

DISCIPLE

1e JAARGANG Nr.4

MET O.A:

TASWORD II - Maatwerk.....!!

OPENTYPE FILES, hoe werkt het

OPENTYPE BESTAND

A4 CAT - PLUS

BITMAP - 3

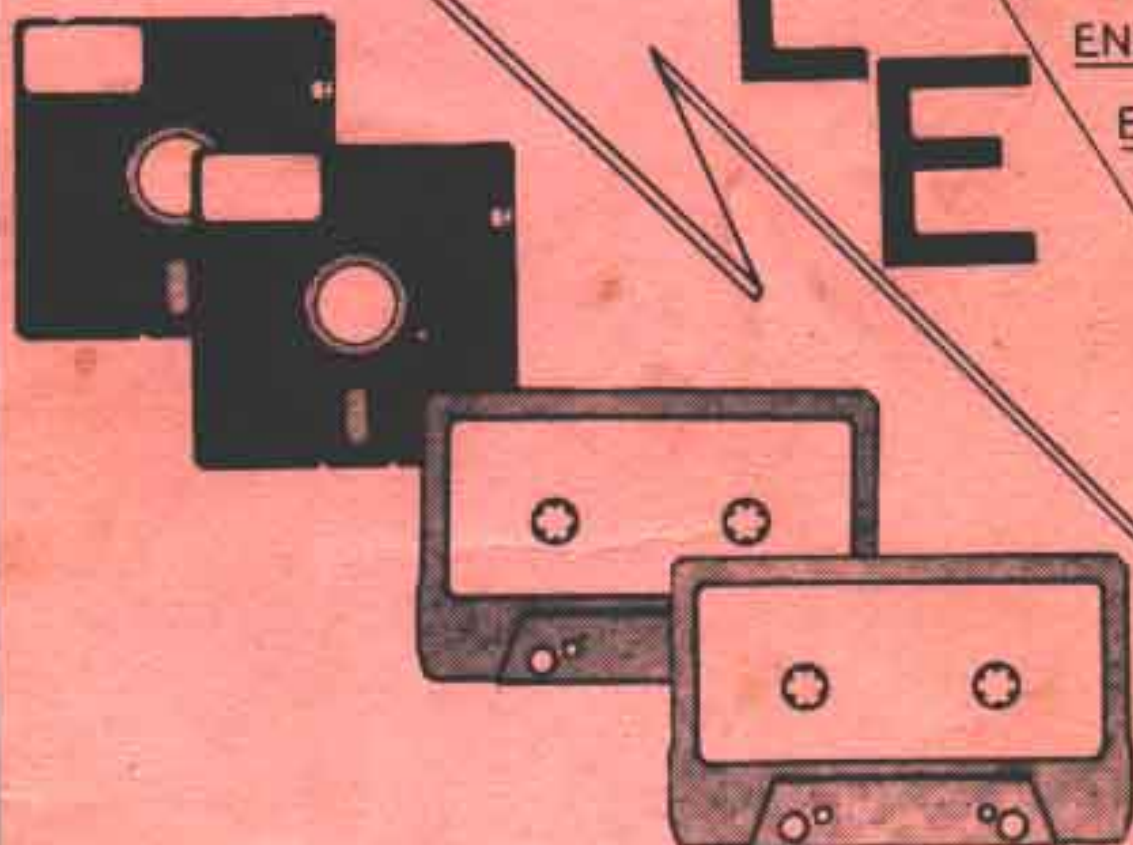
AUTOLOADER

NAAM - CAT

VARIA

ENZ.....

ENZ.....



NIEUWSBRIEF

>>>>>> DISCIPLE NIEUWSBRIEF <<<<<<<<

REDACTIONEEL

Redactie : Disciple Nieuwsbrief,
Peter Faas,
[redacted]
[redacted]

Vormgeving Carl Faas,
en Layout: [redacted]
[redacted]

Gerard Nesselaar
[redacted]
[redacted]

Copy : Tasword 2 files of programma's op 5 1/4 inch schijf
of cassette vergezeld van een hardcopy.
Formaat, SD of DD op de schijf vermelden.
Bewaar een kopie ingeval de PTT verstek laat gaan!
Stuur een gefrankeerde antwoord-enveloppe mee als u de
schijf of cassette terug wilt hebben, en vergeet niet
overal naam en adres op te vermelden.

Svp GEEN artikelen uit andere bladen overschrijven.

>>>>>> : HOE IS DE DISCIPLE NIEUWSBRIEF TE VERKRIJGEN??
- Op de bekende SPECTRUM dagen in de BRON te Utrecht.
- Bij DATASKIP te Gouda en bij KOMIN in Eindhoven.
- Door overschrijving per Bank of Giro, Giro [redacted]
t.n.v. P.Faas, Red. Nwsbr. te [redacted].
Vermeld duidelijk gewenst nummer en TAPE of DISC.

Prijs : Afgehaald; (de Bron, Dataskip of Komin)
-ZONDER disc of tape Fl 5.-
-MET disc of tape Fl 7.50
Per post (via Bank/Giro en incl.porto)
-ZONDER disc of tape Fl 7.45
-MET disc of tape Fl 9.95

>>>>>> : ALGEMEEN INFORMATIE Nr.: [redacted]

Copyright: De redactie aanvaardt geen aansprakelijkheid voor de
overtreding van COPYRIGHT door inzenders.
Inzenders worden geacht zich te houden aan de
geldende regels met betrekking tot het COPYRIGHT van
anderen.

Noot : Alle publicaties in de Disciple Nieuwsbrief zijn
vrij van copyright.

Adverteren: Niet commerciële advertenties gratis binnen de be-
schikbare ruimte.

REDACTIONEEL (COLOFON).....	4-1
INHOUDSOPGAVE/DISCIPLE VARIA.....	4-2
AUTOLOADER/LISTING AUTOLOADER.....	4-3
TASWORD II EN DE DISCIPLE/PLUS-D.....	4-4/4-11
REDACTIONELE NOOT.....	4-11
UITGEBREIDE A4 - CAT/VOORBEELD.....	4-12
LISTING UITGEBREIDE A4 - CAT.....	4-13
ADRESSENBESTAND "RECORD versie 1.3".....	4-14/4-15
DISCIPLE VARIA'S.....	4-15
WERKEN MET OPENTYPE FILES.....	4-16/4-17
NAAM-CAT/LISTING NAAM CAT.....	4-18
BITMAP 3.....	4-19/4-21
UITDRAAI VOLLEDIGE BITMAP.....	4-22
BASICODE EN DE PRINTER.....	4-23
VOORUITBLIK NIEUWSBRIEF 5.....	4-23
COMPLETE VERZAMELING FOOTBOODSCHAPPEN IN GDOS.....	4-24

VARIA * DISCIPLE * VARIA * DISCIPLE * VARIA * DISCIPLE * VARIA *

Rob Duursma zoekt contact met Disciple gebruikers in de provincie Utrecht of omgeving voor het uitwisselen van ervaringen en informatie. [REDACTED]

De Firma KOMIN uit Eindhoven is op ons verzoek bereid gebleken de lezers van de DISCIPLE NIEUWSBRIEF een bijzonder aantrekkelijk aanbod te doen voor de aanschaf van een DUBBELE DISC DRIVE. Deze drive bestaat uit een 5.25" en een 3.5 inch unit, 80 tr. DS DD dus samen goed voor 1560K.....!!! Het geheel is compact en met voeding ingebouwd in een mat zwarte kast, alle kabels zitten er standaard op, stekker in het stopcontact, schijf in de drive en runnen maar. Voor Nieuwsbrieflezers geen Fl.950.- maar Fl.750.- !! Fa. KOMIN te Eindhoven (zie advertentie)

Wist u dat de 8K rom van de Disciple voorbereid is om vervangen te worden door een 16K exemplaar? Adreslijn A13 verzorgt de schakeling tussen de 2x 8K (een soort bankswitching) Het commando OUT 31,8 is er voor gereserveerd. Voorlopig zoiets van " wat in het ROM vat zit, verzuurd niet".

Trouwens, wie een hekel aan "systeemfile laden" heeft kan desgewenst de statische RAM vervangen door een EPROM, met ingeprogrammeerde systeemvariabelen. Wel even pen 27 van deze EPROM op de +5V aansluiten en pen 26 en 28 doorverbinden. Zinvol? welnee! maar wel leuk om te weten.

