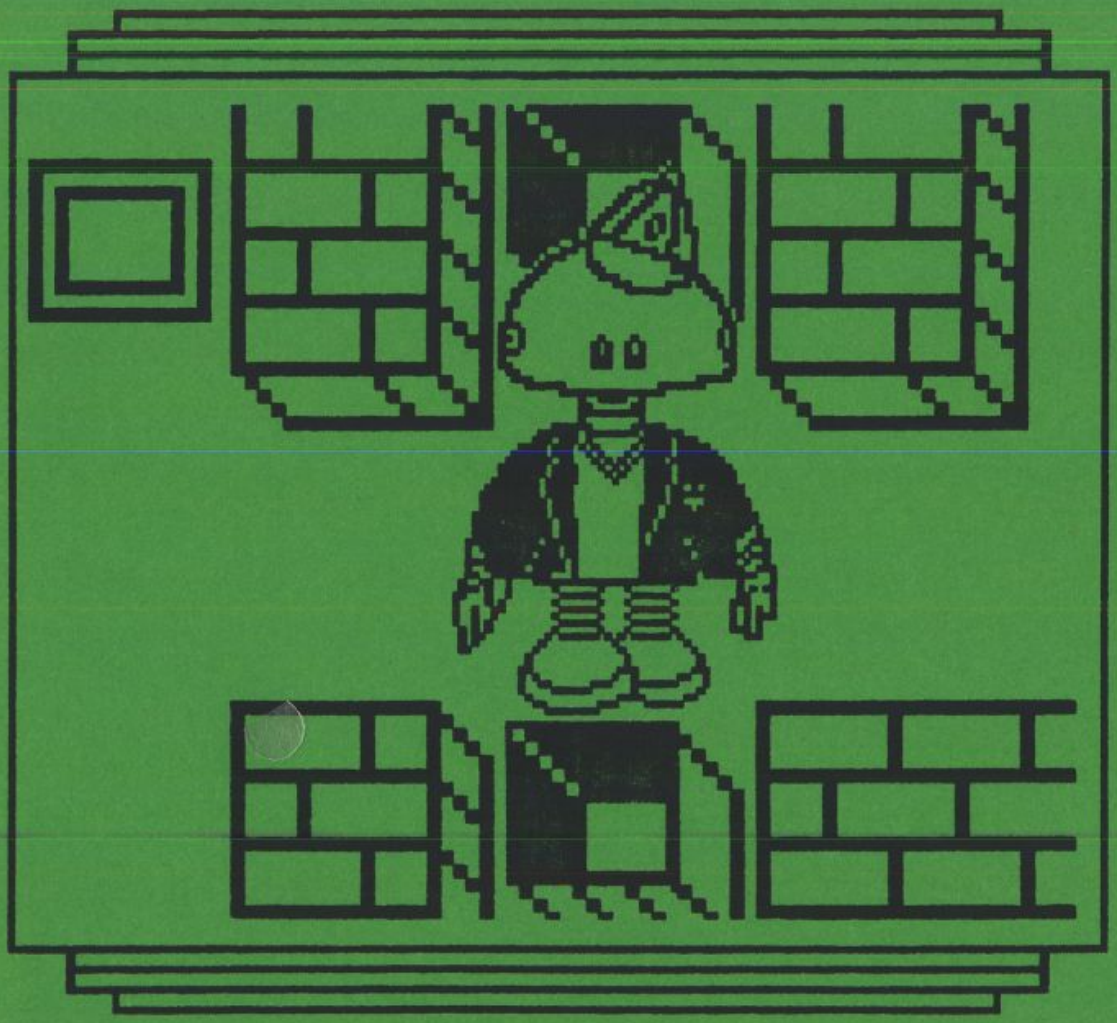
# BULLETIN

## SINCLAIR GEBRUIKERSGROEP GRONINGEN/ASSEN



11e jaargang nr 6
februari '90

# COLOFON



VOORZITTER/
PENNINGMEESTER/
VERHUUR:
Jan Arends
Heiligelaan 66
9636 CP Zuidbroek
tel: 05985-2247

Giro 5965342 t.n.v.
rekening SGG.
ZUIDBROEK.

SECRETARIS:
Martin den Hollander
Numero Dertien 8
9644 TV Veendam
tel: 05978-45474

VICE VOORZITTER/
VICE SECRETARIS:
Roelof Koning
Selwerderstraat 26
9717 GK Groningen
tel: 050-124298

REDACTIE CONTACT/
VICE PENNINGMEESTER/
MATERIAAL COMMISSARIS:
Coen Ballintijn
B. Boermalaan 7
9765 AP Paterswolde
tel: 05907-91482

Redactie: Mevr. F. Elstrodt, Rudy Biesma, Tonnie Stap en Johan Koning.
Correspondentie adres: Coen Ballintijn. adres: zie boven.
Kopij en vragen graag aan de redactie contactpersoon!

Het SGG-BULLETIN is een uitgave van de Sinclair gebruikersgroep Groningen/Assen. Het geeft naast verenigingsnieuws, veel informatie over en voor de SINCLAIR en SAM computers. (Hardware, software, programma- aanpassingen, uitbreidingen enz.) Het Bulletin verschijnt 10 keer per jaar in de maanden september tot en met juni. Artikelen, listings, illustraties en andere inzendingen zijn voor de verantwoordelijkheid van de inzender. Gepubliceerde programma's zijn getest maar niet gegarandeerd zonder fouten.

De sluitingsdatum voor de kopy, wordt in elk Bulletin op pagina 5 vermeld.
Overname van artikelen, illustraties en andere publicaties zijn uitsluitend toegestaan met toestemming van de redactie.

Het lidmaatschap van onze gebruikersgroep bedraagt per kalenderjaar:
f 20,00 voor personen tot en met 17 jaar en
f 30,00 voor personen van 18 jaar en ouder.
Bij deze prijs is het Bulletin inbegrepen.
Losse nummers f 4.00 . (nabestellen van oudere nummers á f 4.00).

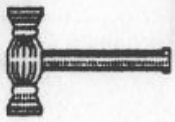
De SGG diskettes zijn in de volgende formaten verkrijgbaar:
3.5 inch 40 tracks SS (OPUS) en 3.5inch 80 tracks DS (DISCiPLE).
Ze kosten: f 5.00. uitgezonderd "TW3 DTP', die is f 10.00.
Verzendkosten f 2,50 per stuk. f 3,00 bij meer.

Advertentiekosten voor niet-leden f 5.00.

## U kunt lid worden van de SGG

Door u op te geven bij de penningmeester.

# VAN DE VOORZITTER



Deze week eindelijk mijn computerhoek eens opgeruimd. Het moest er toch een keer van komen, want de puinhoop werd steeds groter. En wat je dan allemaal tegenkomt. Alle Bulletins vanaf het allereerste begin. Even hier en daar een stukje lezen: "Het loopt goed met het ledenaantal. We zijn de grens van 100 overschreden." Dan een paar maanden later: " Het bestuur heeft besloten een ledenstop in te voeren want het groeit ons eerlijk gezegd toch wel een beetje boven het hoofd." En dan gaat het over 1983/84 en het betreft allemaal noordelingen. Schitterend om te lezen. We zijn nu 10 (wat computers betreft) stormachtige jaren verder en hopen dit jaar op zo'n 60 leden verspreid over het hele land. Het kan verkeren.

