

DISCIPLE

3e Jaargang nr. 15

Met o.a.

Execute renumber routine

Auto-system loader

Sysvar-listen

Directory sorteerder

Supersnel klonen

De thermoprinter

Scart TV als monitor

Enz.

Enz.

Enz.

vragen?



bel de hulplijn!

NIEUWSBRIEF

DATA - SKIP

HET ADRES VOOR AL UW
WENSEN OP GEBIED VAN:

SPECTRUM
DISCIPLE EN PLUS-D
ATARI
PC

WAAR SERVICE NOG EEN
BEGRIIP IS.

BEL ONS VOOR PRIJZEN;
01820 - 20581

OF KOM EENS LANGS;
OOSTHAVEN 58
2801 PE GOUDA



OOK IN
NIEUWSBRIEF
—15—
ZIT WEER
ZAT MUZIEK!



INHOUD

COLOFON.....	15-1
INHOUD/INFO.....	15-2
SECTOR_SAVER EINDELIJK SUPER?.....	15-3
DIRECTORY - SORTERDER.....	15-7
128K "EXECUTE" RENUMBER ROUTINE.....	15-12
AUTOSYSTEM LOADER.....	15-14
DE SCART-TV ALS PERFECTE RGB-MONITOR.....	15-18
TEKENING EN ONDERDELENLIJST.....	15-17
SYSTEENVARIABLEN-LIJST.....	15-18
SUPERSNEL KLONEN, TUUT-TUUT.....	15-20
DE THERMO PRINTER AFGESCHREVEN???	15-22

HITS VAN DE MAAND...!!

* AUTOSYSTEM LOADER...

ZIEN IS GEBRUIKEN...

** ONTSTOF UW THERMOPRINTER...

*** VERGEET DE HCC DAGEN NIET...!!



SECTOR_SAVER EINDELIJK SUPER?

Enig enthousiasme mag mijn mede redacteur en "koppen" verzinner Carl niet ontzegt worden toen hij "SAVER" als super sectorredder aankondigde in nieuwsbrief 14.

Sterker nog, hier was sprake van schromelijke overdrijving. In de praktijk bleek dat "SAVER" maar weinig succes scoorde. Als u de code er op na kijkt, dan is snel duidelijk waarom. Het is waar dat de defecte sector geladen wordt EN dat die na terugschrijven de diskette repareert. Echter of de DATA van die sector het overleeft heeft wordt niet getest.

Gevolg is een zeer geringe kans van slagen en eenmaal verminkte DATA is nooit meer goed te krijgen.

Dat moest dus beter kunnen!

Logisch is de volgende stap, NA het laden even testen of de DATA goed gelezen is! Is dit niet het geval dan moet de mogelijkheid bestaan het nogmaals te proberen of de procedure af te breken. Pas wanneer de DATA korrekt is ingelezen moet de mogelijkheid bestaan die terug te schrijven naar schijf.

Het resultaat vindt u in onderstaand programma.

Riemen vast, hier is de echte super sector saver

```
10 CLS : GO SUB 9000
```

GO SUB 9000 levert deze machinecode op:

LD	A,1	;drivenummer in reg. A
LD	IX,40000	;laadadres voor de sector
LD	D,0	;tracknummer in reg. D
LD	E,1	;sectornummer in reg. E
RST	8	;hookcode met DEFB #44 is routine
DEFB	#44	;LDSIR (load sector into ram), de niet gedocumenteerde hookcode die door Tonnie Stap is gevonden en beschreven in nr.8 en nader uitgelegd in nieuwsbrief 11/blz.24.
JR	C,ERROR	;DATA is foutief geladen, spring naar ERROR
LD	BC,0	;DATA is goed geladen (hoera!), wis eventuele oude foutmeldingen uit reg. BC
RET		;en keer terug naar BASIC
ERROR LD	C,A	;Het reg. A bevat het foutnummer, zet dat nu in reg. C
LD	B,0	;reg. B moet 0 zijn en blijven
RET		;keer met het foutnummer in reg. BC terug naar BASIC.

Opvallend verschil met de routine uit nieuwsbrief 14 is het gebruik van hookcode #44 ipv hookcode #3F. Hookcode #3f plaats de geladen sector in de 512 bytes buffer IN de Disciple, terwijl hookcode #44 de mogelijkheid verschaft de sector direkt in de Spectrum RAM te plaatsen.

```
15 POKE 82001,2
```



Ik POKE hier een 2 omdat ik van drive 2 af werk.
 De routine initialiseert op drive 1.
 Schrap regel 15 als u een enkele drive bezit.

(IV en TV in PRINT statements betekent INVERSE en TRUE VIDEO)

Een zeer groot gedeelte van het programma bestaat uit PRINT statements, zodat u later ook zonder nieuwsbrief nog weet hoe het programma te gebruiken.

```

20 PRINT "IV DISCIPLE & +D SECTOR SAVER TV"
30 PRINT "" Stop 'n defekte schijf in de"
40 PRINT " drive en druk op een wille-"
50 PRINT " keurige toets."
60 PRINT "" Het programma probeert dan een"
90 PRINT " aantal malen de defekte sector"
100 PRINT " te lezen. Lukt dit zonder"
110 PRINT " leesfouten, dan wordt daarna"
120 PRINT " de sector terug geschreven op"
130 PRINT " de schijf. De sector is dan"
140 PRINT " tijdelijk gerepareerd."
150 PRINT "" Maak direkt een kopie van de"
160 PRINT " schijf met bv SAVE TO en"
170 PRINT " formatteer hem daarna opnieuw!"
180 PRINT ""IV (c) P.Faas 1989 TV"
190 PAUSE 0
200 CLS
205 PRINT "IV DISCIPLE & +D SECTOR SAVER TV"
210 PRINT ""Indien een sector-error optreedt"
220 PRINT "dan geven de DISCIPLE en PLUS-D"
230 PRINT "zelf aan waar de defekte sector"
240 PRINT "zich bevindt."
250 PRINT ""Bv: 12:4,Sector error"
260 PRINT ""De defekte sector staat dan op"
270 PRINT "track 12 en het is de 4e sector."
280 PRINT ""Geef nu op waar de defekte"
290 PRINT "sector op uw schijf staat."
295 PRINT ""IV P.Faas 1989 TV"