AUTOLOADER

Dit autoloader programma heeft bovenin een statusbalk met de opties LOAD, ERASE en CAT!'. Een cursorbalkje kan op een van deze opties geplaatst worden met behulp van de toetsen 5 en 8 of de Kempston joystick. Als u de optie LOAD of ERASE kiest en daarna de joystick naar beneden beweegt, cq toets 6 indrukt, dan beweegt een tweede cursorbalkje zich over de programma namen. Dmv <SPACE> kan de juiste keus gemaakt worden. Gebruikt u de 128K Spectrum en wilt u liever met de cursortoetsen werken, ipv de Kempston optie, wijzig dan het programma als volgt:

```
9550 LET I$=INKEY$: LET K=CODE I$ : IF I$ etc.  
9560 K=162 in K=8  
9570 K=161 in K=9  
9580 K=164 in K=10  
9590 K=168 in K=11  
9610 IF I$=" " AND V>0 THEN GO TO 9630
```

Ook hier geldt de beperking dat er niet te veel programma's op schijf mogen staan om de Scroll? boodschap te vermijden en dat de programmanummers OPEENVOLGEND moeten zijn. Niet ERASEN dus.

```
9500 REM RUN PROGRAMMA  
9510 LET E$=" " : LET DO=3: LET K$=" LOAD ERASE  
CAT 1' "  
9520 IF DO=3 THEN CLS : PRINT K$( TO 22); INVERSE 1;K$(23 TO );:  
CAT 1': LET V=0: LET H=0: LET DO=1: LET PR=0  
9530 IF DO=2 THEN PRINT AT 0,0;K$( TO 11); INVERSE 1;K$(12 TO 21  
); INVERSE 0;K$(22 TO )  
9540 IF DO=1 THEN PRINT AT 0,0: INVERSE 1;K$( TO 10); INVERSE 0;  
K$(11 TO )  
9550 LET I$=INKEY$: LET K=IN 31: IF I$=" " AND K=160 THEN GO TO 9  
550  
9555 LET VE=V: LET HO=H  
9560 IF (I$="5" OR K=162) AND H>0 THEN LET H=H-11: IF V=0 THEN L  
ET DO=(1 AND H=0)+(2 AND H=11)+(3 AND H=22): GO TO 9520  
9570 IF (I$="8" OR K=161) AND H<22 THEN LET H=H+11: IF V=0 THEN  
LET DO=(1 AND H=0)+(2 AND H=11)+(3 AND H=22): GO TO 9520  
9580 IF (I$="6" OR K=164) AND V<21 THEN LET V=V+1: IF V=1 THEN L  
ET V=3: PRINT AT V,H: OVER 1: INVERSE 1;E$( TO 10): GO TO 9550  
9590 IF (I$="7" OR K=168) AND V>0 THEN LET V=V-1: IF V=2 THEN LE  
T V=0: PRINT AT 3,H: OVER 1: INVERSE 1;E$( TO 10): LET DO=(1 AND  
H=0)+(2 AND H=11)+(3 AND H=22): GO TO 9520  
9600 IF V<>VE OR H<>HO THEN PRINT AT VE,HO: OVER 1: INVERSE 1;E$  
( TO 10);AT V,H: OVER 1: INVERSE 1;E$( TO 10)  
9610 IF (I$=" " OR K=176) AND V>0 THEN GO TO 9630  
9620 GO TO 9550  
9630 LET I$=" ": FOR K=0 TO 9: LET T$=SCREEN$(V,(H+K)): LET I$=I  
$+T$: NEXT K: IF I$=E$( TO LEN I$) THEN GO TO 9550  
9640 IF DO=2 THEN PRINT AT V,H: INVERSE 1;E$( TO 10): ERASE d1;I  
$: GO TO 9550  
9650 LET PR=(V-3)*3+(1 AND H=0)+(2 AND H=11)+(3 AND H=22)  
9660 LOAD P PR
```

A. Hol

(listing via Basic-tas)

TASWORD II en de DISCIPLE/PLUS-D

De old-timer Tasword II is nog steeds de favoriete tekstbewerker voor de Spectrum. Dat is niet zonder reden. Toch valt er het een en ander aan dat programma te verbeteren. En zolang de conversie van Tasword III alles behalve volmaakt blijkt zullen we het met Tasword II moeten doen. Vooral in combinatie met de Disciple en de Plus D is het gebruiksgemak van TASWORD onovertroffen.

SPECTALE VERSIES

Voor al die mensen die met spanning op Tasword III wachten het volgende. Er is een conversieprogramma. Dat werkt echter met bijna geen enkele Spectrum. De oorzaak is al die verschillende Spectrum "Issues". Het programma mist momenteel de intelligentie om zelf te kunnen bepalen welk type Spectrum u gebruikt. Zolang die problemen niet zijn opgelost wordt het conversieprogramma niet verkocht. Denk hierbij ook aan BETA BASIC. Uit de handleiding weet u wel dat ook dat programma zelf "kijkt" welke hardware er in gebruik is. Telkens wanneer u van hardware wisselt, zoals bijv. een andere Interface One moet u in negen van de tien gevallen ook een nieuwe kopie van uw origineel maken. De makers van BETA BASIC kampen dan ook met hetzelfde probleem. De speciale Discipleuitgave van dat programma werkt nog lang niet volmaakt met alle denkbare Spectrumcombinaties. De problemen gaan daar zelfs "dubbel" op. Welke DISCIPLE hebt u, welke SPECTRUM en zoja welk INTERFACE ONE. Ook hier dus weer, die gedeeltelijk werkende versie van BETA BASIC wil men liever nog niet verkopen. Dat is vragen om narigheid. Over nu naar TASWORD.

DE EXTRA MOGELIJKHEDEN

Dat zijn er nogal wat. Een aantal zijn trouwens van Tasword III afgekeken. De extra mogelijkheden zijn als hieronder aangegeven.

SAVE:

- Wegschrijven van tekstblokken van regel tot regel.
- Wegschrijven onder dezelfde naam.

PRINT:

- Aantal afdrukken.
- Instellen linker kantlijn.

CATalogus:

- Volledige cursorbesturing. U hoeft nu geen namen meer in te typen bij wissen, inlezen of toevoegen van een document.
- Het programma op schijf een en de tekstbestanden op schijf twee

ALGEMEEN:

- Variabelen zoals het aantal afdrukken, de regelafstand en voorkeur diskdrive worden permanent bewaard.
- Met de EDITtoets sluit u de vragenlijst voor afdrukken en wegschrijven van bestanden kort.
- Opgave van het aantal getypte woorden (wordcount).
- Ruimte voor uw eigen toevoegingen aan het programma.

DEELPROGRAMMAS OF OVERLAYS.

Om ruimte te maken voor al die extra faciliteiten is TASWORD in een aantal deelprogrammas gehakt. Dat is gezien de verwerkings-snelheid van de Disciple geen enkel probleem. Met uitzondering van wijlen het BETA Interface en natuurlijk de Disciple en het PLUS D interface is deze methode dan ook met geen der andere opslagsystemen echt handig in het gebruik. Keuze "g" vanuit het hoofdmenu maken we zelden. Het bijbehorende programmablok staat dan ook als deelprogramma "twgr/pr" op schijf.

We kunnen ons hierbij de vraag stellen: "Moet dit in de vorm van een Overlay of als Deelprogramma". Er is een groot verschil. Een "Deelprogramma" verdient verreweg de voorkeur. Met LOAD halen we telkens vanuit het ene deelprogramma het volgende binnen. Naar de regelnummering hebben we geen omkijken. De variabelen vormen echter een struikelblok. Na een LOAD zijn die immers allemaal verdwenen. Na MERGE blijven die waarden weliswaar behouden maar zonder hulp van BETA BASIC luistert de regelnummering erg nauw. Wanneer er ook maar een regel van de vorige overlay blijft staan kan uw hele programma in de soep lopen. De term "Overlay" zal u nu ook wel duidelijk zijn. Het nieuwe deel moet precies, dus regel voor regel, over het vorige deel liggen. In uw eigen programmas is dat onder BETA BASIC een stuk eenvoudiger. Met DELETE wist u dan eerst uw oude blok. Het nieuwe deel hoeft dan slechts vanaf de eerste regel aan te sluiten. Een voorbeeld:

```
10 REM uw hoofdprogramma
20 REM keuzemenu
30 REM ken afh. van uw keuze een naam toe aan a$: 60 TO 980
40 REM rest hoofdprogramma
900 GO TO 10
980 DELETE 1000 TO 2000
990 MERGE d*,a$: 60 TO 1000
```

Er kleeft nog een groot bezwaar aan Overlays. Het inlezen gaat snel genoeg. Daarna moet de Spectrum de regels nog op de juiste plaats invoegen. Bij een lang programma kan dat wel 20 seconden duren. Dat is onaanvaardbaar. Wanneer u i.v.m. variabelen toch van Overlays gebruik moet maken zorgt u ervoor dat ze vooral kort zijn. Hoe het soms ook anders kan ziet u straks. Bij Tasword hebben we geluk. De printeraanpassingen worden alle direct met POKES in het codeblok boven ramtop gezet. Er zijn hier geen zwevende variabelen. We kunnen hier dan ook straffeloos van een deelprogramma gebruik maken. Er volgt nu wel een extra vraag of we de aangebrachte wijzigingen op schijf willen vastleggen.

ALLEEN OP SCHIJF.

