28672	7	1792	7	112	7	7
8	32768	8	2048	8	128	8	8
9	36864	9	2304	9	144	9	9
A	40960	A	2560	A	160	A	10
B	45056	B	2816	B	176	B	11
C	49152	C	3072	C	192	C	12
D	53248	D	3328	D	208	D	13
E	57344	E	3584	E	224	E	14
F	61440	F	3840	F	240	F	15

belangrijke ASCII-waarden in DAIPC

functie/symbool	HEX	DEC
back-space	8	8
TAB	9	9
linefeed	A	10
clear screen	C	12
CURSOR UP	10	16
CURSOR DOWN	11	17
CURSOR LEFT	12	18
CURSOR RIGHT	13	19
space-bar	20	32
�	30	48
A.	41	65
a	61	97
pijltje rechts	89	137
pijltje links	88	136
pijltje boven	5E	94
pijltje onder	8C	140
volle blok	FF	255
verticale lijn	A	10
horizontale lijn	B	11
6 hor lijnen	1D	29

ASCII - HEX - ASCII CONVERSION TABLE

MSD	0	1	2	3	4	5	6	7	
LSD	000	001	010	011	100	101	110	111	
0	0000	NUL	DLE	SP	0	@	P	^	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	\$	4	D	T	d	t
5	0101	ENC	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	.	7	G	W	g	w
8	1000	BS	CAN	(	8	H	X	h	x
9	1001	HT	EM	)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[	k	{
C	1100	FF	FS	,	<	L	\	l	
D	1101	CR	GS	-	=	M	]	m	}
E	1110	SO	RS	.	>	N	^	n	~
F	1111	SI	VS	/	?	O	_	o	DEL

# REMARK

DAInamic  
DAI USERS CLUB

WESTMEERBEEK DEC/80

Beste DAI-gebruiker,

Dit proefnummer is een heruitgave, met artikels uit NEWSLETTERS 0 en 1, welke verschenen in juli en september 80, maar vlug uitgeput waren.

Wij proberen met de NEWSLETTERS het hele toepassingsgebied van DAI PERSONAL COMPUTER te behandelen: van kleunige spelletjes tot industriële toepassingen en alles wat daar tussen ligt.

Naast artikels van de redactie zal U in de rubrieken bijdragen vinden van clubleden: het blad staat open voor alle informatie over DAIPC. We proberen de inhoud onder te brengen in herkenbare rubrieken:

REMARK : redactiepraatje  
TALK : mededelingen, correspondentie  
CATALOG : bespreking van beschikbare programma's  
READ : tijdschrift & boekbesprekingen  
LOOK : over DAI-BASIC  
LIST : programma's  
PEEK&POKE: machinetaal, truucs, interessante adressen  
SOUND : klank en muziek met DAIPC  
TEACH : educatieve toepassingen  
NEW : nieuwe producten

Wij wensen U leezame, prettige uurtjes met de NEWSLETTERS, welkom bij de club...

---

Dear DAUser,

This edition is a compilation of articles from NEWSLETTER 0 and 1. We regret it is not possible to offer you an English edition. If the Dutch language is no magic for you then you can join the Holland-Belgium club, there is an annual fee of 750 Bfr/50 Gld/50 Dm/12 pound. Otherwise you will have to wait until a local club is established. On subscription we will send you all the editions from January on.

# BLADWIJZER

2	REMARK	Redactiepraatje
3	BLADWIJZER	
4	LIST	Getalconversies J.Verdonk
5	LIST	Getalconversies
6	LIST	Getalconversies
7	PEEK&POKE	DCE-bus beschrijving
8	PEEK&POKE	DCE-bus 8255 controlwords/aansluiting
9	PEEK&POKE	DCE-bus BUS MODE
10	PEEK&POKE	DCE:BASIC OUT/ INP ROUTINES
11	LIST	Vier op een rij
12	LIST	Vier op een rij
13	LOOK	tips...Testtape...sound monitor...VAL/STR
14	PEEK&POKE	aansluiting N2219
15	LIST	Raketspel H.Bakker
16	LIST	Raketspel H.Bakker
17	LIST	Raketspel H.Bakker/Grafiek M.Vermeulen
18	PEEK&POKE	Videoram in MODE Ø
19	LIST	Propeller
20	LIST	Propeller + grafische characterset EPSON
21	LIST	4 COLOR DEMO LISSAJOUS
22	LOOK	Merging/grafische opdrachten
23	LIST	Citröen B 14 COACH 1927
24	LIST	Citröen
25	LIST	Citröen
26	READ	"How to program microcomputers"
27	PEEK&POKE	DAIpc I/O aansluitingen
28	PEEK&POKE	Het gebruik van de PADDLE-EVENT ingangen
29	SOUND	Frekwentiebereik muziekinstrumenten
30	SOUND	frekwenties + sireentjes
31	LIST	BLOXY COLORG DEMO
32	PEEK&POKE	8080 instructieset
33	PEEK&POKE	combinaties BASIC-machinetaal +"WARNING"
34	CATALOG	programmabibliotheek *
35	CATALOG	programmabibliotheek ** ***

\*\*\*\*\*

# LIST

## GETALCONVERSIES BK MØ J. VERDONK

```
1  REM *****
2  REM ** OMZETTEN VAN GETALSTELSLS **
3  REM **          J. P. VERDONK          **
4  REM **          APRIL 1980          **
9  REM *****
10 DIM A$(9.0),G$(10.0),A(10.0),B(10.0),G(10.0)
11 PRINT CHR$(12)
20 PRINT "VAN WELK (X) NAAR WELK (Y) TALSTELSEL (MAX 16)?"
21 PRINT "VOOR VERANDEREN VAN TALSTELSEL TYP GETAL =0"
22 PRINT "VOOR EINDE PROGRAMMA TYP X EN Y=0"
30 INPUT "X= ";X:PRINT " ";:INPUT "Y= ";Y:PRINT :GOTO 300
31 REM VERANDEREN VAN EERDER GEKOZEN TALSTELSEL
40 IF X=0.0 AND Y=0.0 THEN END
50 INPUT "GETAL ";Z$:C=0.0:D=0.0:E=-1.0:H=0.0:IF Z$="" THEN PRINT
:PRINT :GOTO 30
51 REM OMZETTEN NAAR DECIMAAL (TOT REGEL 150)
60 IF X=10.0 THEN C=VAL(Z$):GOTO 160
70 FOR I=LEN(Z$)-1.0 TO 0.0 STEP -1.0
80 A$(I)=MID$(Z$,I,1)
81 REM OMZETTEN VAN STRING NAAR ASCII WAARDE
90 A(I)=ASC(A$(I))
91 REM WAARDE BEPALEN PER CIJFER
100 IF A(I)>47.0 AND A(I)<58.0 THEN B(I)=A(I)-48.0:GOTO 350
101 REM OOK VOOR CIJFERS INDIEN NODIG
```

# LIST

## GETALCONVERSIES Part 2

```
120 PRINT :PRINT "HET GETAL ";A$(I);" KEN IK NIET!":GOTO 50
130 C=C+(B(I)*INT((X^D)+0.5)):D=D+1.0
140 NEXT I
150 IF Y=10.0 THEN PRINT " -";X;" = ";C;" -10":GOTO 50
151 REM TERUGREKENEN IN ANDER STELSEL
160 E=E+1.0
170 F=INT(C/INT((Y^E)+0.5))
171 REM GROOTSTE EXPONENT BEPALEN
180 IF F>=1.0 THEN 160
181 REM NU GAAT HET NIET MEER DUS 1 ERAF
190 E=E-1.0:H=H+1.0
191 REM VAN LINKS > RECHTS
200 G(E)=INT(C/INT((Y^E)+0.5))
210 C=C-(G(E)*INT((Y^E)+0.5))
211 REM OMZETTEN NAAR ALFANUMERIEKE ASCII WAARDE
220 IF G(E)>=0.0 AND G(E)<=9.0 THEN G(E)=G(E)+48.0
230 IF G(E)>=10.0 AND G(E)<=16.0 THEN G(E)=G(E)+55.0
231 REM HETZELFDE MET DE REST
240 IF E>0.0 THEN 190
241 REM NU GETALLEN UITPRINTEN
250 PRINT " -";X;" = ";
255 IF G(H)<47.0 THEN G(H)=32.0:H=H-1.0
260 IF G(H)>47.0 THEN PRINT CHR$(G(H));:G(H+1.0)=0.0:G(H)=0.0:IF H>1.0
THEN H-1.0
270 IF H<1.0 THEN PRINT " -";Y:GOTO 50
```

## GETALCONVERSIES part 3

---

```
280  GOTO 255
290  STOP
300  IF X<0.0 OR X>16.0 THEN 20
310  IF Y<0.0 OR Y>16.0 THEN 20
320  GOTO 31
330  STOP
350  IF B(I)>(X-1.0) THEN PRINT :PRINT "HET CIJFER ";A$(I);" HOORT NIET
      THUIS IN HET ";X;"-TALLIG STELSEL !":GOTO 50
360  GOTO 130
370  STOP
```

---

VAN WELK (X) NAAR WELK (Y) TALSTELSEL (MAX 16)?

VOOR VERANDEREN VAN TALSTELSEL TYP GETAL =0

VOOR EINDE PROGRAMMA TYP X EN Y=0

X= ?10 Y= ?2

GETAL ?137 - 10.0 = 10001001 - 2.0

GETAL ?255 - 10.0 = 11111111 - 2.0

GETAL ?60 - 10.0 = 111100 - 2.0

GETAL ?0

X= ?10 Y= ?16

GETAL ?254 - 10.0 = FE - 16.0

GETAL ?33 - 10.0 = 21 - 16.0

GETAL ?100 - 10.0 = 64 - 16.0

# PEEK & POKE

## DE DCE - BUS

DE DCE-BUS IS DE WEG OM DE COMPUTER TE VERBINDEN MET VERDERE APPARATEN. VOORBEELDEN : FLOPPY-DISKS, PRINTER MET PARALLEL INPUT, DOOR DAI GEFABRICEERDE REAL-WORLD-CARDS , (EENVOUDIGE ?) ZELF ONTWERPEN SCHAKELINGEN,...

ER ZIJN 2 METHODES OM DE DCE-BUS TE GEBRUIKEN :

- 1) GEBRUIKEN ALS EEN EENVOUDIGE 3 X 8 BITS PARALLEL I/O. WE ZIJN DAN VRIJ IN DE OPZET VAN DE TE VERBINDEN SCHAKELING EN DE BESTURINGSSOFTWARE. WEL OPGELET VOOR DE JUISTE SPANNING ( ZIE TTL NIVEAU'S ) EN OVERBELASTING ( MAX SINK = 1.6 MA / .4 V ).
- 2) GEBRUIKEN VOLGENS DE DCE-BUS STANDAARD DOOR OFWEL ENKEL DAI REAL-WORLD-CARDS TE GEBRUIKEN OF ZELF COMPATIBELE SCHAKELINGEN ONTWERPEN

### A) BESCHRIJVING VAN DE DCE-BUS:

INTERN IN DE PC BESTAAT DE DCE-BUS VOORNAMELIJK UIT 1 IC : INTELS 8255 (PROGRAMMABLE PERIPHERAL INTERFACE). VANAF NU ZULLEN WE DIT IC AANDUIDEN MET DE NAAM GIC (GENERAL INTERFACE CONTROL).

DIT IC BEVAT 2 8-BITS POORTEN (P0 EN P1) EN 2 4-BITS POORTEN (P2H EN P2L)

DE GIC IS ZO ONTWERPEN DAT WE DEZE 4 POORTEN AFZONDERLIJK ALS IN- OF UITGANG KUNNEN INSTELLEN. VOOR DEZE INSTELLING STUREN WE EEN KONTROLEBYTE NAAR EEN INTERN REGISTER IN DE GIC.

VIA DE DATA, ADRES EN CONTROL BUS IS DE 8255 VERBONDEN MET DE 8080.

ADRESSEN VAN DE GIC : #FE00 : POORT 0  
#FE01 : POORT 1  
#FE02 : POORT 2H + 2L  
#FE03 : KONTROLEBYTE (NIET UITLEZEN !)

EEN POORT DIE ALS UITGANG INGESTELD IS GEDRAAGT ZICH ALS EEN LATCH. DWZ ALS WE ER EEN BEPAALDE WAARDE INSCHRIJVEN, DAN BLIJFT DEZE WAARDE BEHOUDEN TOT WE ER EEN NIEUWE WAARDE INSCHRIJVEN. DEZE WAARDE KUNNEN WE OOK UITLEZEN MET BU PEEK. TOT HIERTOE HEBBEN WE ENKEL MODE0 VAN DE 8255 BESCHREVEN.

DE 8255 HEEFT OOK NOG MODES 1 EN 2 DIE EEN ZEER SNELLE IO TOELATEN (WERKEN MET HANDSHAKING EN INTERRUPT SIGNALEN).