```



In de machinecode worden nu het track en sectornummer gepoked waar de defekte sector zich bevindt.

```

300 INPUT "track : ";t: POKE 62007,t
310 INPUT "sector : ";s: POKE 62008,s
320 CLS
330 PRINT "We beginnen met 10 pogingen de"
340 PRINT "sector foutloos te lezen."
350 FOR f=1 TO 10

```

Door de machinecode aan te roepen door LET a=USR adres ipv RANDOMIZE of PRINT USR adres, krijgt de basicvariabele A het foutnummer zoals we dat in het reg. BC hebben geplaatst. Dat is service van uncle Clive geweest. Is de DATA foutief geladen, dan heeft A een waarde groter dan 0. Is de DATA foutloos geladen, dan zal A nul zijn.

```

360 LET a=USR 62000
370 PRINT AT 10,12;" ";AT 10,3;"Poging : ";f

```

Bij foutnummer 5, FORMAT DATA LOST, kunnen we beter meteen maar stoppen.

```
380 IF a=5 THEN PRINT AT 14,0;"De hele TRACK is niet geformat-  
teerd!""Dit is waarschijnlijk 'n gedeel-""telijk geformatte  
erde schijf!": PAUSE 0: RUN
```

Bij foutief laden gaan we het gewoon nogmaals proberen, NEXT f.

```
390 IF a=4 THEN NEXT f  
400 IF f<=10 THEN GO TO 460
```

Na 10 mislukte pogingen komen we hier terecht. Jammer.

```
410 PRINT ""Het is nu NIET gelukt!"
```

Maar een herkansing is mogelijk, want de aanhouder wint!
Vooral bij ZEER belangrijk gegevens verlies op diskettes waar we
geen back-up van hebben gemaakt (Wet van Murphy) is dit de enige
oplossing.

```
420 PRINT ""Nogmaals een poging wagen? (J/N)"  
430 PAUSE 0  
440 IF INKEY$="n" OR INKEY$="N" THEN RUN  
450 GO TO 320
```

Heft de loftrumpetten, het is dan toch nog gelukt!

```
460 PRINT ""Sector is FOUTLOOS geladen!"
```

Nu de mogelijkheid de sector te redden door terug te schrijven.

```
470 PRINT ""SAVEN? (J/N)"  
480 PAUSE 0  
490 IF INKEY$="n" OR INKEY$="N" THEN RUN
```

Op de plaats van hookcode #44 poken we nu 89 (#45) en dat is de
hookcode voor de routine SSFR, save sector from ram.
Alle parameters in de routine staan goed ingesteld (sector- en
tracknummer, drivenummer en ramadres van de sector), zodat
alleen de leesroutine vervangen hoeft te worden door de
schrijfroutine en er verder geen wijzigingen hoeven plaats te
vinden. LET a=USR 62000 start de machinecoderoutine.

```
500 CLS : POKE 62011,89: LET a=USR 62000  
520 PRINT AT 8,0;
```

De foutmelding wordt nu gebruikt om te testen of de sector
korrekt is weggeschreven.

```
530 IF a=0 THEN PRINT "en FOUTLOOS weggeschreven ";t;";s: PRI  
NT #1;"Druk op een toets.": PAUSE 0: RUN  
540 PRINT "SCHRIJFFOUT opgetreden!"  
550 PRINT ""Nogmaals proberen? (J/N)"  
560 PAUSE 0  
570 IF INKEY$="n" OR INKEY$="N" THEN RUN  
580 GO TO 500
```

Hopelijk is alles gelukt tegen de tijd dat u hier terecht bent gekomen.
Lukt het namelijk niet op deze manier om sectoren te redden, dan weet ik het niet meer.

```
8999 STOP
9000 RESTORE : FOR f=82000 TO 82021: READ g: POKE f,g: NEXT f
9010 DATA 82,1,221,33,84,158,22,0,30,1,207,68,56,4,1,0,0,201,79,
6,0,201
9030 RETURN
9999 SAVE d*"SECTORSAVE"
```

Veel succes!

P.Faas



GOUD VAN OUD

THE BEST OF THE DISCIPLE NIEUWSBRIEF

MET O.A. UN-ERASE-(HERSTELTOOL)
BAS-TAS (BASIC-NAAR TASWORD)
SNELLE RAMTEST-(DIAGNOSETOOL)
SPECWORD 48/+ /128-(WORDPROCESSOR)
TASWORD II- SPECIAL (ZONDER CODEBLOK)
TRANS-D>T-(DISK NAAR TAPE KOPIER)
TRANS-T>D-(TAPE NAAR DISK KOPIER)
TRANS-OTYPE-(OPEN TYPEFILE KOPIER)
MULTIFACER-(MULTIFASE EMULATOR)
DISK-DOKTER-(REPARATIE TOOL)
SCREENMAKER-(VERVANG SNAPSHOT SCREENS)
SECTORPIEK-(MACHINECODE PEEKER & POKER)
AUTO JACK-(AUTO LOADER)
FIN A FILE-(UN-ERASE TOOL)
OKIE POKIE-(POKE DUMPER)
TRANSSNAP-(SNAPSHOT DISK>TAPE KOPIER)

TE BESTELLEN DOOR STORTING VAN FL 12,50 OP
GIRO [REDACTED] T.N.V. P. FAAS, RED.NWSBR, [REDACTED]

DIRECTORY - SORTEERDER

Van Alex Hol ontvingen we dit programma, waarmee het mogelijk is om de directory te sorteren op filenaam of filetype. Vooral met redelijk volle schijven levert dit een merkbare verbetering op bij het zoeken naar een bepaald file. Het programma is een.. verbeterde versie van een eerder in de Format gepubliceerde directory-sorteerder. De snelle sorteerroutine voor 48/128K uit ons adressenbestand is toegevoegd.

Het programma werkt recht toe, recht aan. De directory wordt geladen en de filenamen, voorafgegaan door het filetype byte worden in een string (n\$) geplaatst. Om een goede alfabetische sortering te verkrijgen wordt alles in hoofdletters omgezet, maar wees niet bang de originele gegevens zoals ze in de directory stonden worden niet aangetast.

Bij iedere naam en type in de string hoort een blokje van 256 bytes in de directory. Wanneer de string gesorteerd wordt de nieuwe directory aan de hand daarvan samengesteld.

De string dient dus als referentie en een verplaatsingsroutine verschuift telkens blokjes van 256 bytes op de juiste plaats binnen die nieuwe directory.