Het Basic gedeelte van Tasword II is nogal "springerig" en ook de regelnummering verloopt erg onregelmatig. Uit welingelichte bron weten we dat ook de machinecode van hot naar haar springt. Daar blijven we dus liever af. Het programma is dus eerst gefatsoeneerd met de RENUMBER uit BETA BASIC. Dat is immers de enige hernummer utility voor de Spectrum waarmee je ook programmablokken kunt verplaatsen. Het nadeel is echter dat we niet kunnen

vervolg van "TASWORD II en de DISCIPLE/PLUS-D"

volstaan met het afdrukken van een korte listing met de nieuwe programmaregels. Daarvoor zijn de aanpassingen ook te veelomvattend. U moet dus de bij deze Nieuwsbrief behorende schijf aanschaffen. Daarop vindt u alle programmas. Uiteraard met uitzondering van de oorspronkelijke machinecode. Die wordt u geacht zelf in te brengen. Dit in verband met het copyright van Tasman en Filosoft.

DE INSTALLATIE PROCEDURE.

Die is tamelijk eenvoudig. Op de schijf bij uw Nieuwsbrief vindt u een vijftal Taswordprogrammas en wel:

```
tw
twgr/pr
tw1
tw2
TWCHAR
```

Nu neemt u een lege schijf. Daarop zet u natuurlijk als eerste uw SYSteenfile. Nu neemt u "tw" over. Dat wordt indien gewenst uw "autoload" file. Wanneer u meerdere tekstbewerkers op een schijf zet houdt u natuurlijk de naam "tw" aan. Nu zet u het 10k grote codeblok van Tasword over. Zoals gezegd wordt u geacht dat reeds in bezit te hebben. Geef het als naam "TW1CODE"

Vervolgens leest u "tw2" in. Nu vervangt u de Nieuwsbriefschijf door uw eigen exemplaar en voert een RUN uit. U beantwoordt de vraag of u een of twee diskdrives gebruikt. Het resultaat is een codeblok "TW2CODE" op uw nieuwe schijf. Hierna kopieert u het programma "tw1" en tenslotte "twgr/pr". Dat is voorlopig alles. Wat u met het codeblok "TWCHAR" aanvangt leest u straks.

PROGRAMMAVOLGORDE

Wanneer u alles goed hebt gedaan ziet u na een CAT de volgende programmas in deze volgorde:

```
Sys
autoload (of tw)
TW1CODE
TW2CODE
tw1
twgr/pr
```

Die volgorde is echt belangrijk. De Disciple is razendsnel. Maar alleen met onze hulp. Een programma met een hoger p-nummer heeft een aantoonbaar langere inleestijd. Die snelheid vliegt helemaal met sprongen achteruit wanneer we een "gatenschijf" gaan vullen. Wanneer u 't niet gelooft neemt u de proef maar eens op de som. U pakt daarvoor uw werkschijf. Dat is toch meestal "gatenkaas". Stel de (liefst korte) programmas p4, p7, p10 en zo nog een paar ontbreken. Nu maakt u een kopietje van uw bovenstaande Tasword-schijf. Door de doordachte naamkeuze gaat dat in een wip met

```
SAVE d1*tw* TO d2
```


vervolg van "PROGRAMMAVOLGORDE"

U zult nu constateren dat het kopieprogramma tot wel drie maal langer nodig heeft om te laden. Het verhaal dat u nu leest stond als 42ste file op de werkschijf. Het inlezen kostte 14 seconden. Na bovenstaande handeling was het nog steeds de 42ste file. De inleestijd was terug op 4 seconden! De oorzaak is "bit mapping". Dat is op zich een heel mooi systeem. Zodra u een programma wist staan de vrijgekomen sectoren meteen weer tot uw beschikking. Uw schijf raakt zo nooit vol. Dit in tegenstelling tot BETA en OPUS DISCOVERY waarbij u om de zoveel tijd een MOVE moet uitvoeren. Maar u ziet nu ook het nadeel. De leeskop springt netjes van de ene sector naar de andere over de niet benodigde sectoren heen.

Ergens in de handleiding van de Disciple wordt terloops opgemerkt bij tijd en wijle uw hele disk te kopiëren. U weet nu waarom. Aan de CATALOGUS van die kopie ziet u niets. Hoogstens dat de p-nummering weer aansluit als er tenminste gaten waren gevallen. Dat al uw programmas nu ook netjes op aansluitende sectoren staan ziet u niet. U merkt het echter wel degelijk in het gebruik. De inleestijd is bekort. Het is wel te beredeneren. Na een

```
SAVE d1** TO d2
```

worden de programmas een voor een overgezet. Uw eerste programma wordt overeenkomstig de in de bit map opgeslagen informatie van de betreffende sectoren gehaald. Die sectoren kunnen dus over de hele schijf verspreid staan. Uw tweede schijf was leeg. De kopie wordt nu gemaakt op aansluitende lege sectoren. Bij het tweede en volgende programmas is het natuurlijk van 't zelfde laken een pak. Voer deze handeling uit met een "lege" machine. En denk aan de ramtop. Die moet echt op 65535 staan. Anders gebeuren er soms gekke dingen. De in de fabrieksspecificaties genoemde lees- en schrijftijden gelden duidelijk alleen bij geordende programmas. Lees nu nog eens wat er in uw handleiding staat geschreven bij:

```
FORMAT d1 TO 2
```

Waarschijnlijk gaat u nu een lichtje op. We hebben nog zo'n tip.

WISSEN

Na een CAT verschijnen de programmeergegevens meteen in beeld. Na het laatste programma treedt er een korte pauze op. Daarna leest u ook hoeveel vrije ruimte er nog op schijf staat. Voer nu het volgende programma uit:

```
10 FOR a= 1 TO 40  
20 SAVE d*"test"+STR$ a  
30 NEXT a
```

U wist alles in een ruk met: `ERASE d*"test"`

Doe nu weer een CAT. De programmas die nog op schijf staan ziet u als vanouds meteen verschijnen. De pauze voor de laatste regel is nu waarneembaar langer. Voer het bovenstaande programma nog eens uit. Nu doet u:

vervolg van "PROGRAMMAVL.GORDE"

```
10 FOR a=40 TO 1 STEP -1  
20 ERASE d*"test"+STR$ a  
30 NEXT a
```

Raar maar waar, maar de CAT is weer op z'n oude snelheid. Het is dus aan te bevelen wanneer u een schijf fatsoeneert dat zoveel mogelijk van achter naar voor te doen. Blijkbaar wordt er na het wissen van programma's de een of andere vlag niet altijd correct teruggesteld. We hebben dit fenomeen in Engeland aangekaart. Na al deze voor u hopelijk informatieve zijsprongen terug naar ons TASWORD programma. Ook daarin zitten nog een paar leuke programmeertechnieken verborgen. U kunt die zo in uw eigen programma's overnemen.

DE AANPASSINGEN NADER BEKEKEN.

U kunt nu met uw vernieuwde Tasword werken. Na het opstarten van Tasword belandt u op de bekende plaats in het programma. Op de eerste regel van het werkblad. Ga nu eerst met de STOPtoets naar het hoofdmenu. Dat ziet er uit als bijgaande schermafdruck. Ten eerste springt het kader in het oog. De file bevat nul woorden en heeft geen naam. Verder zien we dat optie "t" is verdwenen. Dat is voor ons, Disciple gebruikers, geen probleem. Met behulp van de wildcard maakt u immers zo een kopie. De zo vrijgekomen geheugenruimte kunnen we echt nuttiger besteden. Wat u met "f" en "u" kunt doen leggen we straks uit. De rest mag als bekend verondersteld worden. Terug dus naar het werkblad met de "y" (van RETURN voor mensen die een nieuwe Spectrum zonder keywords opdruk hebben en dus nooit enige logica in die letterkeuze van Tasword hebben kunnen ontdekken.) Typ nu een korte tekst in. Een regel of vier vijf is meer dan voldoende. Terug naar het menu. In het kader vindt u meteen het aantal getypte woorden vermeld.

SAVE

Kies "s" en typ na de CAT een naam voor uw document. Nu komt er wat nieuws. U kunt een deel van uw document opslaan. Dat is erg handig wanneer u aan uw 320 regels zit en nog extra regels wilt invoegen. Maar ook wanneer u een bepaalde alinea in een ander document moet gebruiken. Meestal zult u het document in z'n geheel willen opslaan. Het is dan tamelijk overbodig de eerste en laatste regel aan te geven. U sluit dat kort door de EDITtoets aan te slaan. Terug in het menu ziet u nu ook in het kader de zojuist door u gebruikte bestandsnaam staan.

LOAD

Kies "j" en gebruik nu de cursortoetsen om het gewenste document in te lezen. U hoeft hier dus geen naam meer in te typen. In deze catalogus is plaats voor zesenvijftig files. Ga nu nog even terug naar SAVE. Omdat uw document nu een naam heeft kunt u het document direct wegschrijven, zonder op de catalogus te wachten. Bij LOAD is er nog een weg terug. Wanneer u met een nieuw document verder wilt en vergeten bent het oude eerst weg te schrijven bijvoorbeeld. In 't oude programma helpt er geen lieve vader of moeder meer. Zodra u vanuit het menu LOAD bevestigde was uw oude document reddeloos verloren.