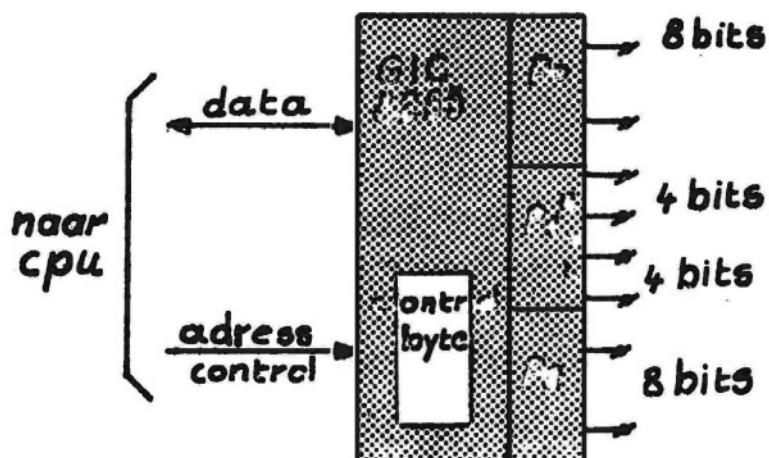
IN DIT ARTIKEL BEPERKEN WE ONS TOT MODE0, OMDAT MODES 1 EN 2 VEEL COMPLEXER ZIJN.

BIJ WIJZIGING VAN DE MODE WORDEN ALLE OUTPUTLATCHES GERESSET. BIJ HARD RESET WORDEN ALLE POORTEN ALS INGANG GEZET.

OP DE VOLGENDE PAGINA GEVEN WE EEN LIJST VAN DE HEX KONTROLE-BYTES VOOR EEN BEPAALDE INSTELLING VAN DE GIC (ENKEL MODE 0), MET DAARNAAST EEN TEKENING VAN DE GIC .



P0	P2H	P2L	P1	KONTR
OUT	OUT	OUT	OUT	80
OUT	OUT	INP	OUT	81
OUT	OUT	OUT	INP	82
OUT	OUT	INP	INP	83
OUT	INP	OUT	OUT	88
OUT	INP	INP	OUT	89
OUT	INP	OUT	INP	8A
OUT	INP	INP	INP	8B
INP	OUT	OUT	OUT	90
INP	OUT	INP	OUT	91
INP	OUT	OUT	INP	92
INP	OUT	INP	INP	93
INP	INP	OUT	OUT	98
INP	INP	INP	OUT	99
INP	INP	OUT	INP	9A
INP	INP	INP	INP	9B



VOOR DE NUMMERS OP DE CONNECTOR VERWIJZEN WE NAAR DE PC  
MANUEL PAGINA 37 (PIN ON PC CARD). WIJZIG VOLGENDE FOUTEN :

P0B2 : ? WORDT 12  
P0B3 : 12 WORDT 10  
+5V : 3 WORDT 1  
GND : ? WORDT 4  
-5V : 4 WORDT 3

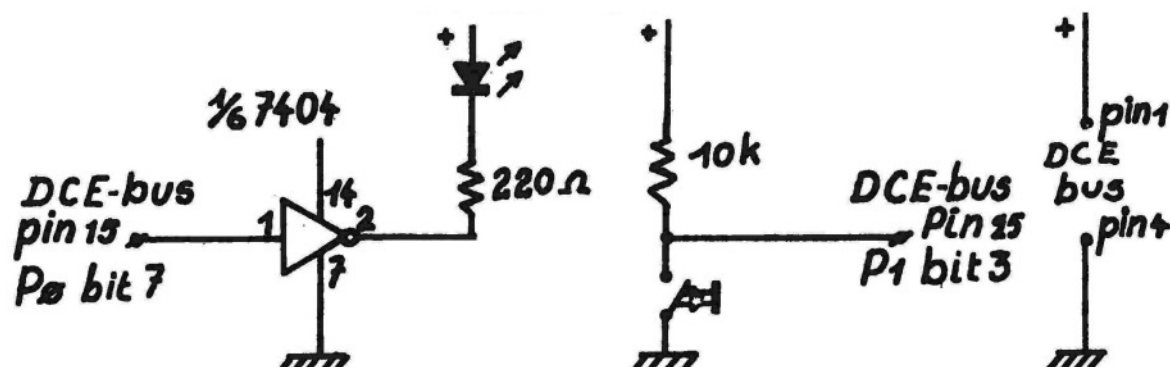
#### B) VOORBEELD AANSLUITING EN BASIC BESTURING :

EENVOUDIG VOORBEELD MET 1 BIT OUTPUT EN 1 BIT INPUT.

OUTPUT : BIT 7 VAN POORT 0 DIE LED ON/OFF STUURT  
BUFFERING MET 7404 TTL IC

INPUT : BIT 3 VAN POORT 1 DIE BESTUURD WORDT DOOR EEN  
DRUKKNOP

OPM : INGEDRUKT = 0 , LOS = 1 !



BESTURING :

DEFINIEER VARIABLE GIC ALS #FE00

GIC = #FE00

SET 8255 KONTROLEBYTE (P0 = OUT, P1 EN P2 = INP)

POKE GIC+3, #8B

```

BESTUUR LED MET VARIABELE A (A=0:LED UIT, A=1:LED AAN) :
SHL 7 WORDT TOEGEVOEGD OM BIT 7 TE BEPALEN :
      POKE GIC,A SHL 7 : REM BIT 7
ZET VARIABELE B ZOALS STAND DRUKKNOP (LOS : B=0, INGEDRUKT :
B=1) : IXOR #FF OMDAT INPUT NEGATIEF WAAR IS ; SHR 3 EN
IAND .1 OM TE BEKOMEN DAT B ENKEL 0 OF 1 WORDT :
      B = ( ( PEEK(GIC+1) IXOR #FF ) SHR 3 ) IAND 1

```

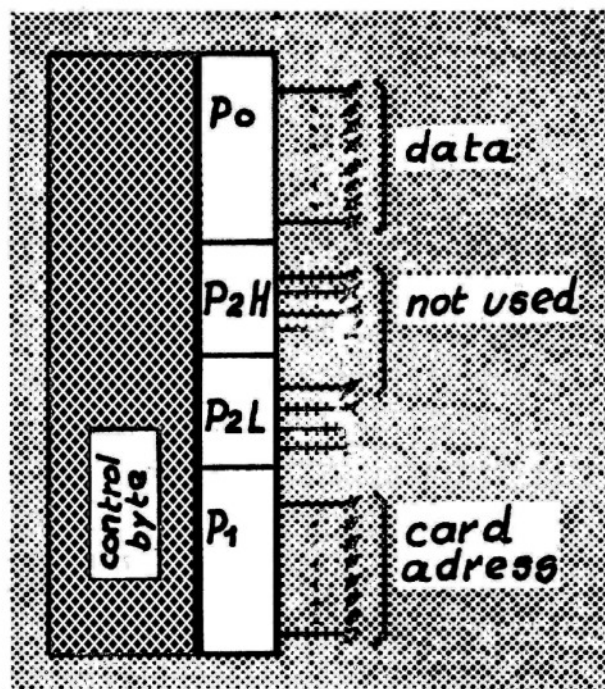
### C) DE DCE BUS MODE

IN DE DCE BUS MODE WORDEN DE POORTEN VAN DE 8255 VOLGENS  
BEPAALENDE AFSPRAKEN GEBRUIKT. HET DOEL IS EEN UNIVERSEEL  
SYSTEEM TE BEKOMEN WAARAAN VERSCHIEDENE IO KAARTEN TEGELIJK  
VERBONDEN ZIJN.

```

SIGNALLEN : DATA      : BIDIRECTIONEEL KANAAL VOOR DATA NAAR
                        EN VAN KAART
RD'         : PULS VOOR LEZEN DATA VAN KAART OP DATA      P2 BIT2
WR'         : PULS VOOR SCHRIJVEN VAN DATA NAAR KAART      P2 BIT1
BUS EXP     : BUS EXPAND : UITBREIDING TOT 31 KAARTEN      P2 BIT0
              (TOT 15 KAARTEN : GEBRUIKT ALS BUS ON/OFF
              1 = OFF, 0 = ON)
CARD ADRES : SELEKTEERT 1 KAART + VERDERE REGISTER
              SELEKTIE OP KAART

```



DE BASIC INSTRUKTIES OUT EN INP WORDEN GEBRUIKT OM DATA UIT  
TE WISSELEN MET DE KAARTEN. OP VOLGENDE PAGINA GEVEN WE EEN  
ASSEMBLER LISTING VAN DE ROUTINES DIE AANGEROEPEN WORDEN  
DOOR DEZE BASIC INSTRUKTIES. BESTUDERING VAN DEZE LISTING  
ZAL DUIDELIJK MAKEN HOE HET DCE BUS PROTOCOL VERLOOPT.

```

002          *THIS ASSEMBLER PROGRAM SHOWS THE ROM RESIDENT ROU-
003          *TINES OUT AND INP TO DRIVE THE DCE-BUS
004          *
005          *STORE BUS-ADRES IN REGISTER D
006          *FOR OUT : STORE DATA IN REGISTER E
007          *FOR INP : ROUTINE RETURN WITH DATA IN REGISTER E
008          *
009          GIC      EQU      :FE00      ADRES OF GIC DATABUS
010          RWMOP    EQU      :80        GIC MODE ALL PORTS OUTPUT
011          RWMIP    EQU      :90        GIC MODE P0=INP,OTHERS=OUT
012
013          OUT      PUSH     PSW
014          D8C9     E5              PUSH     H
015          D8CA     2103FE          LXI      H,GIC+3      ADRES GIC CONTROL IN H,L
016          D8CD     3680            MUI      M,RWMOP      SET GIC MODE
017          D8CF     2B              DCX      H              ADRES GIC P2 IN H,L
018          D8D0     36FE            MUI      M,:FE         CLEAR BUS EXPAND SIGNAL
019          D8D2     EB              XCHG    IN L: DATA ; IN H: BUSADRES
020          D8D3     2200FE          SHLD    GIC           DATA IN P0 : BUSADRES IN P1
021          D8D6     EB              XCHG    ADRES GIC P2 IN H,L
022          D8D7     34              INR     M              SET BUS EXPAND SIGNAL
023          D8D8     36FD            MUI      M,:FD         SET WRITE STROBE TRUE  (#)
024          D8DA     36FF            MUI      M,:FF         RESET STROBE
025          D8DC     35              DCR     M              CLEAR BUS EXPAND SIGNAL
026          D8DD     E1              POP     H
027          D8DE     F1              POP     PSW
028          D8DF     C9              RET
029          D8E0     F5              BINP     PUSH     PSW
030          D8E1     E5              PUSH     H
031          D8E2     2103FE          LXI      H,GIC+3      ADRES GIC CONTROL IN H,L
032          D8E5     3690            MUI      M,RWMIP      SET GIC MODE
033          D8E7     2B              DCX      H              ADRES GIC P2 IN H,L
034          D8E8     36FE            MUI      M,:FE         CLEAR BUS EXPAND SIGNAL
035          D8EA     7A              MOV     A,D            BUSADRES IN A
036          D8EB     3201FE          STA     GIC+1         STORE BUSADRES IN GIC P1
037          D8EE     34              INR     M              SET BUS EXPAND SIGNAL
038          D8EF     36FB            MUI      M,:FB         SET READ STROBE TRUE  (#)
039          D8F1     3A00FE          LDA     GIC           DATA TO A
040          D8F4     5F              MOV     E,A            DATA IN E
041          D8F5     36FF            MUI      M,:FF         RESET STROBE
042          D8F7     35              DCR     M              CLEAR BUS EXPAND SIGNAL
043          D8F8     E1              POP     H
044          D8F9     F1              POP     PSW
045          D8FA     C9              RET
046          * (#) = ON THIS MOMENT THE DATA EXCHANGE ON THE
047          *          DCE-BUS TAKE PLACE

```

048 D8FB END

\*\*\*\*\*  
\* S Y M B O L T A B L E \*  
\*\*\*\*\*

BINP D8E0 GIC FE00 OUT D8C8 RWMIP 0090  
RWMOP 0080



# LIST

```
60 PRINT CHR$(12)
100 REM SPELER A
110 INPUT "SPELER A....WELKE RIJ ";R
111 IF R>7 GOTO 110
115 PRINT
116 FOR X=10.0 TO (R+1.0)*12.0+20.0:NOISE 0 15:DRAW X,100 X,105 3:NEXT
117 SOUND OFF
120 C=3.0:GOSUB 1000
130 GOSUB 2000
140 FILL 10,100 (R+1)*12+30,110 0
200 REM SPELER B
210 INPUT "SPELER B.....WELKE RIJ";R
211 IF R>7 GOTO 210
215 PRINT
217 FOR X=XMAX-10.0 TO (R+1.0)*12.0+25.0 STEP -1.0:NOISE 0 15:DRAW X,10
0 X,105 14:NEXT
218 SOUND OFF
220 C=14.0:GOSUB 1000
230 GOSUB 2000
240 FILL XMAX-11,100 (R+1)*12+25,105 0
250 GOTO 100
1000 FOR X=1.0 TO 7.0:IF U(R,X)=8.0 THEN GOSUB 1500:U(R,X)=C:RETURN
1010 NEXT:PRINT "DE RIJ IS AL VOL !!":RETURN
1500 SOUND 1 0 15 0 FREQ(1000.0):SOUND 1 0 15 2 FREQ(100.0)
1505 FOR M=7.0 TO X+1.0 STEP -1.0
1510 FILL R*12+30,M*12 R*12+38,M*12+8 C
1520 FILL R*12+30,M*12 R*12+38,M*12+8 8
1525 NEXT
1530 FILL R*12+30,X*12 R*12+38,X*12+8 C
1535 SOUND OFF
1540 RETURN
2000 FOR X=1.0 TO 7.0:FOR Y=1.0 TO 7.0
2010 FILL X*12+30,Y*12 X*12+8+30,Y*12+8 U(X,Y)
2020 NEXT:NEXT
2030 FOR X=1.0 TO 7.0:FOR Y=1.0 TO 4.0
2040 IF U(X,Y)<>8.0 THEN IF U(X,Y)=U(X,Y+1.0) AND U(X,Y)=U(X,Y+2.0) AND
U(X,Y)=U(X,Y+3.0) THEN GAME=U(X,Y):GOTO 3000
2045 NEXT:NEXT
2049 FOR Y=1.0 TO 7.0:FOR X=1.0 TO 4.0
2050 IF U(X,Y)<>8.0 THEN IF U(X,Y)=U(X+1.0,Y) AND U(X,Y)=U(X+2.0,Y) AND
U(X,Y)=U(X+3.0,Y) THEN GAME=U(X,Y):GOTO 3000
2055 NEXT:NEXT
2100 RETURN
3000 IF GAME=14.0 GOTO 3500
3010 FOR X=10.0 TO XMAX:DRAW X,0 X,YMAX 3:NEXT:GOTO 20
3500 FOR X=XMAX-10.0 TO 0.0 STEP -1.0:DRAW X,0 X,YMAX 14:NEXT:GOTO 20
```