Toch blijven we proberen de vereniging draaiende te houden door o.m het meedoen aan beurzen. Op zaterdag 26 maart zullen we aanwezig zijn op de Computer Beurs Assen in de Sport en Evenementenaccommodatie "de Smelt" ,de voormalige Thrianta-hal, van 10.00 tot 16.00 uur. Dit is altijd een leuke beurs waar best het een en ander te snuffelen valt.

Eveneens op 26 maart zal voor de vijfde keer de Friese HCC-dag door de HCC afd. Friesland worden georganiseerd. Namens de HCC Sinclair Gebruikersgroep zijn wij gevraagd voor hen de honneurs waar te nemen, hetgeen we met plezier zullen doen. Deze beurs wordt gehouden in zaal Prins, Heliconweg 50 d.i. de voorzijde van de Frieslandhal. Tijd 10.00 tot 17.00 uur.
We hopen velen van u hier te ontmoeten.

Op zaterdag 26 februari vindt het Noordelijk Amateur Treffen plaats in de Martinihal te Groningen. Of we hier aanwezig zullen zijn, is op dit moment nog niet bekend.

met vr. gr.
Jan Arends

# VAN DE REDACTIE



HALLO ALLEMAAL,

Voor u ligt weer een Bulletin, ons eigen verenigingsblad. En juist dat laatste woord is zo belangrijk. Het betekent dat we in eerste instantie u voorzien van nieuws over en voor onze vereniging. Meestal wordt dat verzorgd en bezorgd door bestuur en / of redactie.

In tweede instantie betekent het dat we proberen u van zoveel mogelijk nieuws te voorzien dat betrekking heeft met onze gemeenschappelijke interresse. Namelijk het Sinclair- en / of Sam computergebeuren. Dat nieuws wordt meestal bijeengegaard door de redactie. Verder vindt u in het Bulletin artikelen over hard- en software, of byzondere toepassingen hiervoor en / of hiervan. Deze artikelen worden als het goed is door de leden aangeleverd. Alle mensen die er het afgelopen jaar hebben toe bijgedragen dat wij van de redactie nog duimen over hebben. willen we nog van ganser harte bedanken! Misschien hebt u ook dit jaar weer copy voor ons. Bij voorbaad dank.

Het artikel op pag. 5 bereikte mij pas op het allerlaatste moment, zodat ik helaas niet meer in de gelegenheid was om het op opeen volgende pagina's te plaatsen.

tot de volgende keer.

In dit nummer:


| | auteur | blz. |
|---|---|---|
| - Coverscreen | : zie art. op pag 6 | |
| - Colofon | : Bestuur | 2 |
| - Van de voorzitter | : Jan Arends | 3 |
| - BETA BASIC TIPS (1) | : Rob Willig | 5 |
| - SAM: Brainsport | : F. Elstrodt | 6 |
| - RAMdata files in Exp. Art Studio (2) | : C.M. Ballintijn | 8 |
| - Uni-DOS: Create File (2) | : H. v. Leeuwen | 10 |
| - Input/Output: de Functie VAL$ | : Impuls / redactie | 15 |
| - MCODE: Stap voor Stap (11) | : J. v. Alteren | 18 |
| - Esgeegeetjes | : leden | 20 |
| - vervolg van: Beta Basic Tips (2) | : Rob Willig | 21 |
| - Bijeenkomsten | : Bestuur | 22 |
| - HCC-bijeenkomsten | : HCC | 22 |
| - Emulator-besteladres | : G. Lunter | 22 |
| - Reparatie-adres | : A. Hoekman | 22 |
| - Beurs data | : redactie | 23 |
| - Handel (2e hands) | : F. Vink | 23 |


Sluitingsdatum copy:

Maart nummer : 22 Februari

April nummer : 19 Januari

# BETA BASIC TIPS (1)

door: Rob Willig.

Beta BASIC is een zeer mooie uitbreiding van de toch al fraaie Spectrum BASIC. Met behulp van extra toetswoorden en functies kunt u de fraaiste databases of spellen opzetten met razendsnelle zoekacties en fraaie schermeffecten. En dat alles met programma's die u dank zij PROCedures heel mooi modulair kunt opbouwen, d.w.z. een stuurprogramma met daaronder alle subroutines (of -modules). Over modulair programmeren hebben in het verleden al artikelen in RAM gestaan (Bioritme berekening, spel en doorlopende klok).

*IMITEREN*

Het blijkt heel best mogelijk te zijn, sommige functies die in Beta BASIC voorkomen, in gewoon Spectrum-Basic te "imiteren". Beta BASIC kent bijvoorbeeld functies om grote hoeveelheden getallen zuinig op te slaan. Ziehier:

```
DPEEK  : FN P (X)  = PEEK X + 256 * PEEK ( X + 1 )
MEM    : FN M ()   = 65535 - USR 7962
CHAR$  : FN C$ (X) = ( CHR$ X - 256 * INT ( X / 256 ) ) +
                     CHR$ INT ( X / 256 )
NUMBER : FN N (X$) = 256 * CODE X$ ( 1 ) + CODE X$ ( 2 )
```

Denkt u er wel aan, dat u met CHAR$() en zijn tegenhanger NUMBER() alleen hele getallen (integers) kunt verwerken. Als u dus met gebroken getallen (cijfers achter de komma (punt)) werkt, moet u ze eerst met 10, 100 of 1000 vermenigvuldigen. De factor hangt af van de precisie die u wilt en de maximale grootte van de getallen (0 >= X <= 65.535).

*LETTERS*

Een andere aardigheid van Beta BASIC is de mogelijkheid letters te PLOTten op elke plaats en om letters van elke grootte te produceren. Ook daarvoor hebt u Beta BASIC niet nodig. (Bijna) iedere Spectrum gebruiker heeft namelijk ergens een "horizons" bandje slingeren. Dat was die instructiecasssette waarmee u het toetsenbord verkende. Ook daarin zaten van die mooie grote letters. Die werden gemaakt m.b.v. een blokje machinecode, dat bij ieder programma apart werd geladen (dan wisten de programmeurs tenminste zeker dat het er was en dat u niet zomaar hangups zou krijgen). Het heeft weinig zin om die code hier te gaan publiceren. Ten eerste is die code beschermd door PSION copyright en ten tweede is het intikken van 300 getallen in hex of decimale vorm geen pretje (zeker niet als u geen numeriek blok hebt). U kunt die code er net zo goed "uitsnijden". U doet het volgende:

1. Houd een leeg bandje of een diskette bij de hand;
2. Spoel de "HORIZONS" band terug naar het begin van kant a en lees het programma "hardware" in:

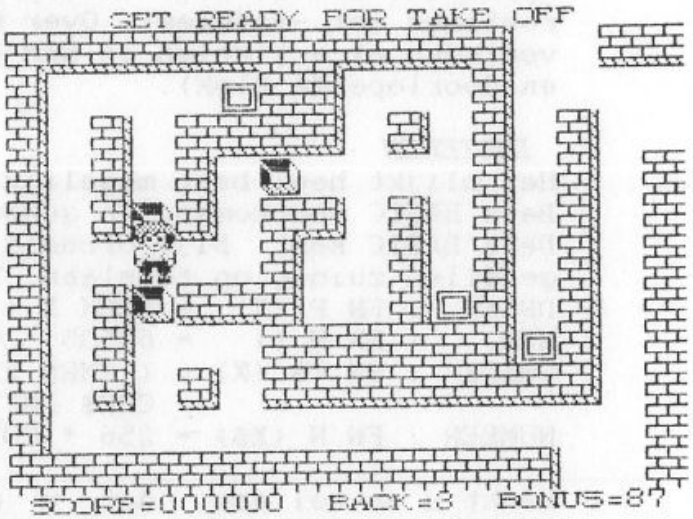
Lees verder pag. 21 →

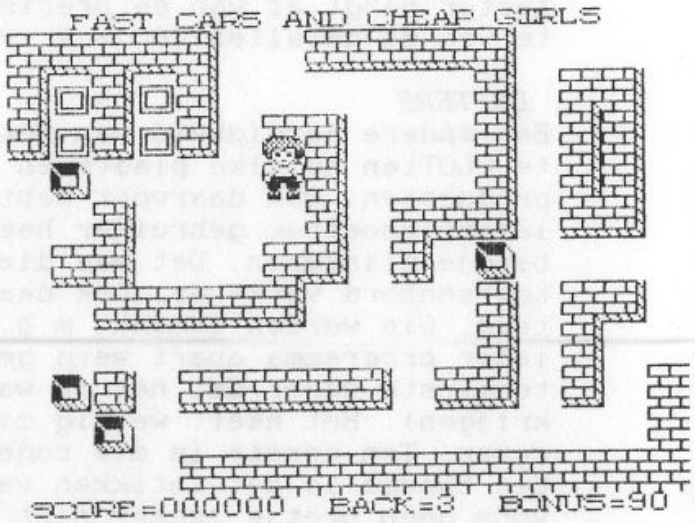
# BRAINSPORT



door: Flora.

Zoals je al uit de kop kunt vernemen, heb ik het over een denkspel. En wat voor één, welke leeftijd je nu ook hebt, als je even niet oplet krijg je gegarandeerd grijze haren. Ik zag het spel op één van onze gebruikersdagen, en herinnerde het ééns gezien te hebben op een PC. Het spel was daar wel wat uitgebreider, zo moest je geloof ik eerst met een lift gaan, om bij de diverse levels te komen. Dus allemaal show en geheugen verslindend. Goed wat houdt Brainsport in: Je ziet een soort doolhof met daarin op diverse plaatsen stenen liggen, en op verschillende plaatsen gemerkte hokjes. Onderaan het beeld zie je de Score, Back 3 en de Bonus 99. De Bonus telt af, terwijl jij denkt, maar je kunt er zo lang over denken als nodig is, dan maar geen bonuspunten. Een vrolijk manneke wacht totdat jij met hem de stenen naar de hokjes brengt.

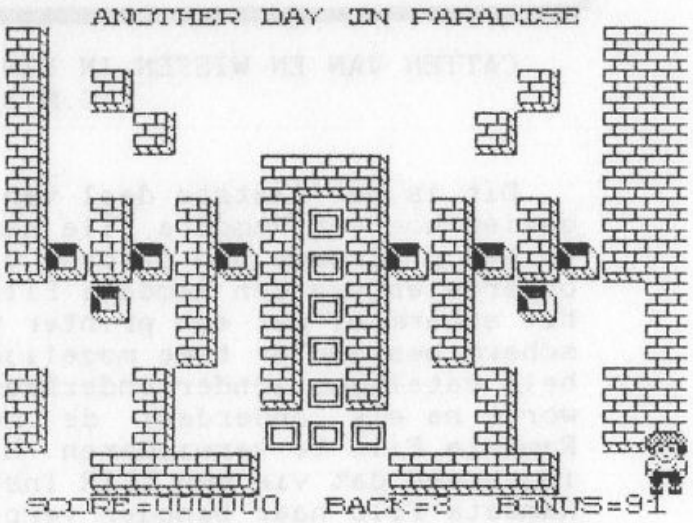


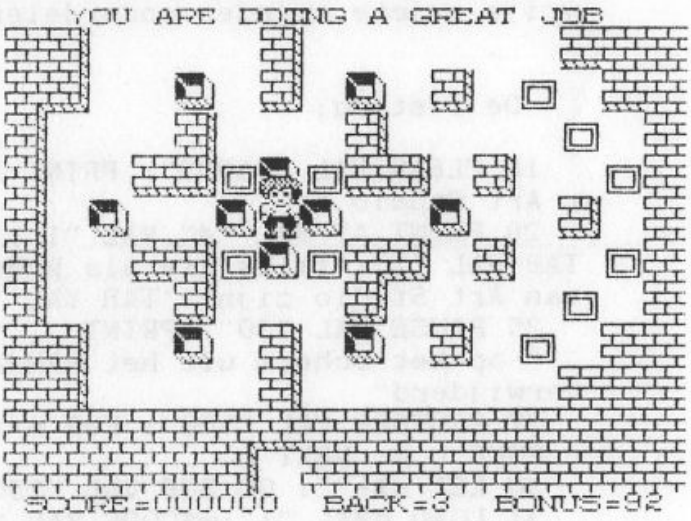



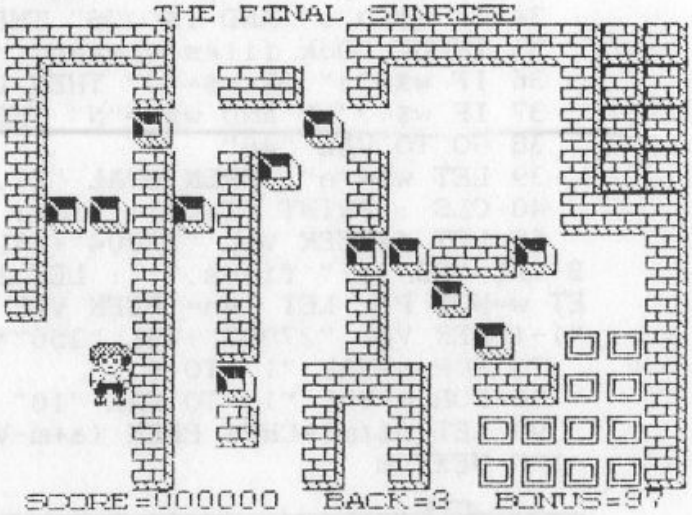
Dit is makkelijker gezegd dan gedaan, je moet namelijk wel even vooruit denken, wil je niet vast komen te zitten. Een tip hier gegeven, meestal moet je omlopen. Als je het level gehaald hebt, klapt het manneke in z´n handjes. Je gaat dan naar het volgende level, en daar krijg je het level- nummer te zien met een nieuw paswoord.

Het is aan te bevelen deze even te noteren, want het nummer correspondeert wel met het juiste paswoord. Er zijn zo´n tachtig levels te gaan, de één wat moeilijker dan de ander. Het is een spel voor de Spectrum, maar je kunt het ook op de Sam Coupé spelen, b.v. met Spekmaker. Als je een Disciple Snap hebt, kun je het beste Optie 5 van Spekmaker kiezen. Eénmaal in Spekmaker maak je door het drukken op de NMI knop en toets 4 je eigen Snap. Dat is handig omdat je niet telkens weer bij level 1 wilt beginnen. Dat zou nachtwerk worden. In het begin Menu kun je kiezen uit de volgende Opties:

Begin new Game - Select Field - Editor - Game Options - Quit.
Bij Game Options kun je kiezen uit:
Change border - Information - Game Controls - Hall of Fame.
Bij Game Controls kun je kiezen uit:
Keybord - Kempston - Cursor.