MERGE en ERASE

Ook hier tikt u de gewenste naam aan. Merk op dat de balk zowel "bovenlangs" als "onderdoor" kan. Een of twee keer naar boven is een stuk sneller dan twaalf keer naar beneden. De praktijk leert dat u meestal een groep documenten moet wissen. Daarom wordt na elke gewiste file een nieuwe CATalogus gegeven. Met "y" gaat u terug naar het menu. Dit werkt sneller dan steeds met de aanslagen "e" en ENTER vanuit het menu terug te gaan naar de wisoptie.

PRINT

Wederom zijn er een aantal extra mogelijkheden. U vindt ze in de schermafdruck. De begin- en eindregel kennen we. De linker kantlijn is ook hier in te stellen. De ervaring leert dat je afhankelijk van de tekst soms brede, dan weer smalle kantlijnen nodig hebt. Het is gewoon veel te veel gedoe dat telkens apart via de keuze "g" uit het hoofdmenu te doen. Je moet dan immers ook die hele waslijst grafische tekens enzovoorts doorploegen. De regelafstand staat op (1). Wanneer u die wijzigde zult u bemerkt hebben dat die steeds weer op (1) terugsprong. Nu niet meer. Eens veranderd blijft veranderd. Totdat u hem zelf weer terugstelt. Het aantal afdrucken is nieuw. Wanneer u bij een volgende printronde alles bij het oude wilt laten sluit u de hele lijst (of ook een gedeelte) kort met de EDITtoets.

DEFINE GRAPHICS/PRINTER

Keuze "g" is meestal een eenmalige aangelegenheid. Omdat keuze "t" is vervallen volgt er nu een vraag deze instellingen op uw schijf vast te leggen. Iets vreemds. Per grafisch blokje kunnen we tot 4 besturingscodes opgeven. Toch geeft de Disciple er maar twee door. Vanuit Basic werkt het echter wel goed. Uiteraard met inachtneming van de POKE 26,1/0 ga je vanuit Basic in de proportionele modus met zowel:

LPRINT CHR\$ 27;"pl" als LPRINT CHR\$ 27;CHR\$ 112;CHR\$ 49

Met 27,112,49 zoals gedefinieerd op foets 8 werlt het NIET.

PROGRAMMATECHNISCHE ASPECTEN.

Na bovenstaande korte samenvatting van de faciliteiten moet u nu onbekommerd met uw vernieuwde Tasword kunnen werken. Er zit wel een klein addertje onder het gras. Begin met het inlezen van een bestaand document van schijf. Wis dat met de X-toets op de gebruikelijke manier. U hebt nu een schoon werkblad voor u. Sla een enkele letter aan. Dat is al voldoende. Ga vervolgens naar het hoofdmenu. In het kader leest u nu de naam van uw vorig document. Terug naar uw werkblad, wis die ene letter, weer naar 't hoofdmenu. Nu is die oude naam wel verdwenen. Even op letten.

Met keuze "b" belandt u in het programma. Bekijk regel 15 van de listing. De oude ramtop van Tasword was 31999. Die is nu iets verlaagd. Een documentnaam n\$ wordt teruggelezen vanaf de nu beschermde geheugenplaatsen 31928 tot 31937. Het drivenummer d vinden we terug op 31940. In uw eigen programma's kunt u dit ook toepassen. Reserveer een handvol bytes boven ramtop en stel daar uw belangrijke variabelen veilig. Zo maakt u nu ook probleemloos

vervolg van "PROGRAMMATISCHE ASPECTEN"

gebruik van Deelprogrammas. Dit werkt over het algemeen sneller en eenvoudiger dan het telkens openen en sluiten van een serie-file naar schijf. Hoewel dat de meer gangbare werkwijze is bij het gebruik van Deelprogrammas. Boven de verlaagde ramtop staat ook de routine voor de woordtelling. Die vonden we in z'n oorspronkelijke vorm terug in een Sinclair User van jaren geleden.

In regel 30 wordt het kader uit het hoofdmenu getekend. Wanneer u de PLOT co-ordinaten aanpast is er rechts naast de "p", "s" en "j" keuzes ruimte voor nog zo'n kader. Daarin zou u bijvoorbeeld kunnen aangeven welke tekensets uit TASPRTN aanwezig zijn.

Voor de volledigheid. In regel 200 vindt u de "bergplaats" voor zowel het aantal afdrukken als de regelafstand. In ondermeer regel 300 ziet u hoe de beschreven "kortsluiting" met de EDITtoets (CHR\$ 7) wordt onderkend. Fanatieke programmeurs storen zich be-slist aan de slordige GO TO vanuit de subroutine. Het is geen doodzonde (na het afdrukken wordt een RUN uitgevoerd zodat ook de GO SUB/RETURN stack weer schoon is) maar 't blijft slordig.

DISKDRIVEVARIABLE

In het handboek van de Disciple staat dat je voor het drivenum-mer een variabele mag gebruiken. In de regels 550, 555, 680 en 690 vinden we hiervan voorbeelden. We hebben nog een tip voor de bezitters van twee diskdrives. Je kunt die variabele ook als vlag gebruiken en zo in een keer twee back-ups maken:

```
100 LET d=1
110 ERASE dd,n$: SAVE dd,n$: REM denk aan een rits DATA arrays.
120 IF d=1 THEN LET d=2: GO TO 110
```

In regel 595 zien we hoe de naam van uw document wordt bewaard. In regel 600 e.v. vindt u de cursorbesturing voor de CATALOGUS terug. Het is onmogelijk zomaar over het hele scherm te wandelen. De cursor blijft netjes boven de regels met namen. 't Maakt niet uit of dat veel of weinig namen zijn. Uw bewegingsvrijheid wordt door het programma precies afgebakend.

ERRORTRAPPING

In het handboek wordt daar vluchtig op ingegaan. In regel 775 vindt u een eerste voorbeeld. De CATALOGUS neemt het hele scherm in beslag. Dus ook de beide onderste regels. Er is dan ook geen plaats meer voor foutmeldingen. Wanneer u bent vergeten een schijf in de drive te stoppen zou de resulterende foutboodschap een fraaie crash veroorzaken. WEG document. Omdat er geen specifieke foutboodschap is gedefinieerd blijft de zaak nu draaien en kunt u alsnog uw schijf plaatsen. Over het precieze hoe en wat van die foutcodes leest u nog wel in uw Nieuwsbrief.

DE RESTJES

Het programma is niet voorzien van Headers, Footers en pagina-nummering. Omdat Tasword II toch geen "page break" kent (een dun stippellijntje om de zoveel regels) werkt dat niet fijn. Reken maar dat een nieuwe alinea vaker niet dan wel net op de laatste regel van de bladzijde blijkt te beginnen. Dat is werkelijk geen gezicht. Niets weerhoudt u natuurlijk dat zelf te programmeren. Vanuit het hoofdmenu hebt u nog twee keuzemogelijkheden. En wel de "f" en de "u". De ene springt naar regel 1000 en de andere naar regel 1100 en direct weer terug ook. We geven een voorzetje:

Op uw Nieuwsbriefschijf staat nog een programmatje "TWCHAR". Dat is een beter leesbare tekenset met aanpassingen voor de BROTHER CE daisywheelpriesters. U vindt er het guldensteken, de lange ij en nog een paar tekens. Niets werkt immers fijner dan wanneer u op uw beeldscherm ook meteen de juiste lettertekens ziet.

Verlaag de ramtop met een byte naar 31926. Bewaar het nummer van uw tekenset op 31927.

```
1000 LET i=31927
1000 IF PEEK i=0 THEN LOAD d1;"TWCHAR1"CODE: POKE i,1: RUN
1010 LOAD d1;"TWCHAR0"CODE: POKE i,0: RUN
```

Zo wisselt u met keuze "f" op eenvoudige doch doeltreffende wijze tussen twee tekensets. Met een tussenmenu kunt u extra tekensets toevoegen. Uit het voorgaande volgt ook hoe u bij het maken van uw eigen Deelprogrammas en Overlays met een eenvoudige PEEK opdracht eerst zou kunnen kijken of het door u gewenste programmagedeelte niet reeds in het geheugen zit. Dat bespaart bij aanwezigheid een dan overbodige LOAD instructie.

Na keuze "u" zou u bijvoorbeeld uw naam en adres gegevens kunnen laten afdrukken. Die paar LPRINT opdrachten hebt u zo in het programma aangebracht. Enz. enz. Aan u het initiatief.

Jaap Kuiper

Redactionele noot:

Het zal u als lezer waarschijnlijk ook niet ontgaan zijn dat dit artikel nogal aan de pittig lange kant is, om de woorden van de steller zelf te gebruiken, "de lengte is gierend uit de klauw gelopen". Na de nodige aarzeling hebben we toch maar besloten om het in een keer te plaatsen, mede omdat er genoeg interessante neven informatie in verpakt zit.

Bovendien krijgt u er nu de TW II software bij geleverd (op het codeblok na natuurlijk), en om daar optimaal gebruik van te maken heeft u de informatie van het hele artikel nodig. Bovendien is een goede tekstverwerker goud waard! Vandaar dus.