### EEN TEST-TAPE VOOR DE CASSETTE-INTERFACE

ALS ER NOG PROBLEMEN ZIJN MET VOLUME EN TOON VOOR DE CASSETTE-INTERFACE KAN EEN TEST-TAPE ERG NUTTIG ZIJN. WE MAKEN DEZE TAPE DOOR EEN 20-TAL HERKENNINGSSIGNALEN NA ELKAAR OP TE NEMEN. WE CLEAREN ONZE DAIPc DOOR UIT EN AAN TE ZETTEN OF DOOR "NEW" OF DOOR DE RESETKNOP IN TE DRUKKEN. NU GAAN WE DIT BLANCO-PROGRAMMA 20 KEER SAUEN:

```
FOR X=1 TO 20:SAVE "TEST" :NEXT
```

DE RECORDER IN OPNAME ,TELKENS SPACE-BAR DRUKKEN EN ONZE TEST-TAPE IS KLAAR.

MET "CHECK" OF "LOAD" KUNNEN WE NU EENVOUDIG DE JUISTE INSTELLING VAN VOLUME EN TOON VASTSTELLEN.

WE KUNNEN OOK 20 KEER CALLM hexD2B8 DOEN, DAAR ZIT NAMELIJK DE ROUTINE DIE HET HERKENNINGSSIGNAAL PRODUCEERT.  
(ZIE HANDBOEK P.136 02C5 WOPEN)

### SOUND-MONITOR

BIJ DE MEESTE TV'S IS HET MOEILIJK OM GELUID EN BEELD VAN DAIPc GELIJKTIJDIG OPTIMAAL TE TUNEN. WE KRIJGEN EEN PRACHTIG GELUID ALS WE ONZE DAIPc AANSLUITEN AAN DE STEREOSET.

TWEE KLEINE VERSTERKERTJES GEBRUIKEN IS NATUURLIJK OOK EEN GOEDE OPLOSSING. IN HET JULI/AUG 80 NUMMER VAN ELEKTUUR WORDT EEN EENVOUDIGE CONSTRUCTIE BESCHREVEN. HET ONTWERP IS GEBASEERD OP IC LM386 ,EEN PRINTONTWERP (5cm X 5cm) IS BIJGEVOEGD. ELEKTUUR JULI/AUG 80 P. 8-32 Nr 100: UNIVERSELE LUIDSPREKER-UNIT.

VAL(xxxx)----STR\$(xxxx)

VAL( ) EN STR\$( ) ZIJN TWEE COMPLEMENTAIRE FUNCTIES. TOCH KUNNEN ER RARE DINGEN GEBEUREN ALS WE ZE SAMEN GEBRUIKEN. PROBEERT U MAAR:

```
A=1234 :A$=STR$(A) : PRINT VAL(A$)
```

WE KRIJGEN NIET DE OORSPRONKELIJKE WAARDE VAN A MAAR INVALID NUMBER !! OORZAAK VAN DIT MISVERSTAND IS DE SPATIE VOOR HET GETAL , DEZE SPATIE IS MEE OPGENOMEN IN A\$ EN DOET BASIC BESLUITEN DAT DE WAARDE VAN A\$ ONMOGELIJK TE VINDEN IS.

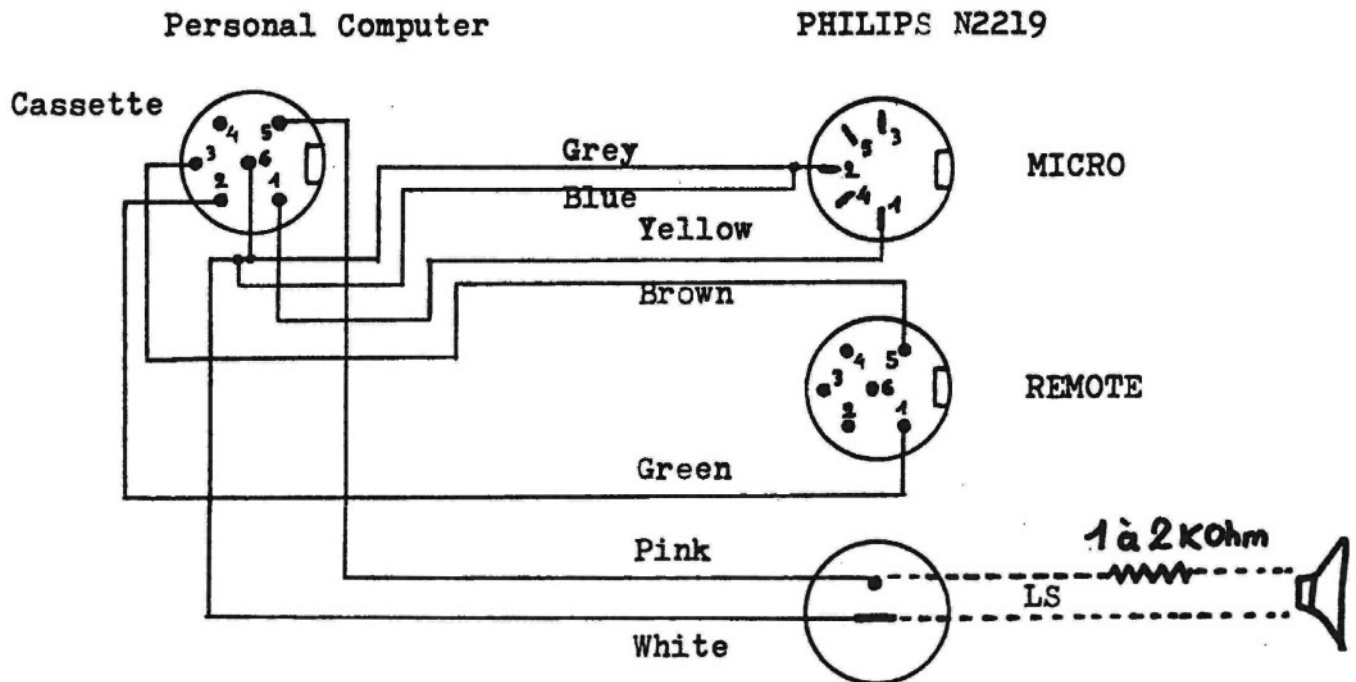
HET HANDBOEK GEEFT OP P. 101 EEN (ONLEESBARE)OPLOSSING VOOR DIT PROBLEEM:

```
10 INPUT A:PRINT
20 A$ = STR$(A)
30 A = VAL(RIGHT$(A$,LEN(A$)-1))
40 IF LEFT$(A$,1)="-" THEN A=A*(-1)
50 PRINT A: GOTO 10
```

DOOR DE FORMULERING OP LIJNNUMMER 30 KUNNEN WE DUS DE CONVERSIE-PROBLEMEN OMZEILEN.

# PEEK & POKE

## PERSONAL COMPUTER TO PHILIPS N2219 CASSETTE-CABLE



BOVENSTAAND SCHEMA GEEFT DE AANSLUITINGEN TUSSEN DAIPc EN PHILIPS N2219 (ONS STANDAARD-TOESTEL).  
 DE REMOTE-AANSLUITING VINDEN WE NIET ZO INTERESSANT OMDAT DEZE PLUG OOK HET SNELSPOELEN ONDERBREEKT.  
 DE AANSLUITING IN STIPPELLIJK WILLEN WE ZEKER AANRADEN: DEZE GEEFT DE MOGELIJKHEID OM TIJDENS DE OPNAME EN WEERGAVE MEE TE LUISTEREN, DIT WERKT ERG PRETTIG.  
 DE WAARDE VAN DE WEERSTAND IS NIET KRITISCH: HOE GROTER DE WEERSTAND, HOE KLEINER HET VOLUME VAN ONZE "MONITOR-LUIDSPREKER". VOOR DE LUIDSPREKER GEBRUIKTE WIJ EEN 60 Ohm TYPE, 0.2 WATT. DEZE INGRIEP KAN OOK GEBEUREN DOOR DE INWENDIGE LUIDSPREKER VIA EEN WEERSTAND MET DE LS-PLUG TE VERBINDEN, MAAR DAN MOET JE WEL HET CIRCUIT VAN DE RECORDER BESTUDEREN.  
 DE WEERSTAND NIET TE KLEIN KIEZEN WANT DAN GAAN WE HET SIGNAAL BELASTEN...  
 WIJ GEBRUIKEN DE RODE FERRO CASSETTES VAN PHILIPS, BIJ VOLUMES TUSSEN 7 EN 10 LUKT HET LADEN GEGARANDEERD...  
 DE PLUGGEN VOOR DAIPc ZIJN O. A. TE VERKRIJGEN IN DE TANDY-WINKELS.

```

1  MODE 0
2  COLORT 10 0 0 0
3  DIM A!(6.0)
10 PRINT CHR$(12):PRINT
20 PRINT "          ***** R A K E T S P E L *****":PRINT
30 PRINT "          === H. Bakker / DAI-Computer ===":PRINT :PRINT
40 PRINT " Probeer met 6 raketten zoveel mogelijk vliestuizen te"
45 PRINT " raken. Lanceer de raket door het erbij behorende cijfer"
50 PRINT " in te typen, als er geen raket in de lucht is.":PRINT
51 PRINT " Het resultaat wordt onder in het beeld bijgehouden. "
52 PRINT :PRINT " LET OP....De raket vliet niet altijd even hard....!"
53 PRINT :PRINT " Let daarom op de te verwachten MACH-snelheid..!"
54 PRINT " Deze snelheid verschijnt rechtsonder in het beeld. "
55 PRINT :PRINT " (snelheid vliestuis = MACH 0,5)"
57 PRINT CHR$(13):PRINT TAB(10):"Zullen we dan maar ?"
58 C!=GETC:IF C!=0.0 THEN 58
59 PRINT :PRINT TAB(10):"***** SUCCES *****!":WAIT TIME 40
60 E!=INT(RND(1.0)*3.0)+1.0
65 X!=0.0:FOR Q!=0.0 TO 6.0:A!(Q!)=0.0:NEXT
70 COLORT 0 14 0 0
80 PRINT CHR$(12):MODE 1A
90 D!=0.0:X!=0.0:F!=0.0
100 FOR A!=10.0 TO 60.0 STEP 10.0
110 DRAW A!,2 A!,5 15:DRAW A!-1,1 A!+1,1 10
120 PRINT TAB(A!-A!/4+1):A!/10.0::NEXT A!
130 DRAW 1,0 70,0 6:PRINT
140 PRINT " Aantal afgeschoten raketten:";F!

```



```

150 PRINT " Aantal geraakte oliestuizen:";D!
155 CURSOR 40,2:PRINT "Raket snelheid:"
160 CURSOR 40,1:PRINT "MACH. ";E!
170 GOSUB 1000
200 C!=GETC:IF C!=0.0 THEN WAIT TIME 14:GOSUB 1000
210 IF C!=0.0 THEN 200
211 IF C!>54.0 OR C!<49.0 THEN 200
220 C1!=10.0*(C!-48.0):Q!=C1!/10.0
222 IF A!(Q!)<>0.0 THEN 200
224 A!(Q!)=1.0
230 F!=F!+1.0
240 B!=2.0
300 FOR I!=1.0 TO E!*2.0:B!=B!+1.0
310 DRAW C1!,B!-2 C1!,B!+3 15:DRAW C1!-1,B!-1 C1!+1,B!-1 15
320 DRAW C1!-1,B!-2 C1!+1,B!-2 0:IF B!<40 THEN 342
330 IF B!>40.0 AND B!<50.0 AND X!<C1! AND X!+6>C1! THEN 500
342 DOT C1!,B!-1 0
350 NEXT I!:GOSUB 1000
360 IF B!>54 THEN 700
370 GOTO 300
500 P!=5.0:FOR T!=0.0 TO 2.0:FILL X!+P!,55 X!,40 15
510 FILL X!+P!,55 X!,35 0:P!=P!+3.0:NEXT T!
520 X!=0.0:D!=D!+1.0:GOTO 700

```

```

700 E!=(INT(RND(1.0)*3.0))+1.0:CURSOR 30,2:PRINT F!
720 CURSOR 30,1:PRINT D!
725 CURSOR 45,1:PRINT E!
730 IF F!=6.0 THEN INPUT "*** NOG EEN KEER *** (J/N)";D$:GOTO 770
740 GOSUB 1000
760 GOTO 200
770 IF D$="J" THEN 60:PRINT " ";:END
1000 X!=X!+1.0
1010 DRAW X!+2,45 X!+6,45 15
1020 DRAW X!+2,46 X!+5,46 15:DRAW X!+2,44 X!+5,44 15
1040 DRAW X!+3,47 X!+4,47 15:DRAW X!+3,43 X!+4,43 15
1060 DOT X!+3,44 0:DOT X!+3,46 0:DOT X!+3,47 0:DOT X!+3,43 0
1080 DRAW X!+1,43 X!+1,47 0
1090 IF X!=64.0 THEN X!=0.0:E!=INT(RND(1.0)*3.0)+1.0:FILL 70,48 62,43 0:CURSOR
45,1:PRINT E!
1100 RETURN

```

GRAFIEK SK M. VERMEULEN

```

10 MODE 3A
20 FILL 0,0 XMAX,YMAX 10
30 FOR A=0.0 TO 80.0 STEP 3.0
40 DRAW 80,0 A,100 0
50 DRAW 0,50 XMAX,100-A 0
60 DRAW 80,100 A,0 0
70 DRAW XMAX,50 0,A 0
80 NEXT
90 END

```

# PEEK & POKE

## EEN OVERZICHT VAN DE VIDEORAM IN MODE 0 (CHARACTER MODE)

Deze tabel geeft respectievelijk lijnummer, color code byte van het eerste character van een lijn, informatiebyte van het eerste character en ook deze locaties voor het laatste character van een lijn. U kan met peek en poke deze organisatie verkennen, zolang de controlebytes van de lijnen onveranderd blijven, is alles o.k. (De lijn-controle bytes zitten nog voor de informatie voor het eerste character : voor een 48 K is de eerste controlebyte BFEF, voor een 32 K 7FEF. De color byte van de eerste lijn zit respectievelijk op BFEE en 7FEE).