# ARTSTUDIO(128K): RAMDATA FILES

CATTEN VAN EN WISSEN IN EEN ART STUDIO (128k) RAMDATA FILE.
C.M.Ballintijn.

Dit is het laatste deel van de serie. Vorige keer hebben we gezien hoe een Ramdata File kan worden samengesteld. Deze keer is een programma aan de orde dat een overzicht geeft van de onderdelen van een Ramdata File. De catalogus kan naar keuze op het scherm of met een printer worden geprint. Bij printen op het scherm bestaan er twee mogelijkheden. Bij de eerste wordt de hele catalogus zonder onderbreking geproduceerd. Bij de tweede wordt na elk onderdeel de mogelijkheid gegeven het uit de Ramdata File te verwijderen. Hiervoor is een klein mc routinetje ingebouwd dat via een LDIR instructie het hogere deel van de Ramdata File naar beneden verplaatst en zo de vrijkomende ruimte opvult. De gegevens daarvoor worden uit de Ramdata headers gehaald. De algemene header wordt voor de nieuwe situatie (meer vrije ruimte, minder onderdelen) bijgewerkt.

De listing:

```
  10 CLEAR VAL "26999": PRINT BRIGHT VAL "1":" RAMdata lezer voo
r Art Studio."
  20 PRINT AT VAL "4",VAL "1";"Geeft nummer en titel van alle"''
TAB VAL "2";"files die als RAMdata op de"''TAB VAL "2";"RAMdisk
van Art Studio zijn"''TAB VAL "11";"geplaatst."
  25 PAUSE VAL "30": PRINT '''"Files kunnen tijdens het printen"
''" op het scherm uit het RAMdata"''TAB VAL "3":"bestand worden
verwijderd"
  27 RESTORE VAL "500": FOR n=VAL "65515" TO VAL "65526": READ w
: POKE n,w: NEXT n
  30 LET s$="": GO SUB VAL "250": INPUT "Naam RAMdata file ";m$
  31 LOAD dVAL "1";m$CODE VAL "27000"
  32 CLS : INPUT "Naar scherm     s"'"Naar printer    p ";i$
  33 IF i$="p" OR i$="P" THEN GO TO VAL "39"
  34 IF i$<>"s" AND i$<>"S" THEN GO TO VAL "32"
  35 INPUT "Ook files wissen? j/n  ";w$
  36 IF w$="j" OR w$="J" THEN LET m$="Nog geen.": GO TO VAL "40"
  37 IF w$<>"n" AND w$<>"N" THEN GO TO VAL "35"
  38 GO TO VAL "40"
  39 LET w$="n": OPEN #VAL "2","p"
  40 CLS : PRINT "Titel: ";m$;
  60 LET f=PEEK VAL "27004"+VAL "256"*PEEK VAL "27005": PRINT TA
B VAL "22";f;" files."'': LET a=VAL "27006": DIM n$(VAL "10"): L
ET w=NOT PI: LET len=(PEEK VAL "27000"+VAL "256"*PEEK VAL "27001
")-(PEEK VAL "27002"+VAL "256"*PEEK VAL "27003")
  70 FOR n=VAL "1" TO f
  80 FOR m=VAL "1" TO VAL "10"
  90 LET n$(m)=CHR$ PEEK (a+m-VAL "1")
 100 NEXT m
```

```
110 PRINT TAB VAL "6";: IF n-w<VAL "10" THEN PRINT " ";
120 PRINT n-w;"  ";n$;
130 IF w$="n" OR w$="N" THEN GO SUB VAL "330"
135 IF w$="j" OR w$="J" THEN GO SUB VAL "300"
140 NEXT n
150 PRINT ''TAB VAL "7":"EINDE RAMDATA"
155 CLOSE #VAL "2"
157 IF s$="t" OR s$="T" THEN GO TO VAL "172"
160 IF w$="n" OR w$="N" THEN GO TO VAL "190"
172 RANDOMIZE ((PEEK VAL "27000"+VAL "256"*PEEK VAL "27001")-le
n): POKE VAL "27002",PEEK VAL "23670": POKE VAL "27003",PEEK VAL
 "23671"
 175 RANDOMIZE (f-w): POKE VAL "27004",PEEK VAL "23670": POKE VA
L "27005",PEEK VAL "23671"
 176 INPUT "File nog eens tonen   t"'"File direct SAVEN     s  "
:s$: IF s$="t" OR s$="T" THEN GO TO VAL "32"
 177 IF s$<>"s" AND s$<>"S" THEN GO TO VAL "190"
 179 LET s$="": GO SUB VAL "250": INPUT "SAVE naam nieuwe RAMdat
a File"'m$
 180 SAVE dVAL "1":m$CODE VAL "27000",len-VAL "1"
 190 INPUT "Zelfde File nog eens   z"'"Nieuwe File bewerken   n
  ":i$
 192 IF i$="z" OR i$="Z" THEN GO TO VAL "32"
 195 IF i$<>"n" AND i$<>"N" THEN GO TO VAL "190"
 197 GO TO VAL "30"
 200 STOP
 250 INPUT "Cat?          j/n",i$: IF i$="j" OR i$="J" THEN CAT VAL
"1": GO TO VAL "250"
 260 IF i$<>"n" AND i$<>"N" THEN GO TO VAL "250"
 270 RETURN
 300 INPUT "Wissen?   j/n  ":i$: IF i$="n" OR i$="N" THEN GO TO
VAL "330"
 301 IF i$<>"j" AND i$<>"J" THEN GO TO VAL "300"
 302 LET w=w+VAL "1": LET l=PEEK (a+VAL "12")+VAL "256"*PEEK (a+
VAL "13")
 304 FOR m=VAL "1" TO VAL "14": PRINT CHR$ VAL "8";: NEXT m: PRI
NT "                ";: FOR m=VAL "1" TO VAL "20": PRINT CHR$ VAL
"8";: NEXT m
 306 RANDOMIZE (len-(a-VAL "27000")): POKE VAL "65522",PEEK VAL
"23670": POKE VAL "65523",PEEK VAL "23671": REM aant
 307 RANDOMIZE a: POKE VAL "65519",PEEK VAL "23670": POKE VAL "6
5520",PEEK VAL "23671": REM naar
 308 RANDOMIZE (a+l+VAL "14"): POKE VAL "65516",PEEK VAL "23670"
: POKE VAL "65517",PEEK VAL "23671": REM van
 310 LET len=len-l-VAL "14"
 320 RANDOMIZE USR VAL "65515": RETURN
 330 LET l=PEEK (a+VAL "12")+VAL "256"*PEEK (a+VAL "13"): LET a=
a+VAL "14"+l: PRINT
 340 RETURN
 490 STOP
 500 DATA VAL "33",NOT PI,NOT PI,VAL "17",NOT PI,NOT PI,VAL "1",
NOT PI,NOT PI,VAL "237",VAL "176",VAL "201"
9990 STOP
9999 CLEAR : SAVE dVAL "1":"KvkWisRamD"
```