Het plaatsen van een dergelijk lang artikel is echter wel een eenmalige zaak. Houdt uw artikelen zo kort mogelijk of houdt er met schrijven rekening mee dat het in delen verwerkt moet kunnen worden.

(Red.)

UITGEREIDE A4-CAT

DEZE A4-CAT verzorgt, zoals de naam al zegt, een printeruitdraai van de directory op A4-formaat. Nu is dat natuurlijk al mogelijk dmv het commando CAT#3;1, waarmee de originele directory naar de printer gestuurd wordt. Dit programma verzorgt een uitdraai met uitgebreidere informatie. De opbouw van de gegevens is als volgt:

1. (No) programma nummer
2. (FILENAAM) programma naam, de aanduiding <ERASED> wordt 'toegevoegd' als het programma niet meer 'officieel' op schijf staat. (Herstel mogelijk met UN-ERASE uit NB-1).
3. (Tr) De track waarop het eerste blok van het programma staat.
4. (Se) De sector waarop het eerste blok van het programma staat.
5. (As) Het totaal aantal sectoren waaruit een programma bestaat.
6. (TYPE) Het soort file.
7. (PARAMETERS) De parameters voor codeblokken en arrays, zoals startadres en lengte.
Bij BASIC wordt de totale programmalengte aangegeven met tussen haakjes de lengte van de basic + variabelen.

8. () Indien aanwezig de BASIC startregel.

Voor OPENType files staat de lengte direkt achter het type. De rest mag genegeert worden.

Het programma is volledig in basic en kan dus naar believen worden aangepast. De opbouw valt in twee hoofdblokken uiteen. De in regel 10 gedeclareerde variabelen worden door regels 100 t/m 299 gevuld met de informatie uit de directory. Regels 300 t/m 700 doen niets anders dan die variabelen zichtbaar maken op scherm en printer. Eventuele layout veranderingen moet u daar kwijt. Veel plezier ermee.

P.Faas

voorbeeld:

Disciple Nieuwsbrief A4 PRINTER-CAT >>>>>> (c) Zaansoft 1987.

No	FILENAAM	Tr	Se	As	TYPE	PARAMETERS	
1	TRANSFORM	4	1	24	BAS	11722(11716+6)	9997
2	lader2	12	2	3	BAS	1363 (441+992)	
3	data	5	6	1	D.AR	27326,403	
4	ltcode	5	10	19	CODE	55781,9500	
5	TAS_setup	7	9	6	BAS	2864 (2800+64)	
6	char.array	5	7	2	\$.AR	26518,805	
7	m/c	10	1	12	CODE	58624,6100	
8	openfile	12	5	1	BAS	272 (234+38)	
9	TRANS	13	7	51	CODE	40001,25534	
10	testfile	12	6	5	OPEN	2550 (0+2550)	0
11	copy-oft	13	1	1	BAS	346 (346+0)	
12	testfile <ERASED>	11	9	1	OPEN	0 (0+0)	0
13	totaalsys	13	2	33	CODE	0,16384	

```

10 DIM e$(80,8): DIM n$(80,10): DIM a(80): DIM b(80): DIM c(80)
   ): DIM d(80): DIM e(80): DIM f(80): DIM g(80): DIM h(80)
15 REM a=aantal sect. b=track c=sector d=tupe e=startadres f=1
   engte g=length basic h=startregel
20 LET o$="<ERASED>": LET x=1
100 REM laad dir in n$
110 FOR t=0 TO 3
120 FOR s=1 TO 10
125 LET adres=40000
130 LOAD @*,t,s,adres
135 FOR q=1 TO 2
140 IF PEEK (adres+1)=0 THEN GO TO 300
145 LET m$=""
150 FOR w=1 TO 10: LET m$=m$+CHR$ PEEK (adres+w): NEXT w
155 LET n$(x)=m$
160 IF PEEK adres=0 AND PEEK (adres+1)=0 THEN LET e$(x)=o$
170 LET a(x)=256*PEEK (adres+11)+PEEK (adres+12)
180 LET b(x)=PEEK (adres+13)
190 LET c(x)=PEEK (adres+14)
200 LET d(x)=PEEK (adres+211)
210 LET e(x)=PEEK (adres+212)+256*PEEK (adres+213)
220 LET f(x)=PEEK (adres+214)+256*PEEK (adres+215)
230 LET g(x)=PEEK (adres+216)+256*PEEK (adres+217)
235 LET h(x)=PEEK (adres+218)+256*PEEK (adres+219)
240 LET x=x+1: LET adres=adres+256
250 NEXT q
260 NEXT s
270 NEXT t
299 REM inlezen klaar
300 REM filetype
301 LPRINT "Disciple Nieuwsbrief   A4 PRINTER-CAT
   Zaansoft 1987."
302 LPRINT
305 PRINT "Filenaam";TAB 20;"T-";TAB 23;"S-";TAB 26;"A"
306 LPRINT "No";TAB 3;"FILENAAM";TAB 23;"Tr";TAB 27;"Se";TAB 31
   ";TAB 35;"TYPE";TAB 41;"PARAMETERS": LPRINT
310 FOR q=1 TO 80
320 IF n$(q)="          " THEN STOP
330 LET t=d(q)
332 IF a(q)=97 OR a(q)=258 THEN LET t$="CODE": LET l=9
340 IF t=0 THEN LET t$="BAS ": LET s$=STR$ h(q): IF h(q) < 10000
   THEN LET s$=""
350 IF t=1 THEN LET t$="D.AR"
360 IF t=2 THEN LET t$="$.AR"
370 IF t=3 THEN LET t$="CODE"
380 IF f(q)=0 AND g(q)=0 THEN LET t$="OPEN"
500 REM print n$ op scherm
700 PRINT n$(q);TAB 11:e$(q);TAB 20;a(q);TAB 23;b(q);TAB 26;c(q)
   );TAB 29;d(q)
750 IF t<>0 THEN LPRINT q;TAB 3;n$(q);TAB 14:e$(q);TAB 23;b(q);
   TAB 27;c(q);TAB 31;a(q);TAB 35;t$;TAB 41:f(q);".":e(q)
800 IF t=0 THEN LPRINT q;TAB 3;n$(q);TAB 14:e$(q);TAB 23;b(q);T
   AB 27;c(q);TAB 31;a(q);TAB 35;t$;TAB 41:e(q);TAB 46;"(";q(q)
   );"+":e(q)-g(q);")";TAB 59;s$
900 NEXT q
9998 STOP
9999 SAVE d*"A4-CAT"

```

(Listing d.m.v. 'basic-tas')

ADRESSENREISTAND.

De heer A. Hol stuurde ons een adressenbestand dat gebruik maakt van de combinatie OPENTYP file en pseudo RAMdisk dmv de LOAD @ en SAVE @ commando's. Daardoor is een bestand van 180K of meer mogelijk. Het is in basic geschreven en daarmee is meteen de beperking aangegeven. Wij hopen echter dat dit programma voor anderen een aanzet is om ook een leuke toepassings mogelijkheid voor de Spectrum/Disciple combinatie te bedenken. U vindt het programma op de cassette/schijf daar de listing iets te veel van het goede was. Veel plezier met de bestudering van dit programma en bedankt A. Hol van de sqg Apeldoorn.

(red.)

RECORD versie 1.3

Dit programma werkt vanaf GDOS 3 met iedere Double Density drive. (Zowel 40/80 SS of DS).

Het principe bestaat er uit dat een volledig leeg file op schijf wordt gezet d.m.v. OPEN#, PRINT# en CLOSE#. Daarmee worden de sectoren voor ons bestand op schijf gereserveerd. Op iedere sector moet een geheel aantal records passen. Iedere willekeurig sector van het file is 512 bytes lang. Van deze 512 bytes worden steeds de laatste 2 bytes gebruikt om aan te geven waar de volgende sector van het file zich bevindt. Hieruit kunnen we het volgende concluderen :

- * Een file loopt van de laagste track/sector naar de hoogste track/sector.
- * Per sector kunnen we 510 bytes voor records gebruiken.
- * Het aantal records per sector moet totaal 510 bytes lang zijn. (vb recordlength= 85 karakters, records per sector 510/85=6)
- * Per record lengte zal er 1 karakter afgetrokken moeten worden voor de code 13 (ENTER).

In een 'record-map' wordt bijgehouden op welke sector we een record hebben weggeschreven. In dit file staat ook verkorte informatie over de records die op schijf staan. In dit geval de recordnummers 1 t/m 2160, het eerste karakter van de achternaam, voornaam, straat en plaats. Op deze karakters zoekt en sorteert het programma de records.

PROGRAMMA WERKING

OPSTARTEN

Dit kan d.m.v. RUN of GO TO 1000. Na een RUN gegeven te hebben komen we in een menu met de opties: 'Bestand opstarten' en 'Bestand laden'.