Volgend programma drukt voor U deze control en colorbytes :

```
10 FOR A=0 TO 23:PRINT A+1,:PRINT HEX$(#FEF - #86*A),
20 PRINT HEX$(#FEE - #86*A):NEXT
   #BFEF voor 48 K ,#7FEF voor 32 K ,#1FEF voor 8 K
```

LINE NUMBER	# LOCATION COLOR CODE BEGIN LINE	# LOCATION BEGIN LINE	# LOCATION COLOR CODE END LINE	# LOCATION END LINE
23.0	# BFEA	# BFED	# BF6A	# BF6D
22.0	# BF64	# BF67	# BEE4	# BEE7
21.0	# BEDE	# BEE1	# BE5E	# BE61
20.0	# BE58	# BE5B	# BDD8	# BDDB
19.0	# BDD2	# BDD5	# BD52	# BD55
18.0	# BD4C	# BD4F	# BCCC	# BCCF
17.0	# BCC6	# BCC9	# BC46	# BC49
16.0	# BC40	# BC43	# BBC0	# BBC3
15.0	# BBBA	# BBBD	# BB3A	# BB3D
14.0	# BB34	# BB37	# BAB4	# BAB7
13.0	# BAAE	# BAB1	# BA2E	# BA31
12.0	# BA28	# BA2B	# B9A8	# B9AB

LINE NUMBER	# LOCATION COLOR CODE BEGIN LINE	# LOCATION BEGIN LINE	# LOCATION COLOR CODE END LINE	# LOCATION END LINE
11.0	# B9A2	# B9A5	# B922	# B925
10.0	# B91C	# B91F	# B89C	# B89F
9.0	# B896	# B899	# B816	# B819
8.0	# B810	# B813	# B790	# B793
7.0	# B78A	# B78D	# B70A	# B70D
6.0	# B704	# B707	# B684	# B687
5.0	# B67E	# B681	# B5FE	# B601
4.0	# B5F8	# B5FB	# B578	# B57B
3.0	# B572	# B575	# B4F2	# B4F5
2.0	# B4EC	# B4EF	# B46C	# B46F
1.0	# B466	# B469	# B3E6	# B3E9
0.0	# B3E0	# B3E3	# B360	# B363

```
5 CLEAR 1000
6 COLORG 0 0 0 0
10 MODE 6A
15 PRINT "BUSY CALCULATING"
20 HX!=XMAX/2.0:HY!=YMAX/2.0
30 S!=SQR(0.4*YMAX)
31 GOSUB 1000
35 QQ=0
43 FOR R=0 TO 251 STEP 6
45 Z!=ZZ!(R):QQ=QQ+2
48 FOR AW=0 TO 8:W=AW*112:COLR=(AW MOD 3)+21:PP=QQ+W:GOSUB 2000:PP=1176-QQ+W:
GOSUB 2000:NEXT: NEXT
49 MODE 6
50 FOR TT=30 TO 1 STEP -1
70 COLORG 0 0 0 15:WAIT TIME TT:COLORG 0 0 15 0:WAIT TIME TT:COLORG 0 15 0 0:
WAIT TIME TT
75 IF TT=6 GOTO 70
80 NEXT:STOP
1000 DIM ZZ!(253.0)
1010 ST!=PI/504.0:STP!=-ST!
1020 FOR QQ=0 TO 252:STP!=STP!+ST!
1030 ZZ!(QQ)=S!*SIN(STP!)
1040 NEXT
1050 RETURN
2000 SS=(PP+1008) MOD 252
```



# LIST

## 4 COLOR DEMO (LISSAJOUS)

Dit programma illustreert duidelijk de mogelijkheden in 4-kleuren modes ( 2,4,6).Op lijnnummers 16-50 doen we een aardigheidje om het tekenen vlugger te laten verlopen : we berekenen eerst de coördinaten en stoppen ze in arrays A en B.

Op lijnnummers 100-120 voeren we de tekening uit.

Op lijnnummers 300-345 goochelen we een minuutje met de kleurenregisters van COLORG en van 400-430 laten we het lot(rnd) de effecten bepalen.

8K en 12K toestellen kunnen dit programma lopen in MODE 4 mits volgende aanpassingen:

```
5 CLEAR 2100
```

```
10 MODE 4
```

```
40 A(N)=XMAX/2 + 75*COS(X):B(N)=YMAX/2 + 50*SIN(X)
```

```
110 DRAW XMAX/22,YMAX A(X),B(X) 0
```

```
5 CLEAR 5000
```

```
10 MODE 6
```

```
16 DIM A(250.0),B(250.0)
```

```
20 COLORG 8 0 15 3
```

```
30 FOR X=0.0 TO 2.0*PI STEP 3E-2
```

```
40 A(N)=XMAX/2.0+100.0*COS(X):B(N)=YMAX/2.0+100.0*SIN(X*2.0)
```

```
45 N=N+1.0
```

```
50 NEXT
```

```
100 FOR X=0.0 TO 209.0
```

```
110 DRAW 150,125 A(X),B(X) 0
```

```
115 DRAW 0,0 A(X),B(X) 3
```

```
116 DRAW A(X),B(X) XMAX,0 15
```

```
120 NEXT
```

```
300 FOR X=0.0 TO 50.0
```

```
320 COLORG 0 A 0 0
```

```
330 WAIT TIME 15
```

```
335 COLORG 0 0 A 0
```

```
337 WAIT TIME 15
```

```
338 COLORG 0 0 0 A
```

```
339 WAIT TIME 15
```

```
340 A=A+1.0:IF A=16.0 THEN A=1.0
```

```
345 NEXT X
```

```
400 FOR X=0.0 TO 50.0
```

```
410 COLORG RND(15.0) RND(15.0) RND(15.0) RND(15.0)
```

```
420 WAIT TIME 20
```

```
430 NEXT X
```

## MERGING VAN BASIC-PROGRAMMA'S

Verschillende leden vroegen naar de methode om BASIC-programma's aan elkaar te plakken (merging). De oplossing is erg eenvoudig. De vereiste handelingen in de juiste volgorde zijn :

1. CLEAR 5000  
De omvang van clear hangt af van de lengte van uw programma's. Voor korte programma's zou clear 1000 ook voldoende kunnen zijn.
2. LOAD "eerste programma"
3. EDIT (wachten tot de edit-listing klaar is !)
4. Tweemaal BREAK
5. LOAD "tweede programma"
6. POKE #135 ,2 (haalt uw eerste programma uit de edit-buffer)

Uw beide programma's zijn nu samengesmolten. Er wel op letten dat beide programma's verschillende lijnnummers gebruiken.

## GRAFISCHE OPDRACHTEN BONDIG EN OVERZICHTELIJK...

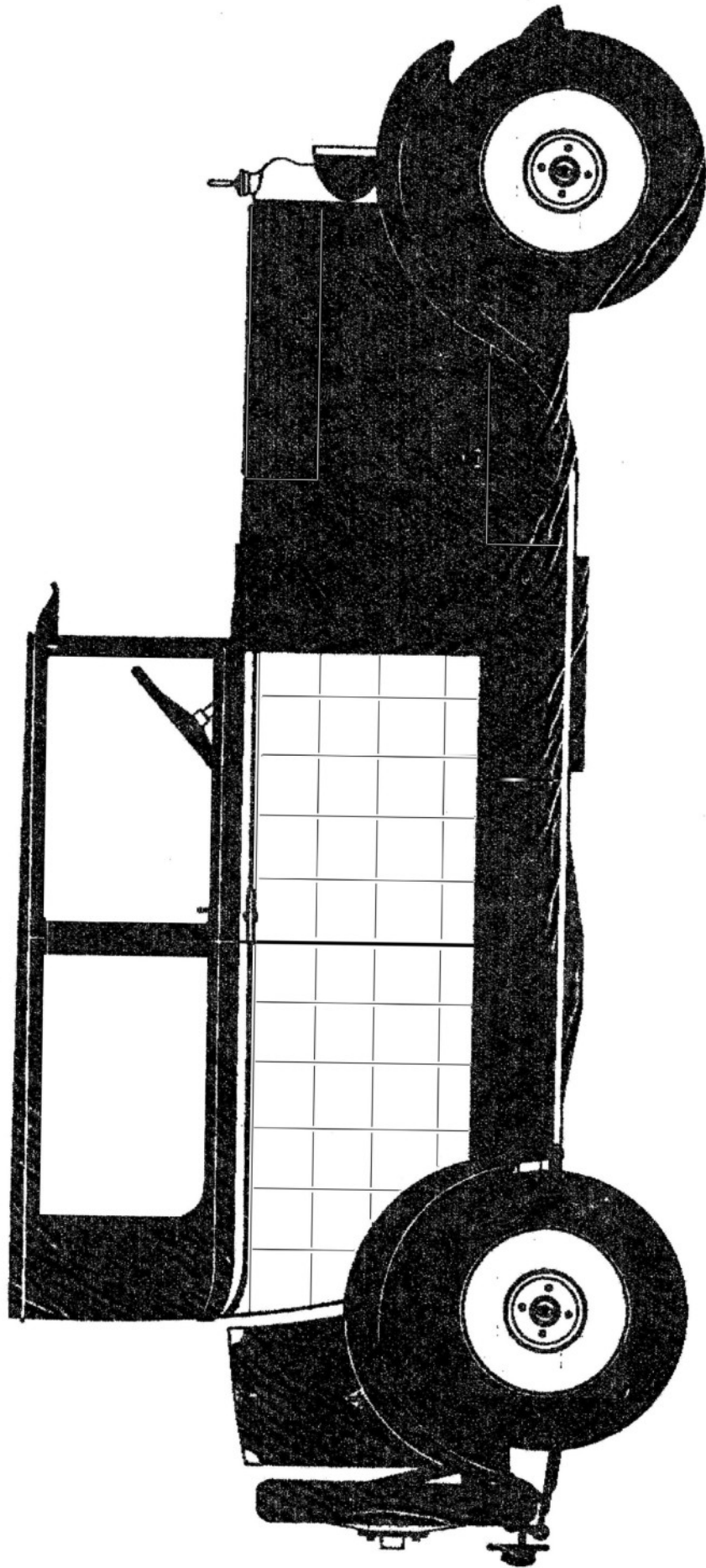
Een tekening samenstellen in een van de grafische modes kan erg omslachtig, langdradig en geheugen-gulzig zijn. Stelt U zich maar voor : 100 keer DRAW 123,65 267,54 14 , nog eens zoveel dots en fills en een aardig stukje van uw ram is gevuld met overbodige informatie. Wij stellen het gebruik van read en datastate-ments voor : deze houden het programma bondig, overzichtelijk en vooral geheugen-efficiënt !