De DATA voor die routine staat in regel 30 en wordt in regel 20 op zijn plaats gepoked. Wat daarbij opvalt is dat die routine gepoked wordt in het UDG geheugengebied gepoked, de plaats waar u anders uw eigen grafische tekenjes kan maken. De keus hiervoor was aan de programmeur, maar ieder andere vrije plek in het geheugen had ook gemogen.

De sorteerroutine om de gedimensioneerde string (n\$) voor type en filenaam te sorteren mag ook door een paar regeltjes basic vervangen worden. Het programma werkt dan iets langzamer, maar nog even goed. Dit voor een ieder die later wel dit artikel leest, maar geen kans meer ziet de bijbehorende schijf met sorteerroutine in handen te krijgen.

Laten we de listing eens volgen.

Laad de sorteerroutine en maak de verplaatsingsroutine aan.

```
10 CLEAR 63542
20 IF PEEK 63800<>58 THEN LOAD D:"DIRSORT MC"CODE : RESTORE 30
: FOR k=0 TO 11: READ 1: POKE (USR "A"+k),1: NEXT k
30 DATA 33,0,0,17,0,0,1,0,1,237,176,201
```

De data levert de volgende routine op:

33	0	0	LD	HL,00000	;van adres (wordt later gepoked)
17	0	0	LD	DE,00000	;naar adres (wordt later gepoked)
1	0	1	LD	BC,00256	;aantal van 256
237	176		LDIR		;verplaats
201			RET		;klaar

```
40 CLEAR 39999:
```

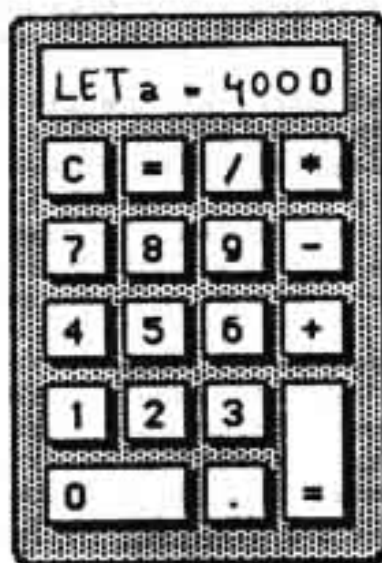


Hoofdprogramma:

```
50 GO TO 360
```

LAAD directory subroutine:

```
80 LET a=40000
70 FOR t=c TO b
80 LOAD @*,t,1,a
90 LOAD @*,t,3,a+1024
100 LOAD @*,t,5,a+2048
110 LOAD @*,t,7,a+3072
120 LOAD @*,t,9,a+4096
130 LOAD @*,t,2,a+512
140 LOAD @*,t,4,a+1536
150 LOAD @*,t,6,a+2560
160 LOAD @*,t,8,a+3584
170 LOAD @*,t,10,a+4608
180 LET a=a+5120
190 NEXT t
200 RETURN
```



Klopt Precies! Roken MAAR NA.

SAVE directory subroutine :

```
210 LET a=50240
220 FOR t=c TO b
230 SAVE @*,t,1,a
240 SAVE @*,t,3,a+1024
250 SAVE @*,t,5,a+1024*2
260 SAVE @*,t,7,a+1024*3
270 SAVE @*,t,9,a+1024*4
280 SAVE @*,t,2,a+512
290 SAVE @*,t,4,a+512+1024
300 SAVE @*,t,6,a+512+1024*2
310 SAVE @*,t,8,a+512+1024*3
320 SAVE @*,t,10,a+512+1024*4
330 LET A=A+5120
340 NEXT t
350 RETURN
```



Hoofdprogramma:

Hier wordt ruimte gemaakt voor de filenamen/typen in n\$. a\$ is een hulp\$ om die typen op het scherm te laten zien in een door ons gekozen formaat van drie letters.

```
360 DIM N$(81,13)
370 DIM a$(12,3)
380 RESTORE 390: FOR k=1 TO 12: READ k$: LET a$(k,1 TO 3)=k$: NEXT k
390 DATA "BAS","D.A","$.A","CDE","S48","M/D","SC$","SPL","128",
"OPT","EXE"

400 CLS : PRINT "DIRECTORIE SORTEREN"....."DOE DE TE SORTEREN D
ISK IN DRIVE"
410 PRINT #0;" *** DRUK EEN TOETS ***"
420 IF INKEY$<>" " THEN GO TO 420
430 IF INKEY$=" " THEN GO TO 430
```

```

440 INPUT INKEY$: PRINT AT 0,0;"DIRECTORIE SORTEREN"....."IK LE
ES NU DIRECTORIE UIT
450 LET C=0: LET B=3
460 GO SUB 80

```

Om een goede alfabetische sortering mogelijk te maken:

```

470 PRINT "IK ZET ZE IN HOOFDLETTERS"
480 LET m=40000
490 FOR t=0 TO 79: LET n$(t+1,12 TO )=STR$ t
500 FOR k=0 TO 10
510 LET L=A+T*256+K: LET M=PEEK L: IF m=0 OR PEEK (1-k)=0 THE
N LET m=127
520 IF m>127 THEN LET m=128
530 IF m>96 AND m<123 THEN LET m=m-32
540 LET n$(t+1,k+1)=CHR$ m
550 NEXT k
560 LET N=CODE N$(T+1,1): IF N=127 THEN LET N=12

```

En u ziet de namen voorbij komen op het scherm.

```

570 PRINT AT 10,0;VAL (n$(t+1,12 TO ))+1;TAB 4;n$(t+1,2 TO 11);
" ";A$(N): NEXT T

```

Keuze uit sorteren op naam of type:

```

580 PRINT ""SORTEREN OP TYPE OF NAAM? T/N ":
590 IF INKEY$<>" " THEN GO TO 590
600 LET K$=INKEY$: IF K$=" " THEN GO TO 600
610 PRINT K$

```

De parameters voor de sorteerroutine.
Hier kan uw eigen basic sorteerroutine dus komen om n\$ te sorteren. n\$(1) bevat het het typenummer. (0,1,2,3 etc)

```

620 LET Q=0: LET I=2: LET Q$=""
630 IF K$="T" OR K$="t" THEN LET I=1

```

poken van de parameters in de routine :

```

640 GO SUB 880

```

en de routine aanroepen.

```

650 RANDOMIZE USR 63800

```

De string is gesorteerd, nu de eigenlijke directory van 80 x 256 bytes nog!

```

660 FOR P=0 TO 1
670 PRINT "IK MAAK TRACK ";0+P*2;" EN ";1+P*2;" AAN. ";
680 FOR K=0 TO 1