BESTAND OPSTARTEN

Als er nog geen bestand op schijf staat MOETEN we eerst deze optie kiezen, waarna het lege bestand wordt geopend. Is dat gebeurd dan komen we automatisch in de optie 'Bestand laden'.

PROGRAMMAWERKING

Opstarten

Dit kan d.m.v. RUN of GO TO 1000. Na een RUN gegeven te hebben komen we in een menu met de opties: 'Bestand opstarten' en 'Bestand laden'.

Bestand opstarten

als er nog geen bestand op schijf staat MOETEN we eerst deze optie kiezen, waarna het lege bestand wordt geopend. Is dat gebeurd dan komen we automatisch in de optie 'Bestand laden'.

Bestand laden

Dit is een soort RUN programma. Gebruik de toetsen 5,6,7 en 8 of de Kempston joystick om het cursorbalkje op het filenaam te zetten en laad het in via de spatiebalk/vuurknop. Dit laden duurt 3 tot 6 minuten en het programma keert daarna terug naar het hoofdmenu. (record-map wordt bijgewerkt.)

Hoofd menu

Dit bestaat uit: nieuw bestand, invoeren, wijzigen, verwijderen, opzoeken, sorteren, wegschrijven, en 'naar basic'. De opties 'Nieuw bestand' en 'Invoeren' wijzen zich vanzelf. De opties 'Wijzigen', 'Verwijderen', 'Gedeeltelijk' en 'Sorteren' werken met gebruikmaking van de toetsen 5,6,7,8 of joystick om de cursor te besturen en een veld te kiezen. De optie 'Wegschrijven' maakt een kopie van RECORD. (Dus niet van het bestand. Dat wordt tussentijds steeds bijgewerkt.)

Programma aanpassingen.

Het programma is nu ingericht voor een adressen bestand van 2160 records verdeelt over 360 sectoren van de schijf. Als je dat nog te weinig vindt kan je op regel 20 het aantal sectoren verhogen tot max. 540 sectoren (270 k bestand). Ook de elementen omschrijving op de regels 40 t/m 70 kan je naar believen wijzigen, om b.v. een videotheek- of een bibliotheekbestand aan te leggen. Wil je kortere records, een bestand over meerdere sectoren of de lengtes van de elementen wijzigen dan zal je het programma zoveel moeten wijzigen dat een nieuw programma schrijven waarschijnlijk sneller gaat.

A.J.Hol. S.G.G.Apeldoorn

* VARIA * VARIA * VARIA * VARIA * VARIA * VARIA * VARIA * VARIA *

GER VISCHER uit Amersfoort bied zijn DISCOVERY-ONE te koop aan, hij is aangepast voor de 128K. Prijs Fl.350.- tel. [REDACTED]
Ook veel 3.5" diskettes met software bied hij aan voor Fl. 2.50 per stuk...!Zelfde telefoonnr. Voor zijn interface-2 vraagt hij Fl.25.- Liefhebbers grijp je kans.

Werken met OPENTYPE files.

Door het werken met OPENTYP files in een database kunnen we grotere bestanden verwerken dan we aan RAM ter beschikking hebben. Een groot verenigingsbestand kan bv opgesplitst worden in een junioren-, dames- en senioren file. Ieder file kan beurtelings in het hoofdprogramma geladen worden.

Het principe bij het werken met OPENTYP files berust op het openen en sluiten van STREAMS, waarlangs de informatie van en naar schijf gestuurd wordt. Een 'stream' kan gezien worden als een eenrichting transportlijn tussen de computer en de diskdrive en kan geopend worden als een OUT stream (schrijffile UIT de computer naar drive), of IN stream (leesfile van schijf IN de computer.). Streams 0 t/m 3 zijn in gebruik bij de Spectrum voor o.a. printer, beeldscherm en toetsenbord. De overige streams 4 t/m 15 mogen we naar believen gebruiken. We mogen meerdere streams tegelijk openen.

Alle 256 codes & karakters kunnen over deze streams getransporteerd worden. Dit kan door gebruik te maken van een variabele. Deze variabele 'laden' we met het karakter dat we versturen willen en via de PRINT# opdracht versturen we de INHOUD van die variabele naar schijf. Zo kunnen we de WAARDES van numerieke variabelen of de ASCII codes van string variabelen naar schijf wegschrijven.

Voorbeeld :

```
5 LET a$="Dit wil ik naar schijf schrijven"
10 OPEN #4;dl;"test" OUT : opent 'lijn 4' naar een
   schrijffile genaamd "test". (Niet tram 4,red.)
20 PRINT #4;a$ : schrijf de INHOUD van
   a$ van regel 5 via stream 4 naar "test"
30 CLOSE #4 : sluit de stream af,
   leeg de diskbuffer op de schijf en update de
   directory.
```

De tekst "Dit wil ik naar schijf schrijven" is nu daadwerkelijk via stream 4 naar schijf geschreven. U had natuurlijk vrijelijk ieder streamnummer tussen 4 t/m 15 mogen kiezen. Om te controleren of het ook echt op schijf staat typt u :

```
MOVE dl;"test" TO #2 of MOVE dl;"test" TO #3
```

Het file wordt nu getoond op scherm of naar de printer gezonden. Het file is ook weer terug lezen in een variabele dmv het INPUT# commando. Dit hoeft NIET perse dezelfde variabele a\$ te zijn!

Voorbeeld :

```
10 OPEN #5;dl;"test" IN : via stream 5 willen we
   het file "test" INlezen.
20 INPUT #5;b$ : lees het file in b$
30 CLOSE #5 : sluit de boel netjes af
```

Type nu PRINT b\$ en u ziet de tekst weer terug op het scherm.

vervolg van "OPENTYPE files"

Het op deze manier verzenden van codes van en naar schijf dmv PRINT # en INPUT # kent enkele addertjes onder het gras! * De stream reageert op de codes 13, voor ENTER en 35, voor #.
* Mogelijk zijn er ook andere controlecodes die de stream kunnen verstoren, bv printer controle codes die in een tekst verwerkt zijn.
* Numerieke waarden kunnen niet in een stringvariabele worden gelezen en v.v.

Stel u wilt een willekeurig file inlezen in een stringvariabele, ongeacht wat voor codes in het file voorkomen. Het INPUT # commando voldoet nu niet meer ivm de genoemde beperkingen. Het commando INKEY# # brengt nu uitkomst. Inplaats dat het gehele file ineens in EEN variabele gelezen wordt, wordt door INKEY# # het file code voor code gelezen en in een gedimensioneerde string geplaatst.

Voorbeeld :U wilt het file "Listing" lezen in de variabele a\$ en later weer wegschrijven naar een andere schijf. Na een CAT ziet u dat het file 8 sectoren groot is.

```
10 DIM a$(4096) : B*512 bytes
20 OPEN #4:dl;"Listing" IN : inlezen
30 FOR f=1 TO 4096
40 LET a$(f)= INKEY# #4
50 NEXT f
```

Omdat het file niet exact 4096 bytes groot hoeft te zijn krijgt u waarschijnlijk de foutmelding End of file, 40:1. Gewoon niets van aan trekken. We weten nu de lengte van het file, nl. f-1. Stop een andere schijf in de drive.

```
Nu maken we de kopie : 60 CLOSE #*4 : leesfile moet nog dicht.
70 OPEN #4:dl;"Listing" OUT
80 FOR h=1 TO (f-1)
90 PRINT #4;a$(h);
100 NEXT h
110 CLOSE #*4
```

De ; achter a\$(h) in regel 90 is essentieel omdat anders na iedere code van a\$(h) de code 13 geplaatst wordt! Uw file is dan ineens dubbel zo groot en behoorlijk verminkt.

Het programma om OPENTYPE files te kopiëren in dit nummer is op het gebruik van INKEY# # gebaseerd. Experimenteert u zelf ook eens met commando's als :

```
INPUT #s;LINE a$
PRINT #s;CHR$ (0-255)
```

```
LIST #s
CAT #s:1
```

Artikel vrij bewerkt naar het origineel van A.Hol van de s.g.g. Apeldoorn.

NAAM - CAT

Dit programma is afgeleid van 'NAME-DISC', geschreven door Tycho Pandelaar en afgedrukt in Nieuwsbrief no. 3.

Bij dit programma heb je het oorspronkelijke initialisatie programma niet meer nodig.

De bedoeling van "NAAM-CAT" is om een schijf van een naam te voorzien bij het maken van een CAT naar 'n A4 printer, zodat je weet welke uitdraai bij welke schijf hoort.

Er wordt gebruik gemaakt van het commando CAT #3:* waardoor dit programma niet geschikt is voor printers die van de Sinclair RS232 poort gebruik maken.