Het hoofdprogramma zou er als volgt kunnen uitzien :

```

100 REM data inlezen en opdrachten uitvoeren
110 READ A$:IF LEN(A$)>1 THEN READ A$
deze formulering laat toe in onze data-tabellen titels van tekeningen te plaatsen
120 IF A$="STOP" THEN GOTO .....
130 IF A$="O" THEN READ A,B,K:DOT A,B K :GOTO 110
140 IF A$="A" THEN READ A,B,C,D,K :DRAW A,B C,D K:GOTO 110
150 IF A$="I" THEN READ A,B,C,D,K :FILL A,B C,D K:GOTO 110
de data-tabel :
500 DATA HUIS,A,10,10,10,10,3,I,10,10,10,11,6,.....
           (draw)           (fill)           (color)

```





# LIST

```
5 CLEAR 4000: DIM A(255.0), B(255.0)
10 COLORG 8 0 14 15
15 FOR X=0.0 TO 2.0*PI STEP 2.5E-2
16 A(N)=COS(X): B(N)=SIN(X)
17 N=N+1.0: NEXT
18 PRINT N
20 MODE 6
100 REM FILL
110 FOR X=1.0 TO 8.0
120 READ A, B, C, D, CO
130 FILL A, B, C, D, CO
140 NEXT
150 DATA 69, 92, 178, 128, 0, 85, 98, 126, 124, 8, 132, 98, 174, 124, 8, 69, 40, 248, 92,
0
160 DATA 69, 55, 176, 92, 14, 248, 70, 254, 92, 14, 46, 46, 64, 90, 0, 36, 44, 43, 88, 0
200 REM wilen
210 FOR X=0.0 TO 128.0
220 DRAW 71, 43 71+25*A(X), 43+30*B(X) 15
225 DRAW 254, 43 254+25*A(X), 43+30*B(X) 15
230 NEXT
250 FOR X=0.0 TO 255.0
260 DRAW 71, 43 71+22*A(X), 43+22*B(X) 0
270 DRAW 254, 43 254+22*A(X), 43+22*B(X) 0
272 DOT 71+17*A(X), 43+17*B(X) 8: DOT 254+17*A(X), 43+17*B(X) 8
273 DRAW 71, 43 71+13*A(X), 43+13*B(X) 14: DRAW 254, 43 254+13*A(X), 43+13*B
(X) 14
274 DRAW 71, 43 71+5*A(X), 43+5*B(X) 15: DRAW 254, 43 254+5*A(X), 43+5*B(X)
15
280 NEXT
281 FOR X=64.0 TO 192.0: DRAW 256, 78 256+6*A(X), 78+5*B(X) 0: NEXT
285 FILL 110, 37 140, 39 0: FILL 158, 37 192, 39 0
290 FOR X=122.0 TO 127.0: DRAW 178, X 187, 122 0: NEXT
292 FOR X=210.0 TO 226.0 STEP 3.0: DRAW X, 62 X, 74 15: NEXT
293 FOR X=129.0 TO 131.0: DRAW 68, X 174, 127 0: NEXT
300 REM draw
305 DOT 162, 43 15: DOT 189, 43 15
310 N=24.0
320 FOR X=1.0 TO N
330 READ A, B, C, D, CO
340 DRAW A, B, C, D, CO
350 NEXT
360 DRAW 130, 89 138, 69 0
365 FILL 250, 92 253, 94 0
370 DOT 36, 44 8: DOT 43, 88 8: DOT 36, 88 8: DOT 43, 44 8
375 FILL 32, 65 34, 69 0
400 DATA 69, 127, 178, 127, 15, 129, 93, 129, 127, 15, 129, 55, 129, 93, 0
410 DATA 70, 90, 176, 90, 0, 70, 89, 176, 89, 15, 70, 88, 176, 88, 0
420 DATA 204, 54, 204, 92, 15, 204, 81, 248, 81, 15, 194, 54, 230, 54, 15
430 DATA 156, 42, 156, 54, 15, 194, 42, 194, 54, 15, 176, 96, 176, 124, 15
```

*Citroën B 14 Coach - 1927*

# LIST

```
440 DATA 36,77,34,72,0,36,54,34,61,0,34,72,34,61,0
450 DATA 160,97,172,110,0,161,97,173,110,0,164,102,167,97,0
460 DATA 35,43,49,41,0,46,87,49,90,15,61,90,64,87,15
470 DATA 33,46,36,46,0,33,41,33,52,0,252,94,252,97,14
472 GOTO 500:REM DELETE THIS LINE FOR HARDCOPY
475 DIM CC(15,0):CC(8,0)=32,0:CC(0,0)=127,0:CC(14,0)=49,0:CC(15,0)=46,0

479 MODE 6A
480 FOR Y=15,0 TO 140,0:FOR X=25,0 TO 104,0:PRINT CHR$(CC(SCRN(X,Y))):
NEXT:PRINT :NEXT
481 FOR X=0,0 TO 10,0:PRINT :NEXT
482 FOR Y=15,0 TO 140,0:FOR X=105,0 TO 184,0:PRINT CHR$(CC(SCRN(X,Y))),
:NEXT:PRINT :NEXT
483 FOR X=0,0 TO 10,0:PRINT :NEXT
484 FOR Y=15,0 TO 140,0:FOR X=185,0 TO 264,0:PRINT CHR$(CC(SCRN(X,Y))):
:NEXT:PRINT :NEXT
485 FOR X=0,0 TO 10,0:PRINT :NEXT
486 FOR Y=15,0 TO 140,0:FOR X=265,0 TO 335,0:PRINT CHR$(CC(SCRN(X,Y))):
:NEXT:PRINT :NEXT
500 FOR Y=20,0 TO 130,0 STEP 2,0:FOR X=30,0 TO 280,0 STEP 3,0:A=SCRN(X,
Y):DOT XMAX-X,115+Y A:NEXT:NEXT
600 FILL 0,0 280,133 0
610 FOR Y=135,0 TO 245,0 STEP 2,0:FOR X=30,0 TO 280,0 STEP 3,0:A=SCRN(X
MAX-X,Y):DOT (XMAX-X)/3,Y-115 A:NEXT:NEXT
611 FOR Y=135,0 TO 245,0 STEP 2,0:FOR X=30,0 TO 280,0 STEP 3,0:A=SCRN(X
MAX-X,Y):DOT (XMAX-X)/3+100,(Y-115)/2 A:NEXT:NEXT
615 FOR Y=135,0 TO 245,0 STEP 2,0:FOR X=30,0 TO 280,0 STEP 3,0:A=SCRN(X
MAX-X,Y):DOT XMAX-(XMAX-X)/3,Y-115 A:NEXT:NEXT
620 FOR Y=135,0 TO 245,0 STEP 2,0:FOR X=30,0 TO 280,0 STEP 3,0:A=SCRN(X
MAX-X,Y):DOT XMAX-((XMAX-X)/3+100),160-((Y-115)/2) A:NEXT:NEXT
1000 GOTO 1000
```

```
110 E=#FF
115 COLORT 0 9 9 0
120 B=#BFEF
125 FOR A=0 TO 23
130 D=B-3
135 FOR C=0 TO 65
140 POKE D,E
145 D=D-2:NEXT
146 RJ=GETC:IF RJ=32 GOTO 20
155 B=B-#86:NEXT
165 E=INOT E%IAND #FF
170 IF GETC=0 GOTO 170
```

In onze rubriek boek en tijdschriften komt als eerste aan de beurt een werk uit de tandy-winkel :

"HOW TO PROGRAM MICROCOMPUTERS" door W. Barden

Het boek beoogt een gids te zijn bij het programmeren met de intel 8080 (micro in uw DAipc), motorola 6800 en mos tech. 6502. Vier grote delen in het boek :

1. Een introductie in de computekunde : microprocessoren, getalverwerking, basisbegrippen en data codes.

2. Werking en architectuur van de drie vermelde microprocessoren: adresseringsmethodes, geheugentypes, stack-concepten, instructiesets, I/O en interrupt technieken.

3. Assembleertalen en programmeringstechnieken.

4. Een aantal belangrijke subroutines, telkens voor de drie processoren :

- simple compare routine
- double-precision shift
- timing routine
- single-precision multiply
- single-precision divide
- multiple-precision add
- multiple-precision subtract
- ascii to binary en omgekeerd
- ascii decimal to binary en omgekeerd
- ascii hex to binary en omgekeerd
- moving data
- fill data
- compare string
- search table
- randum number generator

Van harte aanbevolen voor wie eens direct met zijn 8080 wil gaan converseren. De informatie over de andere micro's is dan misschien wel overbodig, maar geeft wel een ideale gelegenheid om de voor- en nadelen van deze micro's uit te pluizen.

De vermelde subroutines zijn eenvoudig in te voeren via utility of assembler. Er wel aan denken dat basic niet zelf push en pop van de registers verzorgt !

Te verkrijgen in alle tandy-winkels. Prijs : 275 F. Als er belangstelling is willen we graag een paar van deze subroutines onder de DAInamic-loupe nemen...

---

## Beknopte beschrijving van enkele programma's uit onze bibliotheek

### 1. TOWERS OF HANOI \*\* DAInamic

Het oude oosterse spel nu op uw DAipc met kleurengrafieken van mode 3A. Opdracht : 7 (of minder) schijven verplaatsen volgens deze regels : 1 schijf per keer verplaatsen, nooit een grotere schijf op een kleinere plaatsen. Een knappe prestatie als U het klaarspeelt met 7 schijven !

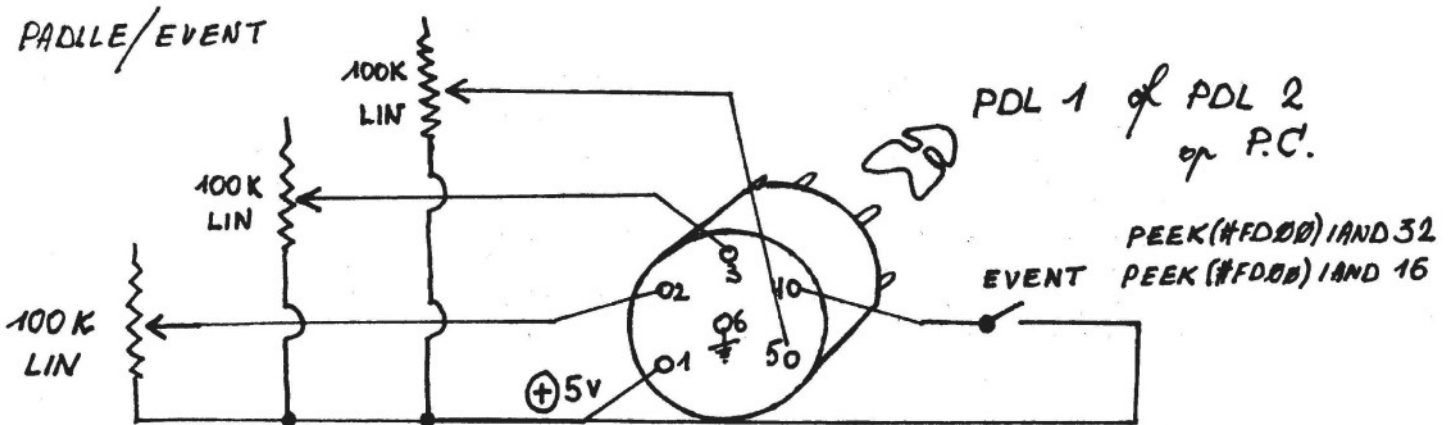
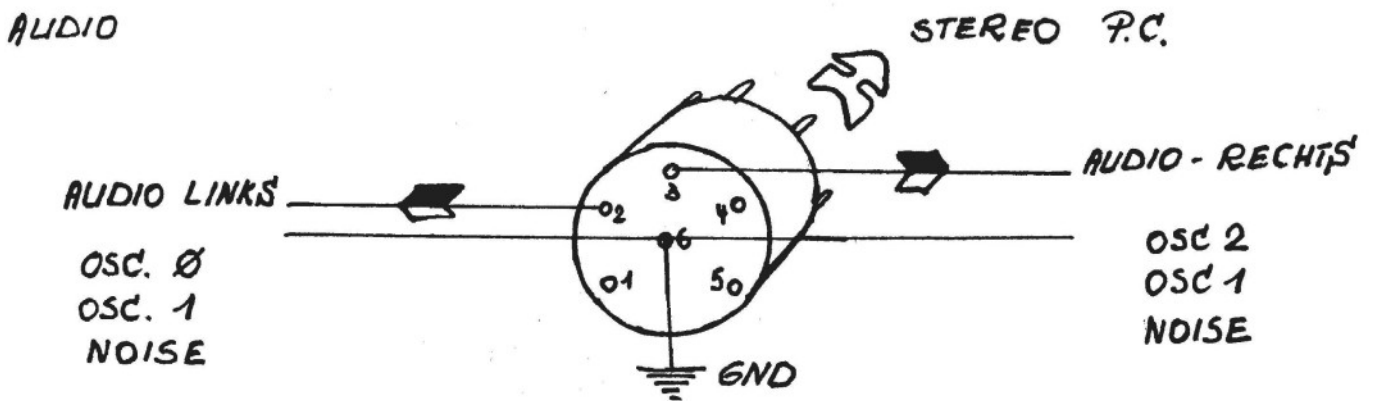
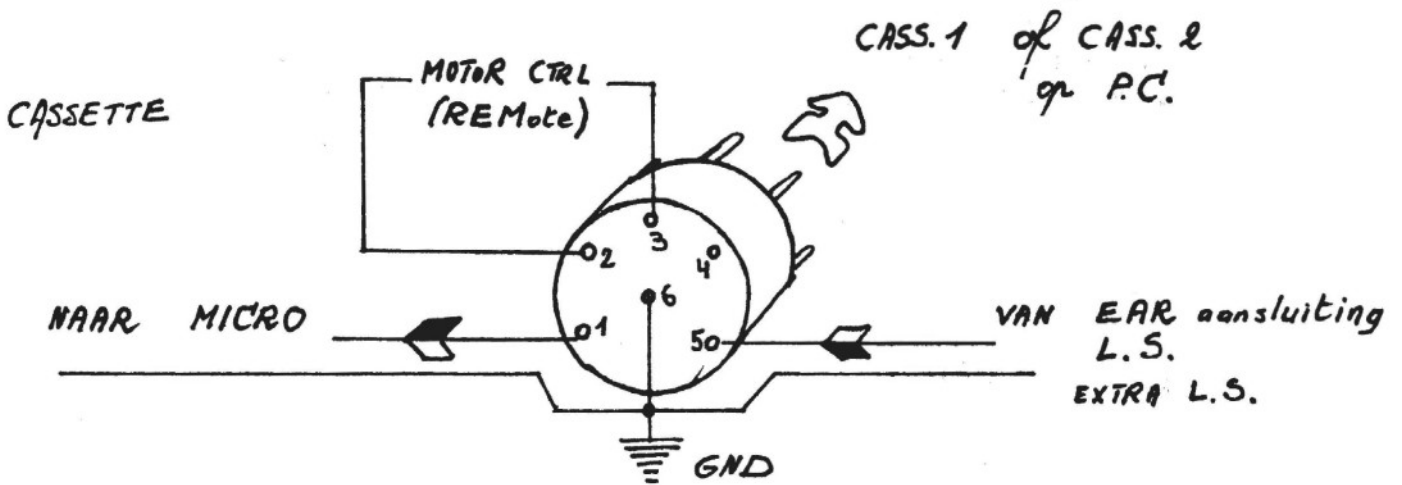
### 2. SIMON \*\* DAInamic

Het programma speelt een melodie terwijl op het scherm gekleurde vlakken verschijnen. U moet trachten de melodie telkens na te spelen met de cursor-toetsen. De melodie wordt wel steeds langer. Knappe koppen halen 20 stappen, terwijl het programma rustig kan doorgaan tot 250 !