```

L = 0 to 39 voor de eerste 40 namen in de eerste 2 tracks.

```

690 FOR L=0 TO 39
700 LET O=VAL N$(p*40+L+1,12 TO ): IF (O>39 AND K=0) OR (O<40 A
ND K=1) THEN GO TO 730

```

De parameters worden nu in de verplaatsingsroutine gepoked en de routine schuift de blokjes van 256 bytes op hun plaats in de nieuwe directory op adres 50240. De eigenlijke directory staat nog op adres 40000.

```
710 LET O=O-K*40: LET M=50240+L*256: LET N=40000+O*256: POKE (U
SR "A"+2),INT (N/256): POKE (USR "A"+1),N-256*PEEK (USR "A"+2):
POKE (USR "A"+5),INT (M/256): POKE (USR "A"+4),M-256*PEEK (USR "
A"+5)
720 RANDOMIZE USR (USR "A")
730 NEXT L
```

SAVE de eerste 2 gesorteerde directorytracks.

```
740 LET C=2: LET B=3: GO SUB 60
```

en herhaal dit met de laatste 40 namen/ 2 tracks.

```
750 NEXT K
760 LET C=O+P*2: LET B=1+P*2: GO SUB 60: GO SUB 210: PRINT " KL
AAR"
770 NEXT P
780 FOR K=1 TO 100: NEXT K
790 CAT *
```

Wat mij betreft nu wat overbodige luxe, maar misschien vindt u het wel handig.

```
800 INPUT "WIJZIGEN, RUN, STOPPEN? W,R,S "; LINE K$
810 IF LEN K$=0 THEN GO TO 800
820 IF K$(1)="R" OR K$(1)="r" THEN RUN
830 IF K$(1)="S" OR K$(1)="s" THEN STOP
840 FOR k=0 TO 79: LET n$(k+1,12 TO )=STR$ k: NEXT k
850 CLS : LET c=0: LET b=1: GO SUB 60
860 GO TO 580
870 STOP
```

Einde van het hoofdprogramma.

Subroutine om de parameters in de sorteerroutine te poken:

```
880 POKE 23500+40248,CODE "n"
890 POKE 23501+40248,LEN q$
900 POKE 23502+40248,i
910 POKE 23503+40248,q
920 POKE 23505+40248,(80*13)/256
930 POKE 23504+40248,(80*13)-256*PEEK (23505+40248)
940 IF q$="" THEN RETURN
950 FOR f=0 TO LEN q$-1: POKE 23530+40248+f,CODE q$(f+1): NEXT
f
960 RETURN
```

Wiss de onderste 2 schermregels-ROM routine:

```
970 RANDOMIZE USR 3438: RETURN

980 CLS #: PRINT 63400-(PEEK 23653+256*PEEK 23654);" BYTES VRIJ
"
9988 STOP
9999 CLEAR : SAVE d*"DIRSORT.BS" LINE 10
```


Mocht u om de een of andere reden uw originele directory nodig hebben, bv omdat er een fout is opgetreden en er nog maar een halve directory gesaved was of u bedenkt zich achteraf, dan is er een methode om hem te redden als u de computer nog niet gereset hebt of een ander programma geladen.
De gegevens bevinden zich nog vanaf adres 40000 in RAM.

Een simpel:

```
LET a=40000
FOR t=0 TO 3
FOR s=1 TO 10
SAVE $*,t,s,a
LET a=a+512
NEXT s
NEXT t
```

is dan voldoende.

U kunt die optie zelf simpel inbouwen vanaf regel 800.

Al met al heb ik dit redelijk gebruiksvriendelijk programma met plezier op een aantal volle schijven losgelaten en was zeer te spreken over het resultaat.

De directories zien er gesorteerd keurig overzichtelijk uit en er hebben zich bij het uittesten geen noemenswaardige problemen voorgedaan.

Het programma werkt direct op zowel 40 als 80 track-drives. Alleen de enkeling die met single-DENSITY drives werkt en maximaal 40 programmanamen in zijn directory kwijt kan moet hier en daar wat FOR/NEXT lussen aanpassen.

Veel plezier met dit programma.



128K "EXECUTE" RENUMBER ROUTINE

Eindelijk is hier dan een werkelijk bruikbaar voorbeeld van een EXECUTE file.

Na een voorzet van Jaap Juursema heb ik "EXEC.MAKER" geschreven, een programma waarmee executefiles aangemaakt kunnen worden die na inladen de standaard *RENUMBER* functie in de 128K-mode naar believen wijzigen. Zowel eerste regelnummer als de stapgrootte zijn instelbaar.

De executefiles worden los van "EXEC.MAKER" gebruikt. U heeft "EXEC.MAKER" alleen nodig om een executefile aan te maken met het gewenste eerste regelnummer en stapgrootte.

Omdat dit eigenlijk het eerste echte EXECUTEFILE in de Nieuwsbrief wordt zal ik het zo uitgebreid mogelijk uitleggen als handvat voor uw eigen experimenten.

Vrouw (of man) aan de afwas? Kinderen naar bed? O.K. hou je vast daar gaan we weer!!

```
10 PRINT "AANPASSEN RENUMBER 128K SPECCIE"  
20 PRINT "DIT PROGRAMMA MAAKT HET EXECUTE FILE."  
30 INPUT "EERSTE REGELNUMMER ? ";A  
40 PRINT AT 10,0;"HERNUMMEREN VANAF : ";A  
50 INPUT "STAPGROOTTE ? ";B  
60 PRINT "STAPGROOTTE : ";B
```



Tot nu toe zijn de twee benodigde variabelen voor eerste regelnummer en de stapgrootte binnen gehaald. Een executefile is echter een stukje machinecode van maximaal 512 bytes.

Die code wordt nu aangemaakt door de FOR/NEXT lus in regel 80 en is in dit geval slechts 11 bytes lang. Mede daardoor is de code ook voor de absolute beginner redelijk te volgen. (Denk ik.)

```
70 RESTORE  
80 FOR F=40000 TO 40010: READ G: POKE F,G: NEXT F  
90 DATA 82,A,50,148,91,82,B,50,150,91,201
```

Dit is dus de essentie van ons programma "EXEC.MAKER". Vijf regels met twee INPUTS een FOR/NEXT en DATA zou voldoende zijn geweest.