# UNI-DOS: CREATE FILES (2)

Door Henk van Leeuwen

Allereerst het voor Beta Basic gebruikers over bekende basic commando DPEEK met een voorbeeld met het gebruik van de zogenaamde 'core' command codes ,tik in in tornado en schrijf dit naar diskette via SAVE d*"NAME..."USR 60000,lengte.

```
(c) * TORNADO '93 *
            ORG  60000
            DUMP 60000

;PRINT DPEEK   <adres>

start       DEFB 1                  ;een syntax
            DEFB #F5                ;PRINT statement
            DEFW print_len

1_print     AND  223                ;hoofdletters
            CP   "D"                ;teken d of D
            RET  NZ
            RST  #28
            CP   #BE                ;basic statement PEEK
            RET  NZ
            RST  #28
            RST  #10                ;invoeren <adres>
            DEFW #1C82
            RST  #18
            DEFB 13                 ;SYN.END

            RST  #18                ;printen naar screen
            DEFB 100                ;PR.SCR
            RST  #10
            DEFW #1E99              ;haal BC op uit stack
            LD   A,(BC)             ;verplaats dit nummer van  BC
            LD   L,A                ;naar het HL register
            INC  BC
            LD   A,(BC)
            LD   H,A
            RST  #18                ;print als een decimaal nummer
            DEFB 92                 ;PR.NUM2
            RST  #18                ;ga naar de volgende regel
            DEFB 89                 ;PR.NL
            RST  #18
            DEFB 14                 ;COM.END

print_len   EQU  $-1_print
```

Of wanneer geen assembler aanwezig is hierna een paar basic regeltjes met data .

```
10 DATA 1,245,31,0,230,223,254,68
11 DATA 192,239,254,190,192,239,215,130
12 DATA 28,223,13,223,100,215,153,30
13 DATA 10,111,3,10,103,223,92,223
14 DATA 89,223,14
```

Sinds mijn lidmaatschap van OUTLET komt het nog al eens voor dat ik regels uit deze grote basic blokken wil bewaren voor later gebruik hierbij doel ik op de basic regels met mc ,door het vele werk om een zo'n regel te saven heb ik de volgende CREATE file ontworpen ,met deze CREATE is het mogelijk om een regel uit een basic programma te saven naar diskette.

```
;EXP <OVER> D<drive> "NAME......" LINE <regel nummer>

          ORG  60000
          DUMP 60000

start     DEFB 1                    ;een syntax
          DEFB 185                  ;EXP basic woord
          DEFW EXP_LEN

1_exp     RST  #18
          DEFB 115                  ;OVER.SYN
          AND  223                  ;hoofdletters
          CP   "D"                  ;gebruik 'D' voor drive
          RET  NZ
          RST  #28
          RST  #18                  ;controleer voor d1,d2 of d*
          DEFB 120                  ;DR.NUM
          RST  #18                  ;volgend karakter   '"' of ','
          DEFB 112                  ;SEPARATOR
          RET  NZ
          RST  #18                  ;vraag de programma naam
          DEFB 123                  ;GET.NAME
          CP   #CA                  ;gebruik het 'LINE' commando
          RET  NZ
          RST  #28
          RST  #10                  ;ga naar de spectrum ROM
          DEFW #1C82                ;vraag om een regel nummer
          RST  #18
          DEFB 13                   ;SYN.END

          RST  #18
          DEFB 111                  ;drive nummer
          RST  #10
          DEFW #1E99                ;zoek regel nummer in stack
          LD   H,B                  ;verwissel BC to HL
          LD   L,C
```

```
            RST  #10                ;ga naar spectrum ROM en zoek
            DEFW 6510               ;het adres van de regel
            JR   Z,CONT_2
            RST  32
            DEFB 141                ;CODE parameter error
CONT_2      LD   (15891),HL         ;startadres regel nummer HDOD
            INC  HL                 ;volgend byte
            INC  HL                 ;eerste byte van de lengte
            LD   C,(HL)             ;verplaats naar C register
            INC  HL                 ;verhoog byte
            LD   B,(HL)             ;verplaats naar B register
            LD   A,B                ;beide nul dan geen regel
            OR   C                  ;in basic
            JR   NZ,cont_1
            RST  32
            DEFB 141                ;CODE parameter error
cont_1      INC  BC                 ;gebruik ook het lage en hoge
            INC  BC                 ;byte van het regel nummer en
            INC  BC                 ;hetzelfde van de lengte regel
            INC  BC
            LD   (15889),BC         ;verplaats naar  DRAM   HDOB
            LD   (15893),BC         ;verplaats naar  DRAM   HDOF
            LD   A,1                ;directory description
            LD   (15877),A
            LD   A,"D"              ;device type
            LD   (15876),A
            LD   A,0                ;file type
            LD   (15888),A
            RST  #18                ;Opend een   write file
            DEFB 61                 ;OPEN.FILE
            RET  NZ
            RST  #18                ;save 9 bytes als een header
            DEFB 44                 ;SAVE.HEAD
            LD   HL,(15891)         ;code blok  start adres
            LD   DE,(15889)         ;code blok  lengte
            LD   (15274),DE
            RST  #18                ;saved een blok van DE bytes
            DEFB 39                 ;SAVE.BLOCK
            RST  #18                ;closed een open write file
            DEFB 66                 ;CLOSE.FILE
            RST  #18
            DEFB 14                 ;COM.END

EXP_LEN     EQU  $-L_EXP
```

Het volgende basic blok met DATA is weer de CREATE file als hiervoor ,voor degene die geen assembler bezitten .

```
  8 DATA 1,185,99,0,223,115,230,223
  9 DATA 254,68,192,239,223,120,223,112
 10 DATA 192,223,123,254,202,192,239,215
 11 DATA 130,28,223,13,215,153,30,96
 12 DATA 105,215,110,25,40,2,231,141
```

```
 13 DATA 34,19,62,35,35,78,35,70
 14 DATA 120,177,32,2,231,141,3,3
 15 DATA 3,3,237,67,17,62,237,67
 16 DATA 21,62,62,1,50,5,62,62
 17 DATA 68,50,4,62,62,0,50,16
 18 DATA 62,223,61,192,223,44,42,19
 19 DATA 62,237,91,17,62,237,83,170
 20 DATA 59,223,39,223,66,223,14
```

De volgende CREATE file is een sector checker aan te roepen via VERIFY @D<drive> ,eerst de assembler listing en daarna het basic blokje met DATA.

```
;disc checker routine.....

            ORG  60000
            DUMP 60000

;VERIFY @  D<drive>

start       DEFB 1
            DEFB #D6                ;VERIFY
            DEFW verif_len

1_verif     CP   "@"                ;use '@'
            RET  NZ
            RST  #28
            AND  223                ;capitals
            CP   "D"                ;use 'D' or 'd' for drive
            RET  NZ
            RST  #28
            RST  #18
            DEFB 120                ;ask for d1,d2 or d*
            RST  #18
            DEFB 13                 ;syn.end

REF         RST  #18
            DEFB 108                ;relocate
            DEFW REF
            DEFB R1+1-REF
            DEFB R2-R1
            DEFB R3-R2
            DEFB R4-R3
            DEFB R5-R4
            DEFB R6-R5
            DEFB R7-R6
            DEFB R8-R7
            DEFB 0

            RST  #18
            DEFB 100                ;open channel to screen
```

```
            RST  #18
            DEFB 111                ;test for drive
            LD   A,0                ;track number side one
R1          LD   (TRACK+1),A
            INC  A                  ;sector number side one
R2          LD   (LDTR1+1),A
            LD   A,80               ;allready to high
R3          LD   (LAST+1),A
R4          CALL TRACK              ;call sector checker
            LD   A,128              ;track number side two
R5          LD   (TRACK+1),A
            LD   A,1                ;sector number side two
R6          LD   (LDTR1+1),A
            LD   A,208              ;allready to high
R7          LD   (LAST+1),A
R8          CALL TRACK              ;call sector checker
            RST  #18
            DEFB 14                 ;com.end

TRACK       LD   D,0                ;start track
LDTR1       LD   E,1                ;start sector
LDSC1       PUSH DE                 ;put this away
            RST  #18                ;call unidos
            DEFB 18                 ;command code'load sector'

            POP  DE                 ;get value DE register
            INC  E                  ;increase sector
            LD   A,E                ;put in A register
            CP   11                 ;compare with 11
            JR   C,LDSC1            ;if no then next sector
            INC  D                  ;increase track
            LD   A,D                ;put in A register
LAST        CP   80                 ;compare with last track
            JR   C,LDTR1            ;if no then next track
            RET

verif_len EQU $-1_verif
```