### 3. A MAZE ...AMAZING \*\* DAInamic

In grafische mode wordt volgens uw keuze een doolhof getekend, waaruit U kan ontsnappen met de cursor-toetsen. Naast de doolhof ziet U de tijd evalueren. Als U over een printer beschikt, krijgt U telkens nog een mooie hardcopy.

# PEEK & POKE



# PEEK & POKE

## HET GEBRUIK VAN DE PADDLE-EVENT INGANGEN

De stand van de potentiometers van de paddles is eenvoudig te bekomen door de functie PDL(X).

Om de stand van de 2 drukknoppen te bepalen is echter geen speciale functie voorzien. De signalen van de drukknoppen worden ingevoerd op bits 4 en 5 van poort met adres =FD $\theta\theta$  : EV1 = bit 4 van poort #FD $\theta\theta$   
EV2 = bit 5 van poort #FD $\theta\theta$

In de overige bits van poort =FD $\theta\theta$  zijn we niet geïnteresseerd en we zullen deze dan ook weg maskeren (bv met IAND).

Uitgewerkt in BASIC : EV1 = PEEK(#FD $\theta\theta$ ) IAND #1 $\theta$   
EV2 = PEEK(#FD $\theta\theta$ ) IAND #2 $\theta$

De waarden die EV1 en EV2 aannemen :

geen drukknop ingedrukt : EV1 =  $\theta$  en EV2 =  $\theta$   
drukknop 1 ingedrukt : EV1 = #1 $\theta$   
drukknop 2 ingedrukt : EV2 = #2 $\theta$

Vermits =1 $\theta$  en =2 $\theta$  niet zo praktisch is voor verdere berekeningen kunnen we een shift right operator toevoegen, zodat EV1 en EV2 enkel  $\theta$  of 1 kunnen worden.

EV1 = ( PEEK( #FD $\theta\theta$  ) IAND #1 $\theta$  ) SHR 4  
EV2 = ( PEEK( #FD $\theta\theta$  ) IAND #2 $\theta$  ) SHR 5

Het bovenstaande is toepasbaar als we in een programma EV1 en EV2 willen bepalen voor gebruik in bv een berekening of een beslissing.

Indien we in een programma willen wachten tot een drukknop ingedrukt wordt, dan kunnen we de WAIT MEM adres, masker, invert functie gebruiken :

wachten tot 1 ingedrukt is : WAIT MEM #FD $\theta\theta$ ,#1 $\theta$   
wachten tot 2 ingedrukt is : WAIT MEM #FD $\theta\theta$ ,#2 $\theta$   
wachten tot beide ingedrukt zijn : WAIT MEM #FD $\theta\theta$ ,#3 $\theta$

Op een volgende lijn bijvoorbeeld kunnen we wachten tot de drukknop terug wordt losgelaten :

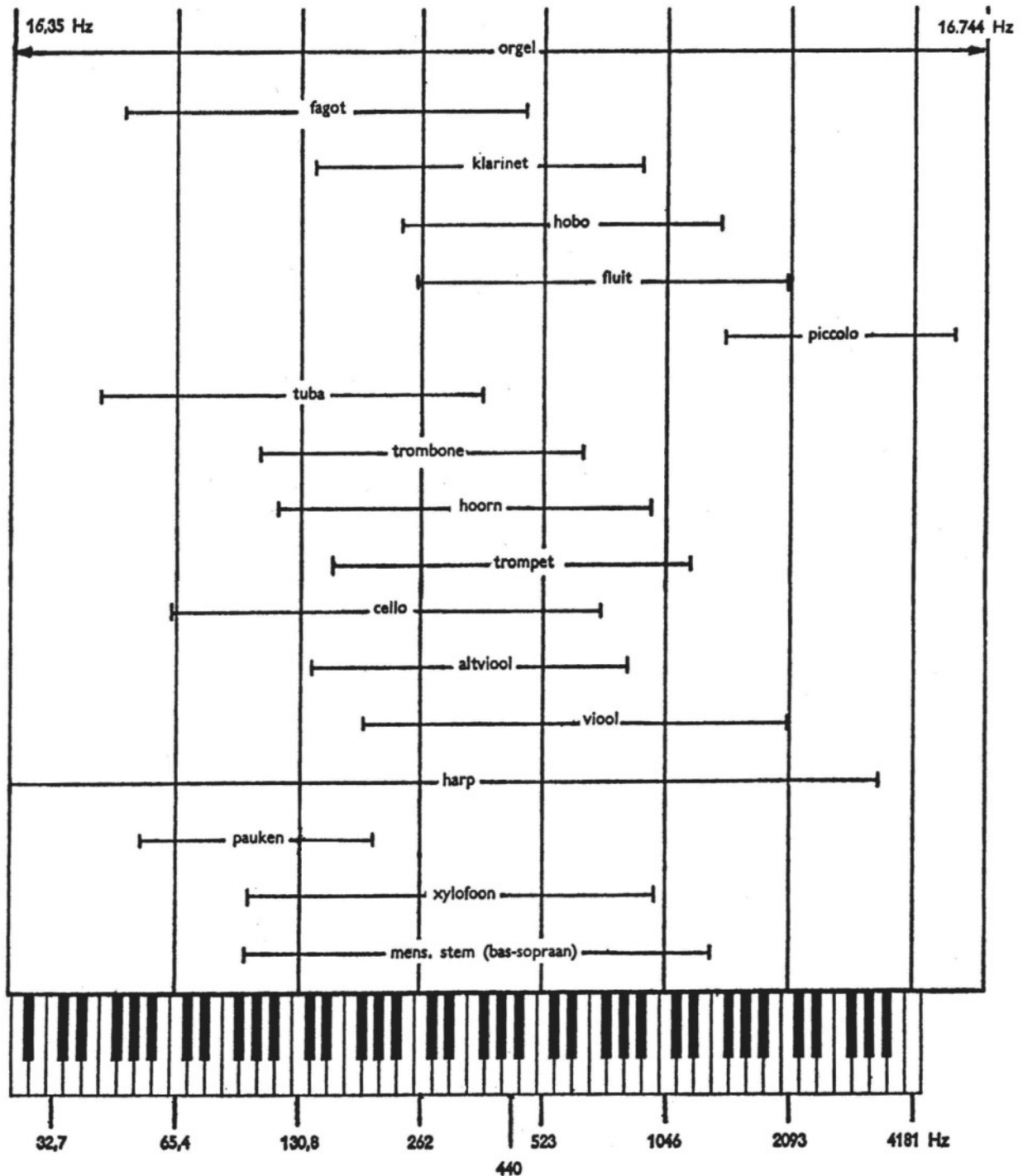
voor drukknop 1 : WAIT MEM #FD $\theta\theta$ ,#1 $\theta$ ,#1 $\theta$   
voor drukknop 2 : WAIT MEM #FD $\theta\theta$ ,#2 $\theta$ ,#2 $\theta$   
voor beide drukknoppen : WAIT MEM #FD $\theta\theta$ ,#3 $\theta$ ,#3 $\theta$

tip.....tip.....tip.....tip.....tip.....tip.....tip.....tip.....tip.....tip...

## EEN EENLIJN CIRKEL VOOR ALLE MODES

FOR X= $\theta$  TO 2\*PI STEP  $\theta.\theta$ 1 : DOT XMAX/2 +YMAX/2 \* SIN(X) , YMAX/2 + YMAX/2 \*  
COS(X) COLOR : NEXT X

# SOUND



FREKWENTIEBEREIK VAN VERSCHILLENDE MUZIEKINSTRUMENTEN  
uit ELECTRONISCHE ORGELS door WIM VAN BUSSEL Prisma 1540

# SOUND

OP BIJGAAND OVERZICHT KUNNEN WE VASTSTELLEN DAT HET FREKVENTIE-  
BEREIK VAN DE MEESTE MUZIEKINSTRUMENTEN TAMELIJK BEPERKT IS.  
VOOR ONZE DAIPc Zouden WE EEN LIJN KUNNEN TREKKEN VAN 31 Hz  
TOT 65535 Hz, HET GEDEELTE BOVEN 16000 IS DAN WEL ENKEL BESTEMD  
VOOR HONDEN EN VLEERMUIZEN...

OP HET TOETSENBORD VINDEN WE DE CENTRALE LA (440 Hz) EN 8 KEER  
EEN TOETS "DO" (32.7 65.4 130.8 262 523 1046 2093 4181)  
DIT ZIJN DE NOTEN "DO" UIT 8 OKTAVEN. EEN HOGER OKTAAF KRIJGEN  
WE DOOR ALLE FREKVENTIES MET 2 TE VERMENIGVULDIGEN.

HET VERSCHIL  
TUSSEN 32.7 Hz EN 33 Hz IS ALLEEN HOORBAAR DOOR EEN ZWEUING  
ALS DE BEIDE FREKVENTIES SAMEN KLINKEN.

LAAT UW DAIPc OOK EENS ZWEVEN:

```
ENVELOPE 0 15 (zonder lijnnummers... command mode)
SOUND 0 0 15 0 FREQ(500)
SOUND 1 0 15 0 FREQ(501)
SOUND 2 0 15 0 FREQ(502)
```

VOOR WE MET HET SERIEUZE MUZIEKWERK BEGINNEN EEN PAAR SUBROUTINES  
OM ONZE PROGRAMMA'S (VOORAL DE GAMES) ATTRACTIEF TE MAKEN:

EEN HOEKIG SIREENTJE:

```
-----
XXXX      ENVELOPE 0 15
XXXX      FOR X = 1 TO .....
XXXX      SOUND 1 0 15 0 FREQ(800):WAIT TIME 15
XXXX      SOUND 1 0 15 0 FREQ(600):WAIT TIME 15
XXXX      NEXT : SOUND OFF : RETURN
```

EEN VLOEIEND SIREENTJE:

```
-----
XXXX      ENVELOPE 0 15
XXXX      FOR X = 1 TO .....
XXXX      SOUND 1 0 15 2 FREQ(800):WAIT TIME 15
XXXX      SOUND 1 0 15 2 FREQ(600):WAIT TIME 15
XXXX      NEXT : SOUND OFF : RETURN
```

"AMERICAN STYLE...":

```
-----
XXXX      ENVELOPE 0 15
XXXX      FOR X = 1 TO .....
XXXX      SOUND 1 0 15 0 FREQ(2000)
XXXX      SOUND 1 0 15 2 FREQ(600):WAIT TIME 20
XXXX      NEXT : SOUND OFF : RETURN
```

EXPERIMENTEREN MET FREKVENTIES EN WAIT TIME GEEFT ONBEPERKTE  
MOGELJKHEDEN, BEZORG ONS UW BESTE RESULTATEN VOOR PUBLICATIE..  
VOLGENDE KEER BESTUDEREN WE EEN EENVOUDIGE MUZIEKPARTITUUR EN

GAAN WE HET MUZIEKSTUKJE PROGRAMMEREN.....