De FOR/NEXT lus POKEd de volgende routine vanaf adres 40000 :

82,A	LD A,n	;Laad reg.A met de waarde van basicvar.A (niet verwarren). Waarde is vergrepen bij de INPUT in regel 30.
50,148,91	LD (23444),A	;Laad de inhoud van adres 23444 met waarde van reg. A

Juist, dit is hetzelfde als POKE 23444,A in basic!

82,B	LD A,n	;Laad reg.A met de waarde van basicvar.B (INPUT regel 50)
------	--------	--

50,150,91 LD (23448),B ;Laad de inhoud van adres 23446
met de waarde van reg. A

oftewel, POKE 23448,B in basic equivalent.

We moeten nu het executefile naar schijf schrijven. In de naam geven we een indicatie van de gebruikte parameters en stellen die dan ook samen uit "REN"+de cijfers van de variabelen a en b. Hierdoor is later in de directory te zien welk executefile we willen gebruiken.

```
100 PRINT ""SAVEN VAN EXECUTE-FILE"  
110 PRINT #1;"DISK IN DRIVE EN DRUK 'n TOETS."  
120 PAUSE 0
```

De syntax voor executefiles is na de naam een "x" gevolgd door het aanmaakadres in RAM van de code. (Zie ook uw handboek.)

```
130 SAVE d*"REN"+STR$ a+"_"+STR$ b+"x,40000  
140 CAT *;"REN*"  
150 PAUSE 0: NEW  
9999 SAVE d*"EXEC.MAKER"
```

Na het runnen NEWT "EXEC.MAKER" zichzelf daar u het niet meer nodig heeft. Na iedere NEW worden de normaal gebruikte parameters 10,10 in de systeemvariabelen teruggeplaatst. Om de nieuwe hernummer parameters te gebruiken moet u dus het executefile van schijf laden.

Het voordeel van executefiles is dat ze automatisch starten en GEEN ram-geheugen in de Spectrum in beslag nemen. Het executefile wordt in de Disciple RAM geladen en uitgevoerd.

Na het laden van het executefile volgt de prompt OK:01 en kunt u normaal verder werken, basic laden, hernummeren etc. en blijven de nieuwe parameters in werking totdat u een NEW ingeeft. U mag op ieder willekeurig moment besluiten het executefile te laden.

Programma's in RAM worden er NIET door aangetast.

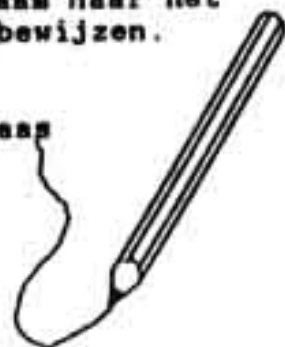
Zoals heb uitgelegd bij de gebruikte machinecode doet die niets anders dan twee POKES. De adressen 23444 en 23448 zijn de 128K systeemvariabelen waar de hernummerroutine zijn parameters vandaan haalt.

U mag dan ook de terechte conclusie trekken dat u ook met twee simpele pookjes kunt volstaan, nl POKE 23444,n en POKE 23448,n waarbij n staat voor regelnummer en stapgrootte. De praktijk leert echter dat er zoveel handige POKES bestaan dat u ze gegarrandeerd vergeten bent op het moment dat u ze nodig heeft.

Een simpel executefile met verwijzing in de naam naar het gebruik zal tot in lengte van dagen zijn nut bewijzen.



P. Faas



AUTOSYSTEM-LADER

Menigmaa! heb ik discussies gevoerd met mensen die op iedere geformatteerde schijf zonodig een systeemfile moesten zetten. Dat lijkt in eerste instantie misschien handig, maar het blijkt in de praktijk zeer onverstandig te zijn.

Ten eerste zijn er al diverse typen systeemfiles in omloop. Denk maar eens aan het "SYSPOKE", waarmee POKES in commerciële spelletjes kunnen worden geplaatst, of "SYSGRIJS" het systeemfile met de verbeterde grijs tinten voor de A4 printer en dan nog systeemfile 3A, 3B en de officiële 3c en 3d versies.

Welke van deze systeemfiles wilt u op iedere schijf plaatsen? Wanneer wilt u welk systeemfile gebruiken? Soms wilt u valsspelen en pookjes invoeren, maar dat systeemfile is gedeeltelijk overschreven en kan geen screen's meer printen.

En dus is het goed om minimaal EEN officieel, goed geïntialiseerd systeemfile achter de hand te houden.

Staat op iedere schijf een systeemfile, dan moeten ze ALLEMAAL worden aangepast als er iets verandert aan uw hardware, zoals een andere printer, tweede drive, ander driveformaat etc....!!

Plus nog eens de verwarring die ontstaat door ruilen van schijven waar systeemfiles van anderen op staan. Kortom, verwarring troef en vreselijk onhandig.

Een goed advies is daarom een "SYSTEM"schijf aan te maken. Op die schijf plaatst u het initialisatieprogramma en een op uw hardware ge-initialiseerd systeemfile.

TIP: SAVE dat systeemfile na de initialisatieprocedure zelf handmatig naar schijf. Gebruik dus niet de optie in het programma. Save het file dan weg dmv:

SAVE d*"SYS 3c" CODE 0,8192 !!(en niet 0,8656)

Dit om de vervelende bugs na "DISK WRITE protected" en "NO ROOM on disk" te vermijden. Laad NA zo'n fout dit systeemfile in en er kan niets gebeuren.

(De andere oplossing was POKE 86899,0 . Wist u dat nog?)

Behalve dat ene goed ge-initialiseerde systeemfile van 8192 bytes kunt u ook het POKESysfile en een ander sysfile naar keuze op die schijf plaatsen.

Op die schijf plaatst u ook deze schitterende "AUTOLOADER".

Wanneer u nu uw systeem opstart, stopt u de "SYSTEMDISK" in uw drive en typt RUN.

U komt in een menuscherm wat voor de 128K gebruikers er bekend uit zal zien.

Kies nu het gewenste systeemfile of toets op <ENTER> om de computer te resetten. Het gewenste systeemfile staat voor u klaar.