# INPUT / OUTPUT




## DE FUNTIES VAL EN VAL$

In 1990 stond er in het "IMPULS" het hierna volgende artikeltje. Door omstandigheden verscheen de reactie op dat artikel pas afgelopen maand in het "IMPULS". Omdat in deze reactie ook weer een oproep om hulp cq. tips stond, en het "IMPULS" inmiddels niet meer verschijnt, leek het ons handig en nuttig om beide artikelen in het BULLETIN te herhalen.
STUUR UW TIPS AAN HET REDACTIE-ADRES. (zie pag. 2).

## IMPULS 82-20

door: E. Weijgers

Nooit heb ik goed begrepen wat nou eigenlijk het nut zou kunnen zijn van de functie VAL$. De SP-handleiding geeft wel een aantal voorbeelden om te proberen (veel gehannes met aanhalingstekens). Het klopt ook allemaal wel, maar toepassingen waar je iets aan hebt vond ik er niet bij, en die kon ik niet bedenken ook.
De functie VAL gebruik ik echter geregeld, en niet alleen om de FPR's te vermijden om geheugenruimte te besparen. Toch ligt bij VAL de sleutel om VAL$ beter te begrijpen, zoals we zullen zien.

Stel dat N$ de naam BEVAT van een ongeDIMde numerieke variabele. Dan stelt VAL N$ de getalwaarde van die variabele voor.

```
LET Jaartal=1900+90:          hierdoor verschijnt
LET N$=" jaar tal ":          op het scherm:
PRINT n$;"=";VAL n$            jaar tal =1990
```

Stel dat N$ de naam bevat van een ongeDIMde stringvariabele. Dan stelt VAL$ N$ de stringwaarde van die variabele voor.

```
LET T$="titel "+STR$ 6:       hierdoor verschijnt
LET N$=" t$":                 op het scherm:
PRINT n$;"=";VAL$ n$           t$=titel 6
```

(Hieruit blijkt dat er bij NAMEN van variabelen geen onderscheid bestaat tussen hoofdletters en kleine letters. Ook blijven alle spaties voor, in en achter die namen altijd buiten beschouwing.)

TOEPASSING

We kunnen een programma schrijven waarbij het variabelengeheugen (VARS t/m E_LINE-1) doorlopen wordt door een adresvariabele A. De naam van een variabele kunnen we telkens in N$ zetten mbv oa:

```
LET T=INT (PEEK A/32): LET N$=CHR$ (PEEK A-T*32+96)
```

Hierna dient N$ nog aangevuld te worden met "$" bij T=2 en T=6, of met de resterende tekens van een langere naam wanneer T=5.

Voor de waarde behoeven we echter niet te PEEKen, dat gaat daarna eenvoudig mbv VAL N$ of VAL$ NS (afhankelijk van het type T). We kunnen de naam van een variabele ook een index I meegeven:

```
VAL$ (N$+"("+STR$ I+")")          [Verwar dit niet met N$(I)]
```

Natuurlijk kunnen er ook allerlei andere combinaties van stringwaardige functies en operaties gebruikt worden (ook binnen N$).

—

Eindelijk eens een echte toepassing van VAL$ ontdekt! Toegegeven dat deze niet erg belangrijk is, maar misschien is er iemand die er meer kent, of hierdoor op een idee komt. Laat het ons weten. Voor de functie SGN heb ik overigens ook nooit iets gevonden wat niet op een andere manier handiger ging. U wel? - EdW -

Het volgende artikel is hier een reactie op.

## IMPULS 93/94 — 65

door: W.G. Haggenburg.

In IMPULS 82-20 werd een toepassing van VAL$ gepubliceerd, met de vraag naar andere gebruiksmogelijkheden. Ik had er eentje.

—

In een programma van mij kwam een constructie in deze trant voor (om diverse redenen kon ik slechts ongeDIMde strings gebruiken):

```
xx10 IF c=1 THEN LET x$=a$
xx20 IF c=2 THEN LET x$=b$
xx30 IF c=3 THEN LET x$=c$
 ...
 t/m
 ...
xx90 IF c=9 THEN LET x$=i$
```

Na vele hoofdbrekens vond ik een vrij simpele manier om al deze negen regels door een enkele regel te vervangen:

```
xx10 LET x$=VAL$ "a$b$c$d$e$f$g$h$i$"(2*c-1 TO 2*c)
```

Eindelijk een nuttige bestemming voor de VAL$-functie, waarvan het nut mij tot nu toe ontgaan was!

—

[Voor de volledigheid wil ik hier nog een mogelijkheid aangeven:

```
xx10 LET x$=(a$ AND c=1)+(b$ AND c=2)+ ... +(i$ AND c=9)
```

Die regel van Wil is evenwel veel korter en veel eleganter. -EdW-]

**Maar nu het omgekeerde, ook dat deed zich voor:**

```
xx10 IF c=1 THEN LET a$=x$
xx20 IF c=2 THEN LET b$=x$
xx30 IF c=3 THEN LET c$=x$
 ...
 t/m
 ...
xx90 IF c=9 THEN LET i$=x$
```

**Naarstig experimenteren, oa ook met VAL$, leverde hiervoor geen bekorting op. Wie helpt? Ikzelf kom niet verder dan deze weinig fraaie constructie:**

```
xx10 POKE PEEK 23637+256*PEEK 23638+5,96+c
xx20 LET a$=x$
```

**Mbv de systeemvariabele NXTLIN wordt hierdoor het programma zelf veranderd, hetgeen een niet erg aanbevelenswaardige techniek is.**

**——— Wil G Haggenburg**

Het is natuurlijk altijd moeilijk om voor een ander oplossingen aan te dragen, zonder dat je de 'aanleidingen' tot het probleem kent. Je loopt zo wel het risico een zeerbekende open deur in te trappen. Maar ja, een oproep hoort natuurlijk wel beantwoord te worden.

Jammer dat de hiervoor genoemde methodes, nl.
LET x$= VAL$ "a$b$. enz" (c*2-1 enz) en
lET x$=(a$ AND c=1) enz.
niet ook omgekeerd kunnen werken. Nu blijft alleen het bekorten nog over. Structureel programmeren is misschien een oplossing ??