N# in regel 10 mag naar believen veranderd worden, zolang de lengte maar 18 karakters bedraagt.

```
5 REM voor rom 3.0-3a-3b
10 LET n$="---Schijf in drive"
20 FOR n=5860 TO 5877
30 POKE @n, CODE (n$(n-5859))
40 NEXT n
50 INPUT "Schijfnaam : "; n$
60 IF LEN n$ < 1 OR LEN n$ > 18 THEN GO TO 50
70 LET n=10-LEN n$: FOR m=1 TO n: LET n$=n$+" ": NEXT m
80 POKE @5882, CODE "=": POKE @5883, CODE "*"
90 FOR n=5884 TO 5893
100 POKE @n, CODE (n$(n-5883))
110 NEXT n
120 CAT #3:*
130 GO TO 50
9999 SAVE d1"NAAM-CAT"
```



Een gewoonte die opgang maakt is om aan het begin van iedere schijf een system-file te plaatsen.

Dit programma is ook erg makkelijk om deze system-files van een kenmerk te voorzien.

Bv : Sys-file in drive 1 = 3a-FT80

Na een CAT* zie je in een oogopslag welk sys-file in de Disciple staat en voor welke printer dat file geïnitieerd is.

Verander regel 10 , 50 , 120 en 130 in

```
10 LET n$="Sys-file in drive "
50 INPUT "Systemfile = "; n$
120 SAVE d*"System 3"CODE 0,6654 : REM overschrijf oude file
130 STOP
```

P. Faas

BITMAP 3

In de Nieuwsbrief nummer 1 heb ik een stukje geschreven over de bitmap. Het daarbij afgedrukte programma "bit-map2" geeft een getalmatig overzicht van de bitmap en de daarin voorkomende programma sectoren. Nu leek het mij een leuk idee om eens een schijf echt in kaart te brengen. Niet met de gegevens uit de directory maar, voor de verandering, eens via een heel andere benadering. Ik wil u daarbij graag stap voor stap meenemen naar het eindresultaat, het programma 'bitmap3'. (Hoe kom ik toch steeds weer op die originele namen?)

U moet weten dat ik op een goeie dag tegen iets in de Disciple-ram aanliep wat verdacht veel op een bitmap leek. Op zoek naar de oplossing van de bug : Not enough SPACE on disk (blz.6/NB 3) vond ik deze bitmap op de adressen 14848 tot 15043. Als u nu een schijf in de drive stopt, en u voert een SAVE of OPEN # commando uit, dan wordt de bitmap van die schijf naar die adressen toe geschreven. Alle gegevens van de schijf worden in de 195 bytes van die bitmap gezet.

Als er een sector beschreven is dan wordt een bit op 1 gezet en als de sector leeg is dan staat het corresponderende bit op 0. Even snel rekenen, $195 \text{ bytes} \times 8 = 1560 \text{ sectoren}$. $1560 / 10 = 156 \text{ tracks}$, ofwel alle maximaal mogelijke minus de 4 tracks van de directory. (160 stuks maximaal.) Een bijkomstigheid is nu dat deze bitmap pas wordt bijgewerkt als u een van beide commando's hebt uitgevoerd. Heeft u nu een schijf waar al 80 programma's opstaan, dan is het SAVE commando niet meer mogelijk. Vandaar dat ik gekozen heb voor de volgende oplossing.

```
10 OPEN #4;d*;"#%&'" OUT
20 CLEAR #
```

De naam in regel 10 doet er niet toe en regel 20 sluit stream 4 weer ZONDER dat de naam in de directory wordt bijgeschreven, waarmee de foutmelding Disc directory full wordt vermeden. De eerst vrije sector op de schijf wordt op 1 gezet. De schijf die u in kaart wilt brengen moet dus minimaal 1 vrije sector bezitten.

Zo, we hebben hiermee de bitmap gevuld met die van onze schijf, maar nu moeten hem nog naar onze Spectrum ram halen. En dat het liefst zonder LOAD en SAVE commando's. Wat ons dan rest is een kleine machinecode routine die deze bitmap voor ons uit de Disciple-ram naar de Spectrum-ram kopieert. Deze machinecode is ons al bijna panklaar aangereikt in het programma "Mover". (Onderdeel van uw initialisatie prog.)

Mover bestaat uit twee routines, n.l. op 49000 een routine om het 'Sys 3' file van adres 42240 naar de Disciple-ram te schrijven en op adres 49030 de routine om het 'Sys 3' file terug te kopiëren naar de Spectrum.
Die routine hebben we nodig en gaan we nader bekijken.
De adressen heb ik voor ons doel al aangepast.

Opcode-operand	HEX-code	Decimaal
DI	F3	243
IN A, (187)	DB BB	219,187
IN A, (123)	DB 7B	219,123



vervolg van "BITMAP 3"

LD HL,14848	21 00 3A	33,0,58
LD DE,44848	11 30 AF	17,48,175
LD BC,195	01 c3 00	1,195,0
LDIR	E0 80	237,176
OUT (123),A	D3 7B	211,123
OUT (187),A	D3 8B	211,187
EI	FB	251
RET	C9	201

Mocht u dit voor het eerst zien, de eerste kolom is de zgn assembler listing van de machinecode, ofwel de leesbare tekst. Het zelfde staat er naast in hexadecimaal- en decimale notatie. Als de getallen uit de rechter kolom in de RAM gepokeet worden vanaf bv adres 40000, dan voert de computer na een RANDOMIZE USR 40000 deze routine uit. Even een korte uitleg.

DI (disable interrupts) koppelt alle interrupts los van de 280, zodat die bezig blijft met onze routine. De IN A instructies zorgen voor het inschakelen van de Disciple ram. LD HL,14848 laad het register HL met het startadres van onze bitmap en LD DE,44848 laad register DE met de plek in RAM waar ik de bitmap heen wil hebben. In het tel register DE wordt nu de lengte van 195 bytes gezet, waarna de instructie LDIR het kopiëren uitvoert. Als alle 195 bytes verplaatst zijn naar het Spectrum-ram dan moet de Disciple-ram uit- en de Spectrum ROM weer ingeschakeld worden. Hiertoe dienen de twee OUT instructies. EI (enable interrupts) koppelt de interrupts weer aan-en via RET keren we terug naar ons BASIC programma. U ziet het, niet zo heel moeilijk.

Deze routine wil ik wegpoken op adres 23296 en u ziet de getallen uit de derde kolom terug in het programma.

```

40 FOR f=23296 TO 23317
50 READ a: POKE f,a
60 NEXT f
70 DATA 243,219,187,219,123,33,0,58,17,48,175,1,195,0,237,176,
211,123,211,187,251,201

```

nu nog even daadwerkelijk het verplaatsen uitvoeren,

```

80 RANDOMIZE USR 23296

```

De bitmap die we nu op adres 44848 in onze Spectrum hebben staan hoeven we alleen nog maar zichtbaar te maken op de door ons gewenste wijze. Alle 195 getallen worden 'pseudo' binair getoond. Dit omdat de getallen gespiegeld in de bitmap staan. Het getal 1 is binair 00000001 In de bitmap is dit echter 10000000, vandaar 'pseudo' binair. Het getal 0 of 255 bezorgt ons wat dat betreft geen problemen. Spiegel maar eens 00000000 en 11111111. Het duidelijkst vond ik een weergave per 4 tracks op een 80 koloms printer.

vervolg van "BITMAP 3"

Vier tracks x 10 sectoren geven 40 bits per regel om af te drukken. Elk getal in de bitmap bevat 8 bits. Om dus deze 4 tracks zichtbaar te maken moeten er dus 5 getallen per regel afgedrukt worden. Hiervoor zorgt de FOR/NEXT loop op regel 1020. Daarmee wordt n\$ met de vijf getallen gevuld, waarna de n\$ in regel 1210 op het scherm gezet wordt. (Of printer als u PRINT door LPRINT vervangt.)

```
990 LET track=4
1000 FOR f=44848 TO 45042 STEP 5
1010 LET n$=""
1020 FOR h=0 TO 4
1030 LET getal=PEEK (f+h)
1040 IF getal=255 THEN LET a$="11111111": GO TO 1120
1050 IF getal=0 THEN LET a$="00000000": GO TO 1120
1060 LET a$="00000000"
1070 LET a=7
1080 FOR g=1 TO 8
1090 IF getal>=2^a THEN LET getal=getal-(2^a): LET a$(a+1)="1"
1100 LET a=a-1
1110 NEXT g
1120 LET n$=n$+a$
1125 NEXT h
1130 LET track=track+4: IF track=84 THEN LET track=track+48
1200 REM ** adrukken **
1210 PRINT "Track ";track-4;" t/m ";track-1;TAB 22;n$(1 TO 10);
PRINT n$(11 TO 20);" ";n$(21 TO 30);" ";n$(31 TO 40)
1250 NEXT f
9998 STOP
9999 SAVE dl"bitmap3"
```

Regel 1130 zorgt er voor dat vanaf track 80 de tracknummering wordt aangehouden zoals die gebruikt wordt in de LOAD & SAVE @ commando's.

Ik hoop dat het stap voor stap doorlopen van een programma idee tot aan het uiteindelijke resultaat, enkelen van u tot programmeren zal aanzetten. De Nieuwsbrief is er voor om de resultaten van uw inspanningen een plaatsje te bieden. Ik ben benieuwd!

En hier is dan het resultaat.....!!
Een kaart van een van mijn, willekeurig gekozen, schijven.