Bij de listing hoort een machinecode blokje, maar we geven de listing toch omdat u zelf wat tekst moet wijzigen, toegesneden op uw eigen situatie.

```
10 LET M48=PEEK 23562=5: LET M=128-(80 AND M48): LET a$="**"+S
TR$ M+" K-mode**"
30 PRINT #0;AT 0,0;"(c)1988 van Leeuwen Research Ltd."
```

Regel 60 zorgt voor de tekstjes in de menubalk, zodat u hier uw eigen filenamen kwijt kunt.

```
60 PRINT USR 84700;AT 5,5;a$, "NEW", "SYS 3c", "SYS 3d", "SYS POKE",
"SYS GRIJS", "CAT*!"
```

En afhankelijk van uw eigen filenamen moet u die uiteraard ook in de LOAD statements wijzigen.

```
70 IF PEEK 84699=1 THEN CLEAR 65535: NEW
80 IF PEEK 84699=2 THEN LOAD d*"Sys 3c"CODE : RUN
90 IF PEEK 84699=3 THEN LOAD d*"Sys 3d"CODE : RUN
100 IF PEEK 84699=4 THEN LOAD d*"Sys POKE"CODE : RUN
110 IF PEEK 84699=5 THEN LOAD d*"Sys GRIJS"CODE : RUN
115 IF PEEK 84699=6 THEN CLS : CAT *!: PAUSE 0: RUN
120 CLEAR 84695: LOAD d*"AUTOCODE.C"CODE 84696: RUN
130 STOP
9999 SAVE d*"AUTOLOAD.B" LINE 120
```

LET OP;

De basic is BREAK bestendig en komt ook na een Disciple-fout niet in basic terug, zodat u het programma moet MERGEN om aan te kunnen passen en denk erom, SAVEN met LINE 120!

Er kunnen ongestraft meerdere files toegevoegd worden in regel 60 en uiteraard de daarbij behorende actie vanaf regel 70. Afhankelijk van de gekozen plaats in het menu verandert ook de PEEK 84699 waarde waarmee de keuze bepaald wordt. Het wijst zich allemaal vanzelf verder.

Henk van Leeuwen leende de machinecode en broude deze prima bruikbare toepassing waarmee een perfecte systeemschijf is samen te stellen. Hartelijk dank voor de inzending Henk.

(p)Henk van Leeuwen



DE SCART-TV ALS RGB-MONITOR

Er zijn al diverse oplossingen aan de hand gedaan om van een TV een monitor te maken. Bijna iedere oplossing had ook wel wat nadelen. Zo was bij de "oude" simpele SCART aansluiting de BRIGHT functie niet meer bruikbaar. Het beeld stond continue in BRIGHT 1.

Klaas Roos uit Wormerveer bouwde deze schakeling en heeft nu een volwaardige monitor van zijn SCART-TV gemaakt.

Een en ander draait nu al tijden tot volle tevredenheid en deze schakeling is een aanrader voor een ieder die toch liever een draagbare TV bij zijn speccie wil hebben.

Het handje simpele componentjes is makkelijk op een stukje printboard te monteren, zonder moeilijke printlay-outs of etswerk.

Hier het verhaal van Klaas:

Met behulp van een paar simpele aanpastrapjes kunnen de TTL-nivo's van de Spectrum omgezet worden naar SCART-nivo's. Dat gebeurt hier met drie identieke transistortrapjes (T1..T6). De aanpassingstrapjes maken elk een uitgangssignaal met twee nivo's. Met P1, P2 en P3 in de middenstand varieert de de uitgangsspanning bij BRIGHT 1 tussen ca. 0,3 en 0,6 V en bij BRIGHT 0 tussen 0 en 0,3 V. Met de potmeters kan de overgangsspanning (0,3 V) worden ingesteld. Ze dienen alle drie in dezelfde stand te worden gezet. De instelling van de intensiteitsverhouding kan geschieden naar persoonlijke smaak.

Het CSYNC signaal wordt via een 1000 Ohm weerstandje rechtstreeks doorverbonden.

De benodigde voedingsspanning van +5 V moet u ergens van de 7805-stabilisator in de Spectrum aftakken. Doorverbinden naar een vrije pen in de 8 polige DIN connector is een mogelijke oplossing. Anders is een apart 5 volt voedinkje ook mogelijk. De stroomopname van de schakeling bedraagt ca. 150 mA.

Al met al een niet al te moeilijk klusje met een niet al te hoge investering.

Het mooiste is als u het geheel in een kastje inbouwd met aan de SCART kant een stukje kabel met konnektor en aan de andere zijde een kabel met 8 polige DIN. Wanneer het ding ook nog een eigen voeding heeft dan kan iedere willekeurige 128K en 128K+2 er op aangesloten worden. Dat is makkelijk als u van machine wisselt of de kabel met ingebouwde schakeling wilt doorverkopen aan een andere liefhebber.

Voor diegene onder ons die echt met twee linker handen geboren zijn blijft niets anders over dan op zoek te gaan naar die handige neef met soldeerbout. Geef hem dan wel even de tekening van het geheel die u op de pagina hierna(ast) vindt.

Ked.

SYSTEEMVARIABLEN-LISTER

Van onze junior programmeur Jasper vd Bent uit Diemen mochten we deze alleraardigste systeemvariabelen lister ontvangen.

Vooral voor mensen die regelmatig schijffjes ontvangen met onbekende systeemfiles is het handig om even de belangrijkste instellingen te bekijken om te zien of zo'n file nog blijven staan of moet worden aangepast.

Het programma is recht toe, recht aan geprogrammeerd en als een rechtgeaarde Spectrum fanaat begon ik direkt de basic aan te passen en te comprimeren met lussen en samenvoegen van regels. Na een paar uur bezig geweest te zijn was de winst iets meer dan 1Kb en van Jaspers programma was niet veel meer herkenbaar.

Dat zou voor hem niet leuk zijn, dus hier zijn originele basic, afgezien van wat kosmetische wijzigingen wat de schermafdrucken betreft en een ander type printroutine.