Er is blijkbaar al gecontroleerd of de variabele c aan de toegestane waarden heeft voldaan, dus dan kunnen de volgende twee regelblokjes mischien een oplossing zijn.

```
xx05 GO TO xx00 + c*10
xx10 LET a$=x$: GO TO x100
xx20 LET b$=x$: GO TO x100
xx30 LET c$=x$: GO TO x100
 .... t/m ....
xx90 LET i$=x$: GO TO x100
```

```
xx05 GO SUB xx00 + c*10
xx10 LET a$=x$: GO TO x100
xx20 LET b$=x$: GO TO x100
xx30 LET c$=x$: GO TO x100
 .... t/m ....
xx90 LET i$=x$: GO TO x100
x100 RETURN
```

Het eerste blok (GOTO) is niet veel korter als al die IF/THEN-regels maar het programma loopt wel veel sneller.

Het tweede blok zou nog iets korter kunnen door i.p.v. telkens GOTO x100 rechtstreeks te RETURNen. Regel x100 is dan niet nodig Met GOTO x100 is de FLOW-structure wel duidelijker.

Deze GOSUB-lus kan gemakkelijk nog uit gebreid worden. Als het programma op meerdere plaatsen de IF/THEN structuur had staan, kunnen ze vermoedelijk op deze manier allemaal door één GOSUB-routine vervangen worden. Dat maakt het dan wel (veel) korter.

JWK.

# MCODE: STAP VOOR STAP (11)

Uit ZX-Computing afl.: juni/juli '85, blz. 74 e.v..
vertaling: J. v. Alteren, SGG; 070393.

**DEEL 4: Bits en Stacks.** (vervolg):

Spectrum voorbeelden:

Bij de spectrum-voorbeelden gebruiken we dezelfde mc-laadroutine als in deel 3. (Kijk maar eens, ik heb een andere mening: JvA) de CLEAR-opdracht verlaagt de RAMTOP en plaatst de STACK onder de nieuwe RAMTOP, zodat je een veilig RAM-deel hebt om in te werken.
Typ de laadroutine in (aan het eind van deze afl.) dan de dataregels uit het eerste voorbeeld toevoegen die je wilt uitproberen.

De eerste routine tast de display-file af en draait elke bit om. Dat wil zeggen, 0 wordt waarde 1 en 1 wordt waarde 0. en dus verandert PAPER in INK. De belangrijke opdracht hiervoor is XOR 255.

De tweede routine tast de file met kenmerken af en schakelt de FLASH-bit om. Dat houdt in, dat de stand 0 wordt omgewisseld in 1. of omgekeerd.
Dus, van een niet-knipperend scherm gaat je hele scherm wel knipperen. **Proberen !**

Beide SPECTRUM-vorbeelden geven een manier om een grote fout in de Z80 te omzeilen. Tot ieders verrassing (nu ook nog tot de mijne), zal de ZERO-FLAG niet teruggezet worden . als je de DEC-opdracht in een 2-byte adres gebruikt en het dubbelle adres de waarde 0 bevat.
Dus dan zal. als je laten we zeggen BC als teller gebruikt, JR NZ na DEC BC niet werken als BC 0 wordt. Het is een begrijpelijke vergissing bij beginners in mc om zonder meer aan te nemen dat dat wel zal gebeuren. De manier hoe dit toch voor elkaar te krijgen ziet u in de voorbeelden

Als je DEC BC laat doen dan zijn/worden de waarden/standen van B en C geORed. (nadat de waarde van B naar het A-adres is overgebracht. Alleen als B EN C op 0 staan, zal het gevolg van de OR-opdracht 0 zijn. De ZERO-flag wordt veranderd, als het resultaat van een logische opdracht 0 is, dus je kan de FLAG pas controleren, NA de OR-opdracht.

Hierna volgen nog eens de antwoorden van de experimenten uit mijn vorige aflevering 10.

AND: 00010001 (17); OR: 11101110 (238); XOR: 10100110 (166).

Als je het niet met deze antwoorden
eens bent, (of je hebt andere uitkomsten), werk je nog eens door
de voorbeelden van fig 1 en 2 in mijn aflvering 9 en 10 heen.

Hierna fig. 4 met de mc-listing erna.

Fig. 4 SPECTRUM-machinecode voorbeelden.

```
10 CLEAR 24999
20 LET i = 25000
30 READ x: IF x = -1 THEN GOTO 200
40 POKE i,x:LET i = i + 1
50 GOTO 30

  **** Hier de DATA-regels van de voorbeelden plaatsen ****

190 DATA -1
200 CLS
210 FOR i = 1 TO 10
220 PRINT "abcdefghijklmnopqrstuvwxyz"
230 PRINT "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
240 NEXT i
250 RANDOMIZE USR 25000
260 IF INKEY$ = ""****N GOTO 260
270 GOTO 250
290 REM
300 SAVE "SPEC-VBLD": PRINT "SPOEL terug": VERIFY ""
```

---

**Eerst mc-voorbeeld:**

```
100 DATA 33,0,64,1,0,24,126
110 DATA 238,255,119,35,11,120
120 DATA 177,32,246,201
```

Listing in MACHINE-TAAL:

```
        LD HL,16384       210040    HL = Display File start
        LD BC,6144        010018    Bytes in display-file / x
LOOP    LD A,(HL)         7E        Overbrengen bytes naar A
        XOR 255           EEFF      Bits omdraaien
        LD (HL),A         77        De byte terugplaatsen
        INC HL            23        Volgende byte
        DEC BC            0B        Teller verlagen
        LD A,B            78        Test of ze 0 zijn
        OR C              B1
        JR NZ, LOOP       20F6      Indien niet, herhaling
        RET               C9        Anders terug naar BASIC
```

---

**Tweede mc-voorbeeld:** (voor de SPECTRUM dus)

```
100 DATA 33,0,88,1,0,3N126,198,128
110 DATA 119,35,11,120,177,32,246,201
```

Listing in MACHINE-TAAL:

```
        LD HL,22528       210058     Start van het merken
        LD BC,768         010003     Getal bij de merk-bytes
LOOP    LD A,(HL)         7E         Laad A met die kenmerken
        ADD 128           C680       Schakel FLASH aan/af
        LD (HL),A         77         Zet de byte terug
        INC HL            23         Volgende byte
        DEC BC            0B         Verlagen van de teller
        LD A,B            78         Controle  van 0-stand
        OR C              B1
        JRNZ, LOOP        20F6       Indien niet, herhaling
        RET               C9         Anders terug naar BASIC
```

---

Zo, deze keer maar een kort gedeelte om de voorbeelden uit te proberen.
Vorige keer voor de ZX81, nu voor de SPECTRUM. Ik ben benieuwd naar Uw ervaringen, dus schrijf eens een reactie

naar: J. v. Alteren, 9351 LP Leek. (of in het Bulletin.)

J. v. Alteren.

---------- WORDT VERVOLGD ------

# ESGEEGEEJES

Wie kan mij helpen aan een muis voor de Sneider PC.
De gangbare PC-miuzen accepteren de bijbehorende software wel, maar de muispijl op de monintor beweegt niet.
Wie helpt mij verder, of heeft een idee waar het fout gaat.
Graag reacties naar F. (Flora) Elstrodt.
Boelemaheerd 151, 9736 HJ Groningen, tel.: 050-424662 .