Eerst een CAT#3:1 :

* DISCIPLE .. DISC 1 DIRECTORY *

1 system	14 CDE	0,6656
2 bitmap3	1 BAS	
6 Fist IIb	97 SNP 48K	
8 The Lords	97 SNP 48K	
9 Dirty Mov	97 SNP 48K	
13 Weird	97 SNP 48K	
14 Witchfight	97 SNP 48K	

Number of Free K-Bytes = 530



vervolg van "BITMAP 3"

Daarna een RUN van "bitmap3" : en.... is het geen schoonheid???

```
Track 4 t/m 7      1111111111-1111110000-0000000000-0000000000
Track 8 t/m 11     0000000000-0000000000-0000000000-0000000000
Track 12 t/m 15    0000000000-0000000000-0000000000-0000000000
Track 16 t/m 19    0000000000-0000000000-0000000000-0000000000
Track 20 t/m 23    0000000000-0000000000-0000000000-0111111111
Track 24 t/m 27    1111111111-1111111111-1111111111-1111111111
Track 28 t/m 31    1111111111-1111111111-1111111111-1111111111
Track 32 t/m 35    1111111100-0000000000-0000000000-0000000000
Track 36 t/m 39    0000000000-0000000000-0000000000-0000000000
Track 40 t/m 43    0000000000-0000000000-0000011111-1111111111
Track 44 t/m 47    1111111111-1111111111-1111111111-1111111111
Track 48 t/m 51    1111111111-1111111111-1111111111-1111111111
Track 52 t/m 55    1100000000-0000000000-0000000000-0000000000
Track 56 t/m 59    0000000000-0000000000-0000000000-0000000000
Track 60 t/m 63    0000000000-0000000000-0000000000-0000000000
Track 64 t/m 67    0000000000-0000000000-0000000000-0000000000
Track 68 t/m 71    0000000000-0000000000-0000000000-0000000000
Track 72 t/m 75    0000000000-0000000000-0000000000-0000000000
Track 76 t/m 79    0000000000-0000000000-0000000000-0000000000
Track 128 t/m 131  0000000000-0000000000-0000000000-0000000000
Track 132 t/m 135  0000000000-0000000000-0000000000-0000000000
Track 136 t/m 139  0000000000-0000000000-0000000000-0000000000
Track 140 t/m 143  0000000011-1111111111-1111111111-1111111111
Track 144 t/m 147  1111111111-1111111111-1111111111-1111111111
Track 148 t/m 151  1111111111-1111111111-1111000000-0000000000
Track 152 t/m 155  0000000000-0000000000-0000000000-0000000000
Track 156 t/m 159  0000000000-0000000000-0000000000-0000000000
Track 160 t/m 163  0111111111-1111111111-1111111111-1111111111
Track 164 t/m 167  1111111111-1111111111-1111111111-1111111111
Track 168 t/m 171  1111111111-1111111111-1111111111-1111111111
Track 172 t/m 175  1111111111-1111111111-1111111111-1111111111
Track 176 t/m 179  1111111111-1111111111-1111111111-1111100000
Track 180 t/m 183  0000000000-0000000000-0000000000-0000000000
Track 184 t/m 187  0000000000-0000000000-0000000000-0000000000
Track 188 t/m 191  0000000000-0000000000-0000000000-0000000000
Track 192 t/m 195  0000000000-0000000000-0000000000-0000000000
Track 196 t/m 199  0000000000-0000000000-0000000000-0000000000
Track 200 t/m 203  0000000000-0000000000-0000000000-0000000000
Track 204 t/m 207  0000000000-0000000000-0000000000-0000000000
```

Peter Faas

Nog even een varia.....

Op de valreep kwam er nog iemand langs die ook al zijn DISCOVERY wil omruilen voor een DISCIPLE, de prijs is Fl.350.- Van dezelfde eigenaar tevens een Multiface-1 (125.-) en een partij schijven met de nodige software (2.50 p.st.) reacties hierop via het info nummer van de redactie, ~~0445-1000~~.

BASICODE en de PRINTER

Behalve de in NB 3 besproken SAVE problemen met Jan Bredenbeeks BASICODE-2, versie 8.01, zijn er ook printer problemen. Ook nu zijn bepaalde routines in het programma daar schuldig aan. In dit geval is het de printer routine die roet in het eten gooit.

De heer Klop vond voor ons de oplossing bij het doorbladeren van wat oude nummers van de Sinclair Gebruiker.

In no. 10, 2e jaargang, staat een artikel over Basicode, waar deze problemen ook besproken worden. De oplossing is als gewoonlijk vrij simpel. (Als je het weet.) De printer-routine mag niet meer opgestart worden.

Verwijder hiervoor ieder RANDOMIZE USR 57993 statement uit het programma, u vindt deze statements in de regels 750, 870, 940 en 960. Verwijder uit deze regels ALLEEN het genoemde statement. Schrap dus niet de HELE regel. Heeft u de versie waar vanaf regel 1000 de handleiding in staat, dan moet u ook regel 1005 aanpassen!

Ik heb het uit geprobeerd en printen gaat nu prima, al is de print breedte beperkt tot 42 karakters per regel. Langzaam maar zeker krijgen we zo toch een redelijk bruikbare BASICODE versie.

*VOORUITBLIK * VOORUITBLIK * VOORUITBLIK * VOORUITBLIK * VOORUIT*

Nummer 4 was weer sneller vol dan ik verwacht had terwijl van alle kanten de informatie nu begint los te komen. Enkele voorbeelden van wat u in nr. 5 te wachten staat :

- * Andre Hermanns (Herrosoft, dus kwaliteit!) schreef voor ons en voor u een Discipleprogramma met allure.
- * Hackers opgelet! Hoe sloop je een 48K Snapshot uit elkaar? Lees het in nummer 5!
- * "Snap-BAS" list basic uit 48K Snapshots. Soms kan het handig zijn een blik te werpen op de basic van een programma, nu is dat ook bij snapshots mogelijk.
- * Een programma dat de mooiste TAPE back-up's maakt van 48K Snapshots, inclusief screen\$, volautomatisch en zonder een kraakinterface en met minimale inspanning! Alleen voor 128K en 128K+2 bezitters ivm het gebruik van de Ramdisk.
- * De pro en contra's van de DISCIPLE PLUS-D
- * Uw eigen bijdrage? Jatochzekernietwaar?

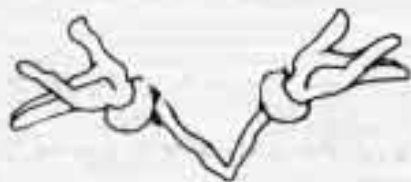
COMPLETE VERZAMELING FOUTBOODSCHAPPEN IN GDOS

Na enig zoekwerk hebben we ter lering en vermaak alle fout boodschappen voor u uit de DISCIPLE geplukt. U kunt de foutboodschappen vinden vanaf adres 15697 in de GDOS. Voor iedere foutmelding staat het selectie nummer en een korte verklaring vind u er achter.

0	: Nonsense in GDOS	de vuilnisbak
1	: Nonsense in GNOS	oepe! nonsense in foutmelding
2	: Statement END error	wel 'n volledig commando intypen
3	: BREAK requested	leun niet op de break toets
4	: ,SECTOR error	ai, programma weg
5	: FORMAT data lost	au, hele schijf weg
6	: NO DISC in drive	stop 'n schijfje in je driveje
7	: No "SYSTEM" file	gebeurt in de beste families
8	: Invalid FILENAME	de naam "kanmaartottien" mag niet
9	: Invalid STATION	computer vandalisme
10	: Invalid DEVICE	printen met een drive kan niet
11	: VARIABLE not found	waar issie gebleven?
12	: VERIFY failed	ik heb gefaald!
13	: Wrong FILE type	we willen allebei iets anders
14	: MERGE error	niet aan elkaar te knopen
15	: CODE error	uw basic is geen code, meneer!
16	: PUPIL set	akela wij doen ons best
17	: Invalid CODE	krukkige code
18	: Reading a WRITE file	lezen uit een OUT file kan niet
19	: Writing a READ file	schrijven naar een IN file? gek!
20	: O.K. GDOS 3	goed gedaan jochie!
21	: Network OFF	de boel staat uit
22	: Wrong DRIVE	eerst effe drive 2 initialiseren!
23	: Disc write PROTECTED	plakkertje eraf halen
24	: Not enough SPACE on disc	burrp, ik zit vol
25	: Directory FULL	ik ook
26	: File NOT FOUND	gezocht en niet gevonden!
27	: END of file	meer zit er niet in
28	: File NAME used	misbruik Zijn naam niet!
29	: Not a MASTER station	de bluffer leeft boven zijn stand
30	: STREAM used	kraan stond al open hoor
31	: CHANNEL used	dit kanaal is dus al in gebruik

Dit zijn ze dan alle 32. Meer smaken zijn er niet.

NB: Zie je wel Gerrit, nummer 6 staat er echt in.



DISCIPLE.....
THE BEST