DENK BROM, iv en tv in PRINTS betekent dus INVERSE en TRUE video

```
10 PRINT "iv SYS-LISTER      J vd Bent 1989 tv""
40 FOR f=60000 TO 60010: READ g: POKE f,g: NEXT f
50 DATA 219,187,58,0,0,8,0,78,211,187,201
60 LET piek=0: GO SUB 550: LET col=piek
70 LET piek=1: GO SUB 550: LET dr1=piek
80 LET piek=2: GO SUB 550: LET dr2=piek
90 LET piek=3: GO SUB 550: LET srt=piek
100 LET piek=4: GO SUB 550: LET ntw=piek
110 IF col=0 THEN LET a$="FLASH OFF"
120 IF col<>0 THEN LET a$="FLASH ON"
130 IF dr1=40 THEN LET b$="40 TR/SS"
140 IF dr2=40 THEN LET c$="40 TR/SS"
150 IF dr1=80 THEN LET b$="80 TR/SS"
160 IF dr2=80 THEN LET c$="80 TR/SS"
170 IF dr1=168 THEN LET b$="40 TR/DS"
180 IF dr2=168 THEN LET c$="40 TR/DS"
190 IF dr1=208 THEN LET b$="80 TR/DS"
200 IF dr2=208 THEN LET c$="80 TR/DS"
210 IF dr1=0 THEN LET b$="OFF"
220 IF dr2=0 THEN LET c$="OFF"
230 IF ntw=0 THEN LET e$="OFF"
240 IF ntw=1 THEN LET e$="MASTER"
250 IF ntw>1 AND ntw<10 THEN LET e$="ASSISTANT"
260 IF ntw>9 AND ntw<64 THEN LET e$="PUPIL"
270 PRINT "ADRES PEEK NAAM  OMSCHRIJVING"
280 PRINT "-----": PRINT
290 PRINT "  0,":col;TAB 11;"RBCC";TAB 18;a$
300 PRINT "  1,":dr1;TAB 11;"TRAKS1";TAB 18;b$
310 PRINT "  2,":dr2;TAB 11;"TRAKS2";TAB 18;c$
320 PRINT "  3,":srt;TAB 11;"STPRAT";TAB 18;"STEPRATE"
330 PRINT "  4,":ntw;TAB 11;"NSTAT";TAB 18;e$
350 LET piek=11: GO SUB 550: LET prnt=piek
355 IF PRNT=1 THEN PRINT "  11,":prnt;TAB 11;"ZXPNT";TAB 18;"
CENTRONICS OFF": GO TO 535
360 PRINT "  11,":prnt;TAB 11;"ZXPNT";TAB 18;"CENTRONICS ON"
370 LET piek=5: GO SUB 550: LET wid=piek
380 LET piek=6: GO SUB 550: LET ped=piek
```




```

390 LET piek=7: GO SUB 550: LET lsp=piek
400 LET piek=8: GO SUB 550: LET lfd=piek
410 LET piek=9: GO SUB 550: LET lng=piek
420 LET piek=10: GO SUB 550: LET grp=piek
430 IF pcd=0 THEN LET b$="ESC CODES OFF"
440 IF pcd=1 THEN LET b$="ESC CODES ON"
450 IF grp=1 THEN LET e$="GRAPHICS ON"
460 IF grp=0 THEN LET e$="GRAPHICS OFF"
470 PRINT
480 PRINT " @ 5,";wid;TAB 11;"WIDTH";TAB 18;"CHARS/LINE"
490 PRINT " @ 6,";pcd;TAB 11;"PCDE ";TAB 18;b$
500 PRINT " @ 7,";lsp;TAB 11;"LSPC ";TAB 18;"n/72 LINESPACE"
510 PRINT " @ 8,";lfd;TAB 11;"LFEEED";TAB 18;"LINEFEEDS"
520 PRINT " @ 9,";lng;TAB 11;"LMARG";TAB 18;"LEFT MARGIN"
530 PRINT " @10,";grp;TAB 11;"GRAPH ";TAB 18;e$
535 PRINT #1;"ivAFDRUKKEN OP DE PRINTER ? (J/N)iv": PAUSE 0
540 IF INKEY$="J" OR INKEY$="j" THEN PRINT AT 0,0;"SYSLIST - HA
RDCOPY JASPER 1989": RANDOMIZE USR 3438: PRINT #1;"* VOOR DE D
ISCIPLE NIEUWSBRIEF *": SAVE SCREEN# 1: RUN
545 STOP
550 REM *****
560 REM * PEEK @ *
570 REM *****
580 LET piek=piek+664: IF piek>65535 THEN LET piek=piek-65536
590 IF piek=0 THEN POKE 60003,0: POKE 60004,0: GO TO 610
600 RANDOMIZE piek: POKE 60003,PEEK 23670: POKE 60004,PEEK 2367
1
610 LET piek=USR 60000
620 RETURN
9999 SAVE d*"SYSLIST.BS"

```

(p)Jasper vd Bent.

En u herkent vanaf regel 550 de PEEK-routine uit Nieuwsbrief 12. Jasper heeft daar handig gebruik van gemaakt door zijn lister er omheen op te bouwen.

Een mooi voorbeeld over gebruik maken van de "tools" die wij u aanreiken, hopelijk komen meer lezers af met eigen bedachte en geschreven programma's.

Weet u het nog? De nieuwsbrief als doorgeefluik voor u, de gebruikers van de Disciple/Plus-D!

P.Fans

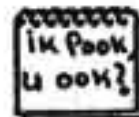


SUPERSNEL KLONEN, TUUT TUUT..

Er zijn al een aantal kloon programma's in de Nieuwsbrief gepubliceerd, met en zonder snelle (BASIC) loaders en savers. De snelsten gebruiken een interleave-faktor van 2 (dus eerst sektor 1, dan 3, enz) om een volledige track binnen te halen (2 disk omwentelingen nodig). Dit is vanuit BASIC de snelste manier naar vanuit machinetaal kun je nog net iets sneller. Vanuit machinetaal kun je namelijk de sectoren gewoon van 1-10 ophalen (1 omwenteling nodig). Het door mij geschreven (met dank aan Tonnie Stap) programmaatje zorgt voor het snel ophalen en wegschrijven van een track op/van ieder gewenst adres.

Het gebruik is eenvoudig:

POKE start+85, drive nummer
POKE start+84, track nummer
POKE start+88-884, load/save adres
RANDOMIZE USR start (load track) of
RANDOMIZE USR start+39 (save track)



De kode is niet relocatable maar door het "ORG"-adres in GENS te veranderen kun je het bijna overal neer zetten. Let er wel op dat nadat je geladen hebt wel het load/save adres weer goed zet voor je begint te saven. (Na gesaved te hebben moet je hier ook aan denken.) Het programma gaat namelijk gewoon verder op het adres waar de het laatst aangeroepen routine gebleven was.