3. Break het programma zodra het inlezen klaar is en type in:
   CLS: LIST 9000 <ENTER>;
4. Schrijf regel 9.390 en 9.400 over. Op regel 9.390 staat een "centreerformule" en regel 9.400 POKEt de variabelen;
5. Geef een NEW om het lesprogramma te wissen en type in:

```
9990 CLEAR 32255: LOAD "GLC" CODE: LET d$="STOP DE BAND":
     LET y=80: LET xs=1: LET ys=2: LET cs=8: GOSUB 9995: STOP
9991 CLEAR: SAVE "GL" LINE 9990: SAVE "GLC" CODE 32256,300:
     VERIFY "GL": VERIFY "GLC" CODE: STOP
9995 REM Regel 9390
9996 REM Regel 9400
RUN 9991 <ENTER>
```

Eventueel kunt u de LOAD/SAVE/VERIFY statements aanpassen aan uw opslaginterface. Ook de variabelennamen mag u veranderen. Hun betekenis luidt:

```
X  = X coordinaat linkerkant van de tekst (hires)
Y  = Y     "      bovenkant   "   "   "      "
XS = breedte van de tekst
YS = hoogte   "   "   "
CS = breedte van de matrix (pixels)
D$ = af te drukken tekst
```

Denkt u er wel aan, dat er in de printerbuffer gepookt wordt. De maximale lengte van de tekst is 240 tekens (23306+5 eerste pos. en 23552-1 laatste).

Door de variabele cs 4, 5 of 6 te maken kunt u ook resp. 64, 51 of 42 kar/reg. afdrukken. U moet dan een geschikte karakterset achter de PSION code zetten en CHARS op 23606/7 daarnaar laten verwijzen. Er voor kan wel, maar dan moet RAMTOP erg ver omlaag. Die karakterset moet gebruik maken van de hoogste bits (links). Als u die niet heeft, kunt u er ook zelf eentje maken door 5 sets UDG's te fabrieken en die achter elkaar te plakken. Van die laatste set gebruikt u dan alleen de eerste 12 letters. Een hulpprogramma zit in de HORIZONS serie, maar ook in de Users Guide Companion Cassette bij de Plus.

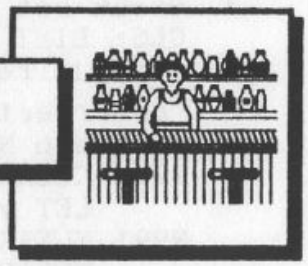
*KLOK*

Vele jaren geleden werd in RAM een klokprogramma van de hand van J. Kuiper gepubliceerd dat werkte met Beta BASIC 3 en 4. (Standaard of opslaginterface versie). Er bestaat echter een veel kortere manier om een beeld"vullende" klok te programmeren:

```
 1 DO
     PRINT AT 10,0;
     PRINT CSIZE 32; TIME$()
   LOOP
```

Eventuele franje mag u hier zelf omheen maken.

# BIJEENKOMSTEN



In het: RPV gebouw
RABENHAUPTSTRAAT 45
GRONINGEN
Telefoon: 050-261379

DATA GRONINGEN:

| | | |
|---|---|---|
| 22 februari | dinsdag | van 19.30 - 22.30 |
| 19 maart | zaterdag | van 14.00 - 17.30 |
| 19 april | dinsdag | van 19.30 - 22.30 |
| 21 mei | zaterdag | van 14.00 - 17.30 |
| 21 juni | dinsdag | van 19.30 - 22.30 |

N.B. bij juni staat (nog) niet ledenvergadering.
Het is voor de leden die wat verder weg wonen, misschien prettiger om de vergadering op een zaterdagmiddag te houden. b.v. in mei, juni of september. Laat ons eens uw mening hierover weten. Schrijf of bel even naar het redactie adres.

---

HCC-bijeenkomsten:

| | | |
|---|---|---|
| 9 maart | zaterdag | van 10.15 - 15.30 |
| 18 juni | zaterdag | van 10.15 - 15.30 |

plaats: de Verenigingszaal. Gereformeerde Kerk.
Cammingalaan , te BUNNIK.

---

LEES OOK pagina 3 en pagina 23

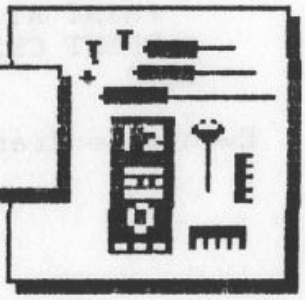
---

# SPECTRUM EMULATOR

Staat u nog niet als gebruiker geregistreed, en wilt u toch graag de nieuwe versie (2.01a) van de Emulator aanschaffen, dan kan dit als volgt:

Maak f 25,-. over op: Girorekening 5945263
t.n.v. Gerton Lunter
met vermelding Emulator Update

# REPARATIE ADRES



A. HOEKMAN,
DE HENNEPE 351
4003 BD TIEL.
TEL: 03440 - 32182

# BEURZEN / HANDEL



## COMPUTERBEURZEN :

| | | |
|---|---|---|
| Zaterdag 19 februari! :<br>NB. Let op !<br>Dit is de Juiste Datum | | 5e COMPUTERBEURS voor het NOORDEN<br>in de MARTINIHAL in GRONINGEN.<br>van 10.00 tot 17.00. |
| Zaterdag 22 en<br>Zondag 23 JANUARI | :<br>: | 10e COMPUTERDAGEN voor OOST-NEDERLAND<br>in de IJSSELHAL in ZWOLLE.<br>van 10.00 tot 17.00 uur. |
| Zaterdag 26 februari | : | het NOORDELIJK AMATEUR TREFFEN.<br>(trefpunt en beurs van radio- en<br>zendamateurs uit NED. en DUITSL.)<br>in de MARTINIHAL te GRONINGEN<br>van 9.30 tot 16.30 uur. |
| Zaterdag 26 maart | : | 5e. FRIESE HCC-dag.<br>in ZAAL PRINS , d.i. in de voorzijde<br>van de FRIESLANDHAL in LEEUWARDEN.<br>van 10.00 tot 17.00. |
| Zaterdag 26 maart | : | COMPUTER BEURS ASSEN.<br>in de Sport en Evenementenaccommodatie<br>"de SMELT" te ASSEN.<br>van 10.00 tot 16.00. |
| Zaterdag 26 maart | : | 3e ZUIDHOLLANDSE COMPUTERBEURS<br>in de WEENAHAL (bij Centraal Station)<br>te ROTTERDAM. van 10.00 tot 17.00 uur. |
| Maandag 4 april<br>(= 2e PAASDAG) | : | 1e internationale consumenten COMPUTER<br>BEURS in het MECC te MAASTRICHT.<br>van 10.00 tot 17.00 uur. |

Enkele van de beurzen worden georganiseerd door het bureau "EXPO PARTNERS", en zijn gericht op de PC-gebruiker, maar als SPECTRUM-gebruiker kunt u er natuurlijk ook wel rondneuzen, op zoek naar b.v. een printer, monitor, joystick, diskettes, stickers enz. Vermoedelijk kunt op op de beurs in Groningen ook een van onze redactieleden vinden achter de HCC stand. Zo dat u met eventule vragen of problemen geholpen kunt worden.

## HANDEL :

U zoekt 2e. handse apparatuur en/of boeken, handleidingen of software. Misschien hebt u bij mij succes.

F.P. Vink , Zuiderkruis 49, Veenendaal. tel: 08385 - 50708 .

DRUKWERK

C. M. BALLINTYN
B. BOERMALAAN 7
9765 AP PATERSWOLDE