# LIST

BLOXY COLORG-DEMO 8...32K MODE 2...6

```
100  MODE 4:COLORG 0 0 0 0:FOR X=5 TO XMAX-10 STEP 12:FOR Y=5 TO YMAX-10 STEP 1
2:FOR Z=0 TO 2
230  C=INT(RND(4.0))+20.0:DRAW X+Z*2,Y+Z*2 X+10-Z*2,Y+Z*2 C
240  DRAW X+Z*2,Y+Z*2 X+Z*2,Y+10-Z*2 C
260  DRAW X+Z*2,Y+10-Z*2 X+10-Z*2,Y+10-Z*2 C
265  DRAW X+10-Z*2,Y+Z*2 X+10-Z*2,Y+10-Z*2 C
270  NEXT:NEXT:NEXT
275  FOR X!=1.0 TO 50.0
280  COLORG 0 RND(16.0) RND(16.0) RND(16.0):WAIT TIME 25:NEXT
300  FOR Y!=1.0 TO 50.0
310  C!=RND(10.0)+6.0
320  COLORG 0 C! 0 0:WAIT TIME 10
330  COLORG 0 0 C! 0:WAIT TIME 10
340  COLORG 0 0 0 C!:WAIT TIME 10:NEXT
900  FOR X!=1.0 TO 30.0
1000 COLORG 0 1 9 12:WAIT TIME 10
1010 COLORG 0 5 13 15:WAIT TIME 10
1020 COLORG 0 3 4 11:WAIT TIME 10
1030 NEXT:GOTO 275
```

DE TEKENOPDRACHTEN WORDEN BEREKEND IN FUNCTIE VAN XMAX EN YMAX,  
ZODAT HET PROGRAMMA GESCHIKT IS VOOR ALLE 4 KLEUREN-MODES.  
VOOR 8K TOESTELLEN VERVANGEN WE OP LIJNNUMMER 100 "MODE 4" DOOR "MODE 2".  
32K TOESTELLEN KUNNEN HET PROGRAMMA GEBRUIKEN IN HOOGSTE RESOLUTIE  
DOOR "MODE 4" TE VERVANGEN DOOR "MODE 6".  
DIT PROGRAMMA MAAKT GEBRUIK VAN DE KLEUREN 20-23.  
OP LIJNNUMMER 230 WORDT RANDOM EEN KLEUR 20...23 GEKOZEN.  
20 KIEST DE KLEUR VAN REGISTER 1, 21 VAN REGISTER 2, 22 VAN REGISTER 3  
EN 23 VAN REGISTER 4. DAAR DE KLEURREGISTERS OORSPRONKELIJK OP  
0 STAAN ZIEN WE DE OPBOUW VAN DE TEKENING NIET.



# PEEK & POKE

INSTRUCTION	FUNCTION	ALIA	INSTRUCTION	FUNCTION	HEX
<b>MOVE GROUP</b>			<b>JUMP GROUP</b>		
MOV A, reg	(A) ← (reg)	reg A B C D E H L M	JMP addr	(PC) ← addr	C3 al ah
MOV B, reg	(B) ← (reg)	7F 78 79 7A 7B 7C 7D 7E	JNZ addr	If Z=0, (PC) ← addr	C2 al ah
MOV C, reg	(C) ← (reg)	4F 48 49 4A 4B 4C 4D 4E	JZ addr	If Z=1, (PC) ← addr	CA al ah
MOV D, reg	(D) ← (reg)	57 50 51 52 53 54 55 56	JNC addr	If CY=0, (PC) ← addr	D2 al ah
MOV E, reg	(E) ← (reg)	5F 58 59 5A 5B 5C 5D 5E	JC addr	If CY=1, (PC) ← addr	DA al ah
MOV H, reg	(H) ← (reg)	67 60 61 62 63 64 65 66	JPO addr	If P=0, (PC) ← addr	E2 al ah
MOV L, reg	(L) ← (reg)	6F 68 69 6A 6B 6C 6D 6E	JPN addr	If P=1, (PC) ← addr	EA al ah
MOV M, reg	(M) ← (reg)	77 70 71 72 73 74 75 --	JF addr	If S=0, (PC) ← addr	FA al ah
			JM addr	If S=1, (PC) ← addr	FA al ah
<b>ACCUMULATOR GROUP</b>			PCHL	(PC <sub>H</sub> ) ← (H), (PC <sub>L</sub> ) ← (L)	B9
ADD reg	(A) ← (A) + (reg)	* 87 80 81 82 83 84 85 86	<b>CALL GROUP</b>		
ADC reg	(A) ← (A) + (reg) + (CY)	* 8F 88 89 8A 8B 8C 8D 8E	CALL addr	(TOS) ← (PC), (PC) ← addr	CD al ah
SUB reg	(A) ← (A) - (reg)	* 97 90 91 92 93 94 95 96	CMZ addr	If Z=0, (TOS) ← (PC), (PC) ← addr	C4 al ah
SBB reg	(A) ← (A) - (reg) - (CY)	* 9F 98 99 9A 9B 9C 9D 9E	CZ addr	If Z=1, (TOS) ← (PC), (PC) ← addr	CC al ah
ANA reg	(A) ← (A) / (reg)	* A7 A0 A1 A2 A3 A4 A5 A6	CNC addr	If CY=0, (TOS) ← (PC), (PC) ← addr	D4 al ah
XRA reg	(A) ← (A) ^ (reg)	* AF A8 A9 AA AB AC AD AE	CC addr	If CY=1, (TOS) ← (PC), (PC) ← addr	DC al ah
ORA reg	(A) ← (A) V (reg)	* B7 B0 B1 B2 B3 B4 B5 B6	CPO addr	If P=0, (TOS) ← (PC), (PC) ← addr	E4 al ah
ORP reg	(A) ← (A) V (reg)	* BF B8 B9 BA BB BC BD BE	CPE addr	If P=1, (TOS) ← (PC), (PC) ← addr	EC al ah
			CP addr	If S=0, (TOS) ← (PC), (PC) ← addr	F4 al ah
			CM addr	If S=1, (TOS) ← (PC), (PC) ← addr	FC al ah
<b>INCREMENT/DECREMENT REGISTER</b>			N.B. (TOS) ← (PC) designates the following:- (SP-1) ← (PC <sub>H</sub> ), ((SP-2) ← (PC <sub>L</sub> ), (SP) ← (SP)-2		
INR reg	(reg) ← (reg) + 1	** 3C 04 0C 14 1C 24 2C 34	<b>RETURN GROUP</b>		
DCR reg	(reg) ← (reg) - 1	** 3D 05 0D 15 1D 25 2D 35	RST	(PC) ← (TOS)	C9
			RNZ	If Z=0, (PC) ← (TOS)	C0
			RZ	If Z=1, (PC) ← (TOS)	C8
			RNC	If CY=0, (PC) ← (TOS)	D0
			RC	If CY=1, (PC) ← (TOS)	D8
			RPO	If P=0, (PC) ← (TOS)	E0
			RPE	If P=1, (PC) ← (TOS)	E8
			RP	If S=0, (PC) ← (TOS)	F0
			RM	If S=1, (PC) ← (TOS)	F8
			N.B. (PC) ← (TOS) designates the following:- (PC <sub>L</sub> ) ← ((SP)), (PC <sub>H</sub> ) ← ((SP)+1), (SP) ← (SP)+2		
<b>REGISTER PAIR GROUP</b>			<b>RESTART GROUP</b>		
INX rp	(rp) ← (rp) + 1	rp B D H SP PSW	RST 0	(TOS) ← (PC), (PC) ← 016	C7
DCX rp	(rp) ← (rp) - 1	03 13 23 33 --	RST 1	(TOS) ← (PC), (PC) ← 816	CF
LDAX rp	(A) ← ((rp))	0A 1A -- -- --	RST 2	(TOS) ← (PC), (PC) ← 1016	D7
STAX rp	(rp) ← (A)	02 12 -- -- --	RST 3	(TOS) ← (PC), (PC) ← 1816	DF
DAD rp	(H, L) ← (H, L) + (rp) ***	09 19 29 39 --	RST 4	(TOS) ← (PC), (PC) ← 2016	E7
PUSH rp	((SP)-1) ← ((rh)), ((SP)-2) ← ((r1)), (SP) ← (SP) - 2	05 05 05 -- F5	RST 5	(TOS) ← (PC), (PC) ← 2816	EF
POP rp	(r1) ← ((SP)), ((rh)) ← ((SP)+1), (SP) ← (SP) + 2	C1 D1 E1 -- F1 *	RST 6	(TOS) ← (PC), (PC) ← 3016	F7
			RST 7	(TOS) ← (PC), (PC) ← 3816	FF
<b>DIRECT ADDRESS GROUP</b>			<b>ROTATE/CONTROL/SPECIAL GROUP</b>		
LDA addr	(A) ← (addr)	3A al ah	RLC	((A <sub>n+1</sub> ) ← (A <sub>n</sub> ), (A <sub>0</sub> ) ← (A <sub>7</sub> ), (CY) ← (A <sub>7</sub> ) ***	C7
STA addr	(addr) ← (A)	32 al ah	RRC	((A <sub>n</sub> ) ← (A <sub>n+1</sub> ), (A <sub>7</sub> ) ← (A <sub>0</sub> ), (CY) ← (A <sub>0</sub> ) ***	CF
LHLD addr	(L) ← (addr), (H) ← (addr+1)	2A al ah	RAL	((A <sub>n+1</sub> ) ← (A <sub>n</sub> ), (A <sub>0</sub> ) ← (CY), (CY) ← (A <sub>7</sub> ) ***	17
SHLD addr	(addr) ← (L), (addr+1) ← (H)	22 al ah	RAR	((A <sub>n</sub> ) ← (A <sub>n+1</sub> ), (A <sub>7</sub> ) ← (CY), (CY) ← (A <sub>0</sub> ) ***	1F
			NOP	No operation	00
<b>IMMEDIATE GROUP</b>			HLT	Processor stopped until interrupt or reset	76
MVI A, data	(A) ← data	3E dd	DI	Interrupts disabled	F3
MVI B, data	(B) ← data	06 dd	EI	Interrupts enabled after next instruction	FB
MVI C, data	(C) ← data	0E dd	XTHL	((H) ← ((SP)), (H) ← ((SP)+1)	E3
MVI D, data	(D) ← data	16 dd	SPEL	((SP <sub>H</sub> ) ← (H), (SP <sub>L</sub> ) ← (L)	F9
MVI E, data	(E) ← data	1E dd	XCHG	((H) ← (D), (L) ← (E))	EB
MVI H, data	(H) ← data	26 dd	DAA	Decimal adjust accumulator	*
MVI L, data	(L) ← data	2E dd	CMA	(A) ← (A)	2F
MVI M, data	(M) ← data	36 dd	STC	(CY) ← 1	37
ADI data	(A) ← (A) + data *	06 dd	CMC	(CY) ← (CY)	3F
ACI data	(A) ← (A) + data + (CY) *	CE dd	OUT port }		D3 port
SUI data	(A) ← (A) - data *	D6 dd	IN port }		DB port
SBI data	(A) ← (A) - data - (CY) *	DE dd			
ANI data	(A) ← (A) & data *	E6 dd			
XRI data	(A) ← (A) ^ data *	EE dd			
ORI data	(A) ← (A) V data *	FE dd			
CPI data	(A) - data *	FE dd			
LXI B, addr	(B) ← ah, (C) ← al	01 al ah			
LXI D, addr	(D) ← ah, (E) ← al	11 al ah			
LXI H, addr	(H) ← ah, (L) ← al	21 al ah			
LXI SP, addr	(SP <sub>H</sub> ) ← ah, (SP <sub>L</sub> ) ← al	31 al ah			

# PEEK & POKE

## NOG EEN PAAR INTERESSANTE COMBINATIES VAN BASIC EN MACHINETAAL

Deze routine geeft de geheugenplaats in de VIDEO-RAM (waar eventueel een dot terecht komt).

De basic routine :

```
10 MODE 4A
20 INPUT "SCRN";X,Y:POKE #2011,Y:POKE #2013,(X IAND #FF):POKE
   #2014,X SHR 8
30 CALLM #2000
40 PRINT:PRINT HEX$(PEEK(#2051));HEX$(PEEK(#2050)):GOTO 20
```

De machinetaal routine (in te voeren in utility met het S (substitute) command):

```
2000 F3 F5 3A 40 00 F5 E6 3F F6 80 32 06 FD E5 D5 C5
2010 3E 3C 01 3C 00 CD B9 EB 22 50 20 C1 D1 E1 F1 32
2020 06 FD F1 FB C9
```

Twee manieren om het toetsenbord te gebruiken als schrijfmachine :

```
1. 1000 A=GETC:IF A=0 THEN 1000
   1010 PRINT CHR$(A);:GOTO 1000
```

U zal merken dat nu ook de cursortoetsen en de tabtoets een character herbergen.

```
2. 10 CALLM #3000
   15 IF PEEK(#2010) >=0 THEN 10
   20 PRINT CHR$(PEEK(#2010));
   30 GOTO 10
```

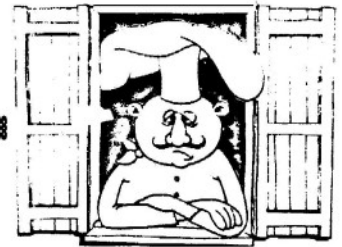
De machinetaalroutine voor dit programma :

```
3000 F5 E5 CD BB D6 32 10 20 E1 F1 C9
```

GETC-routine in BASIC-ROM (#D688)  
plaatst ASCII waarde van toets in register A.

```
1000 PRINT CHR$(12)
1010 FOR A=0 TO 10
1020 POKE #B9E4+2*A,#FF
1025 POKE #B9E4+2*A+#86,#FF
1030 NEXT
1035 CURSOR 23,12:PRINT "WARNING"
1040 FOR B=20 TO 1 STEP -1
1045 COLORT 0 9 9 0
1046 GOSUB 1100
1050 WAIT TIME B
1055 COLORT 0 9 0 9
1056 GOSUB 1100
1060 WAIT TIME B
1065 NEXT
1070 GOTO 1040
1100 RJ=GETC:IF RJ<>32 THEN RETURN
1130 PRINT :INPUT "LIST PROGRAM < Y/N > ":RJ$
1140 IF RJ$="Y" THEN PRINT CHR$(12):PRINT :LIST 1000-1070:GOSUB 2150:GOTO 20
1141 IF RJ$="N" THEN PRINT CHR$(12):PRINT :GOTO 20
1145 CURSOR 0,10:PRINT SPC(30):CURSOR 0,11
1150 RETURN
```

# CATALOG



DE PROGRAMMABIBLIOTHEEK.....

1) INFORMATIE & DEMONSTRATIEPROGRAMMA'S \*

TITEL	GEHEUGEN	MODE	LISTING
- 4 COLOR DEMO (LISSAJOUS)	32K	6	N0
- 16 COLOR/SOUND DEMO	8K	3A	
- OLDTIMER	32K	6	N0
- INFO VIDEORAM MODE0 (DAI)	12K	0	
- THE STING	32K	0	
- MENUET V. BEETHOVEN	32K	0	
- DE VOGELHANDELAAR	32K	0	
- LANDSCHAP	32K	5A	
- PROPELLER (DAI)	32K	6A	N1
- ELECTIONS	32K	5	
- SUM (64 DIGITS-REKENEN) (DAI)	8K	0	
- ROTATING PYRAMIDE (DAI)	32K	6	
- DUTCH DEMO (DAI)	32K	6/5	
- EASY CIRCLES	32K	5A	
- HANEN1	8K	2	
- HANEN2	12K	4	
- HANEN4	32K	6	
- BLOXY1 (COLORG-DEMO)	8K	2	N1
- BLOXY2 (COLORG-DEMO)	12K	4	N1
- BLOXY4 (COLORG-DEMO)	32K	6	N1
- LUNAR LANDING (H. BAKKER)	8K	1A	
- RAKETSPEL (H. BAKKER)	8K	1A	N1
- KANONSPEL (H. BAKKER)	8K	1A	
- SUBMARINE (H. BAKKER)	8K	1A	
- STAR-TREK (H. BAKKER)	8K	1A	
- GETALCONVERSIES (J. VERDONK)	8K	0	N1
- OTHELLO (J. VERDONK)	32K	4A	
- OTHELLO(francais) (JCC)	32K	4A	
- STOCK EXCHANGE (JCC)			
- COLOR-RUIT DEMO (J. VERDONK)	8K	3	N1
- WARI (H. VAN COOTEN)	8K	0	
- DIG/ANALOG CLOCK "	8K	0/GRAFISCH	N2
- VIDEOTEXT "	8K	0	N3
- REACTIESNELHEID "	8K	0	
- VAHTZEE (ESVELD Jr.)			
- BARRICADE (F. DRUYFF)	8K	2	N3

# CATALOG

## PROGRAMMABIBLIOTHEEK part 2

### 2) GAMES - EDUCATIEF - UTILITIES

\*\*

TITEL	minimum	GEHEUGEN	MODE	LISTING
- REAL TIME CLOCK	(ML)	8K	0	N2
- TOWERS OF HANOI		12K	3A	
- ARRANGING NUMBERS + COUNT WITH ME		32K 12K	6 4	
- A MAZE ... AMAZING		48K	5	
- TEKST IN GRAFISCHE MODE		32K	ALLE GRAFISCHE MODES	
- TRAFFICTEST		32K	5	
- MUSIC TUTOR 1	(DAI)	32/48	3/5	
- MUSIC TUTOR + THE STING		48K	6	
ZIE BESCHRIJVING IN NUMMER 3				
- SIMON		12K	3A	
- VIER OP EEN RIJ		12K	4A	N0
- DISASSEMBLER		32K	0	
- HANNIBAL 2000		32K	6	
- QUICK DESIGN 555		48K	6	
- FAST GRAPHICS		8K	1..6	
MAAK ZONDER PROBLEMEN TEKENINGEN EN ANIMATIES IN MACHINETAAL AAN SUPERSNELHEDEN. KENNIS VAN 8080 ML IS NIET VEREIST. WORDT GELEVERD MET HANDLEIDING EN BASIC DEMO.				

### 3) PROFESSIONELE PROGRAMMA'S

\*\*\*

- ASSEMBLER+EDITOR+LOADER+ DISASSEMBLER	32K	0	2250 FR
BIJ DE ASSEMBLER ONTVANGT U EEN UITVOERIGE HANDLEIDING EN EEN HANDIGE 8080 REFERENCE CARD.			
- GRAFIC ARTIST	48K	5/6	1250 FR
ZIE BESCHRIJVING EN HANDLEIDING IN NUMMER 2			
- MUSIC ARTIST	48K	6	1250 FR
- WORD PROCESSOR	32K	0	1250 FR
- SORT (DATABANK)	32K	0	1250 FR
- ADRESBESTAND	32K	0	1250 FR
- FAST GRAF TEXT(MACHINETAAL) +	8K	1..6	1250 FR
FGT TABLE CREATOR:MAAK UW DAIpc COMPATIBEL MET CHARACTER- GRAFIEKEN GEBASEERDE MICRO'S ZOALS PET,OSI,TRS80..... DE SNELHEID BENADERT DE SNELHEID VAN PRINT IN MODE 0 !!! EEN DEMONSTRATIEPROGRAMMA DAT TOONT HOE U FGT KAN GEBRUIKEN IN UW PROGRAMMA'S WORDT BIJGELEVERD. OP AANVRAAG ONTVANGT U TEVENS HET SOURCEPROGRAMMA VAN FGT.			

# DAINAMIC

## PERSONAL COMPUTER USERS CLUB

### LIDMAATSCHAP

---

De jaarlijkse contributie bedraagt 750Bfr/50 Gld.

Het lidmaatschap biedt U het volgende:

- \*tweemaandelijke NEWSLETTER
- \*HOTLINE-TELEFOON SERVICE : 016/698623
- \*toegang tot de programmabibliotheek
- \*hardcopys van uw programma's
- \*hardware kortingen

De betaling kan gebeuren op volgende manieren:

1. Bankoverschrijving naar KREDIETBANK WESTMEERBEEK  
nr 406-3016141-33, mededeling: lidmaatschap DAInamic
2. Eurocheque ten bedrage van 750 Bfr opsturen naar het redactieadres.
3. Giro: Indien U in Nederland een girorekening heeft, kunt U het bedrag overschrijven op de Belgische girorekening van DAInamic.  
vermeld: afdeling buitenland/Kredietbank Westmeerbeek/nr 404-3016141-33  
mededeling: lidmaatschap DAInamic.
4. Contant: het bedrag in biljetten opsturen.

---

### SOFTWARE OP CASSETTES

---

programma's uit categorie \*: 120 Bfr/8 Gld, 45 Bfr/3 Gld per  
extra programma op dezelfde tape.  
uit categorie \*\*: 375 Bfr/25 Gld, 250 Bfr/17 Gld per  
extra programma op dezelfde tape  
uit categorie \*\*\*: zie programmabibliotheek

---

SOFTWARE CASSETTES: 35 Gld of 500Bfr voor 10 blanco cassettes.  
(assortiment C10/C20 naar keuze)  
buitenland 50Bfr verzendkosten extra.

---

OM DE CORRESPONDENTIE VLOT TE LATEN VERLOPEN EN OM DE ZAAK OVERZICHTELIJK TE HOUDEN HADDEN WE GRAAG EEN PAAR GEGEVENS VAN U. SCHRIJF DE GEGEVENS A.U.B. IN DRUKLETTERS ZO VERMIJDEN WE MISVERSTANDEN.

\*\*\*\*\*

NAAM : .....

VOORNAAM : .....

STRAAT, NUMMER : .....

POSTNUMMER, WOONPLAATS : .....

LAND : .....

TELEFOONNUMMER : .....

TYPE COMPUTER(vb 12K color sound) : .....

DAI GEBRUIKER SINDS : .....

BEROEPSBEZIGHEID/OPLEIDING : .....

COMPUTERACTIVITEITEN/INTERESSE : .....

.....

.....

TYPE CASSETTERECORDER/FLOPPY : .....

INTERESSE VOOR 8080 MACHINETAAL (J/N) .....

WENST U EVENTUEEL RECLAMEFOLDERS TE ONTVANGEN (J/N) .....

SUGGESTIES I.U.M. DAInamic .....

.....

.....

\*\*\*\*\*

DAI PERSONAL COMPUTER °°° DAI PERSONAL COMPUTER °°°DAI PERSONAL

```

0 zwart           alle adressen in HEXvorm!
1 blauw
2 d.rood          298-29C   start heap           131,0   output scrn+
3 rood            29D-29E   size heap           131,1   RS232
4 paars           29F-2A0   start text buffer   131,2   screen only
5 groen           2A1-2A2   start symbol table  135,2   edit buffer
6 d.bruin         2A3-2A4   end of symbol table 135,2   read from
7 l.bruin         2A5-2A6   bottom screen ram   edit buffer
8 grijs
9 blauw
10 oranje         75        cursor symbol       MODE    XMAX    YMAX
11 rose           74        cursor mode         1/2    71     64
12 l.blauw        72-73    cursor position     3/4    159    129
13 l.groen
14 geel           40,28    cass motor 1 ON
15 wit            40,18    cass motor 2 ON
                  40,30    1 and 2 OFF
                  MERGE
                  °CLEAR XXX
                  °LOAD "A"
                  °EDIT BREAK/BREAK
                  °LOAD "B"
                  °POKE 135,2

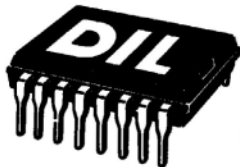
COLORG R1 R2 R3 R4
          20 21 22 23
16 :R2*R1 R4*R3
17 :R1*R2 R3*R4      32K 7XXX
18 :R3*R1 R4*R2      12K 2XXX
19 :R1*R3 R2*R4       8K 1XXX

LIJN     CTRL  COLOR      LIJN  CTRL  COLOR
23       BFEF BFEF      11    B9A7 B9A6
22       BF69 BF68      10    B921 B920
21       BEE3 BEE2       9     B89B B89A
20       BE5D BE5C       8     B815 B814
19       BDD7 BDD6       7     B78F B78E
18       BD51 BD50       6     B709 B708
17       BCCB BCCA       5     B683 B682
16       BC45 BC44       4     B5FD B5FC
15       BBBF BB8E       3     B577 B576
14       BB39 BB38       2     B4F1 B4F0
13       BAB3 BAB2       1     B46B B46A
12       BA2D BA2C       0     B3E5 B3E4

CTRL&COLOR BYTES IN A-MODE
MODE     CTRL     COLOR  LIJN
1A/2A    BAE7     BAE6   3
          BA61     BA60   2
          B9DB     B9DA   1
          B955     B954   0
3A/4A    ACD3     ACD2   3
          AC4D     AC4C   2
          ABC7     ABC6   1
          AB41     AB40   0
5A/6A    7557     7556   3
          74D1     74D0   2
          744B     744A   1
          73C5     73C4   0

FD00 b2 page signal      FF00 ser.inp.buff
      b3 serial out rdy  FF01 b0-6 keyb.inp.
      b4 right paddle    b7 in7 DCE
      b5 left paddle     FF02 Interr.reg.
      b6 random data     FF03 b1 frame error
      b7 cass. input     b2 overrun error
FD01 Trigger paddle     b3 rec.buf.loaded      FF09 TIMER 0
FD04 0-3 volume ch.1(0) b4 trans.buf.empty     FFOA TIMER 1
      4-7 volume ch.2(1) FF04 COMMAND REGISTER  FFOB TIMER 2
FD05 0-3 volume ch.3(2) FF05 BAUD RATE REGISTER FFOC TIMER 3
      4-7 volume noise  FF06 ser.out buf.      FF0D TIMER 4
FD06 b0 cass.out        FF07 keyb.output       8253
      b1/2 paddle select FF08 interr.mask reg.  CH 0 FC00/FC01
      b3 paddle enable   CH 1 FC02/FC03
      b4 cass motor 1    CH 2 FC04/FC05
      b5 cass motor 2    STATUS FC06/FC07
      b6/7 ROM BANK SWITCH
                  TEST EVENT
                  PEEK(éFD00) IAND 32
                  PEEK(éFD00) IAND 16
                  PEEK(éFD00) IAND 48

```

**D.I.L.-ELEKTRONIKA**

Mijnsherenlaan 108, 3081 CH Rotterdam

**ALLE DOE-HET-ZELF ELEKTRONIKA - TECHN. TIJDSCHRIFTEN EN -BOEKEN**

LEGOTRONICS

Middenstraat 8

8800 ROESELARE BELGIE

tel. 051/207878

ORDIMAX

Rue de la Bonnefemme 11

4030 GRIVEGNEE BELGIE

MULTISOFT

Rue Bague 25

75015 PARIS FRANCE

7838837

TELEC

Steenstraat 40

9711 GP GRONINGEN NEDERLAND

MSB R.NEDELA

MARTKSTRASSE 3

POSTFACH 1420

D7778 MARKDORF GERMANY

COMPAC

Plaats 25

2513 AD DEN HAAG NEDERLAND

HCC NEDERLAND hobby computer club

Prinsenhof 11

2641 RN PIJNACKER

NEDERLAND

DAI BRUSSEL

Raketstraat 60

1130 BRUSSEL BELGIE

02/2166010

HCC BELGIE

Borkelstraat 51

2120 SCHOTEN BELGIE

031/589674

DAI NEDERLAND

Van Vollenhovenstraat 15A

3016 BE ROTTERDAM NEDERLAND

010/361288

Stichting BASICned

Tolakkerweg 81

3739 JJ HOLL.RADING NEDERLAND

DIDACOM computers&amp;onderwijs

p/a I.BROEKMAN AVENBEECK 98

2182 RZ HILLEGOM

2520/18032 NEDERLAND