Hier volgt een voorbeeld van een kloon programma van drive 1 naar 2 waarbij je ook nog wat kunt zien:

```
10 FOR t=0 TO 79
20 FOR s=0 TO 128 STEP 128
30 POKE 880088-884,18384
40 POKE 80084,t+s
50 POKE 80085,1
60 RANDOMIZE USR 80000
70 POKE 880088-884,18384
80 POKE 80085,2
90 IF s=0 THEN OUT 91,T-1
100 RANDOMIZE USR 80039
110 NEXT s
120 NEXT t
```

Tijd is geld!



Regel 90 zorgt ervoor dat de disk-controller "weet" dat drive 2 een track achterligt bij drive 1. (poort 91 is het track register van de disk-controller) De machinetaal routines zijn eenvoudig in bijvoorbeeld het kloon programma van NB 12 in bouwen, denk dan wel even om het adres van de kode!

WAARSCHUWING: Als u het basicprogramma runt dan wordt direkt begonnen met klonen van drive 1 naar drive 2!

Na van de verbazing gekomen te zijn, dat was snel! Hier de assemblerlisting, voor alle machinetaal liefhebbers en echt het laat zich net zo makkelijk lezen als basic, kijk maar;

0010		ORG	80000	;aanmaakadres code
0020	LTRACK	CALL	SETDRV	;selekteer drive 1 of 2
0030		LD	A,(TRACK)	;laad A met tracknummer
0040		LD	D,A	;laad D met tracknummer (via A)
0050		LD	E,1	;laad E met sectornummer 1
0060		LD	HL,(LADDR)	;laad HL met laadadres track
0070	LDLOOP	CALL	SETBUF	;stel laadadres in (zie SETBUF)
0080		PUSH	DE	;bewaars het track en sectornummer
0090		RST	8	;op de stapel en lees de sector
0100		DEFB	#3F	;aangegeven in registerpaar DE
0110		POP	DE	;Haal de tr/sectnr. van de stack
0120		JR	C,ERROR	;Is er een foutje geconstateerd?
0130		LD	HL,(LADDR)	;Niet, verhoog dan het laadadres
0140		LD	BC,512	;met 512 bytes voor de volgende
0150		ADD	HL,BC	;sector.
0160		LD	(LADDR),HL	;Bewaars nieuwe laadadres
0170		INC	E	;Verhoog sectornummer met 1
0180		LD	A,E	;en zet sectornummer in reg. A
0190		CP	11	;Zijn er al 10 geweest?
0200		JR	Z,RESBUF	;Ja, dan buffer resetten en klaar.
0210		JR	LDLOOP	;Nee, dan volgende inladen.

Het volgende stukje code is vrijwel identiek aan het hierboven omschreven deel. Alleen de DEFB in regel 0300 is anders. De gevolgen zijn wel anders want ipv RSAD wordt nu WSAD uitgevoerd.

RSAD = Lees de sector waarvan track en sectornummer in DE staat
 WSAD = Schrijf de sector aangegeven in DE

0220	STRACK	CALL	SETDRV	;
0230		LD	A,(TRACK)	;
0240		LD	D,A	;
0250		LD	E,1	;
0260		LD	HL,(LADDR)	;
0270	SALOOP	CALL	SETBUF	;
0280		PUSH	DE	;
0290		RST	8	;
0300		DEFB	#3E	;
0310		POP	DE	;
0320		JR	C,ERROR	;
0330		LD	HL,(LADDR)	;
0340		LD	BC,512	;
0350		ADD	HL,BC	;
0360		LD	(LADDR),HL	;
0370		INC	E	;
0380		LD	A,E	;
0390		CP	11	;
0400		JR	Z,RESBUF	;
0410		JR	SALOOP	;



Buiten deze beide "hoofdprogramma's" die elk EEN volledige track inlezen of wegschrijven zijn er nog wat subroutines.

0420	SETDRV	LD	A,(DRIVE)	;(DRIVE) is het adres waarin het
0430		RST	8	;drivenummer 1 of 2 gepoked wordt
0440		DEFB	#21	;en de IF1 hookcode #21 voert de
0450		RET		;selectie uit.

0460	TRACK	DEFB	0	;POKEadres voor tracknummer
0470	DRIVE	DEFB	0	;POKEadres voor drivenummer
0480	LADDR	DEFW	0	;POKEadres voor laadadres vd track
;				
0490	RESBUF	LD	HL,7126	;Reset de DRAM buffer in Disciple
;				
0500	SETBUF	LD	BC,6851	;De DRAM buffer van 512 bytes
0510		OR	A	;wordt verplaatst vanuit de
0520		SBC	HL,BC	;Disciple RAM naar een adres in de
0530		IN	A,(187)	;Spectrum RAM waar we de sector
0540		LD	(6866),HL	;geladen willen hebben. (6866) is
0550		OUT	(187),A	;de sysvar waar dat bewaard wordt.
0560		RET		;
;				
0570	ERROR	LD	(ERR_NR),A	;Oeps, er is een lees/schrijf
0580		CALL	RESBUF	;foutje geconstateerd, dus
0590		IN	A,(187)	;programma afbreken met een
0600		RST	32	;foutmelding!
0610	ERR_NR	DEFB	0	;adres voor het foutnummer



Rudy Bienna.

DE THERMO PRINTER AFGESCHREVEN????

Tja, door de een verguisd en door de ander geprezen, de meningen over dit stiefkindje uit de printerfamilie zijn aardig verdeeld.

Het leek mij daarom geen slecht idee om eens wat nader in te gaan op het fenomeen "THERMISCHE PRINTER" en het nuttige gebruik ervan met de Spectrum/Disciple (of Plus-D) combinatie.

De bekendste onder dit soort printers zijn zonder twijfel de BROTHER-HR5 en de TXP (of TPX) 1000 van General Electric. Deze laatste verdiend verre de voorkeur door zijn robuustheid en printkwaliteit.

Maar wat is nu eigenlijk het verschil tussen een thermische en een inktlint printer?

Beide werken volgens het "DOT MATRIX" impact systeem, 9 of 24 naaldjes gebundeld in een matrix die in een door de komputer gedicteerde datastroom in een bepaald patroon naar voren komen (impact) om tegen inktlint of warntegevoellig papier te drukken waardoor de informatie op dat papier zichtbaar gemaakt wordt.

De inktlint printer werkt koud, dat wil zeggen dat de naaldjes tegen het inktlint hameren en zo, afhankelijk van de kwaliteit van dat lint een afdruk producerend.

De thermische printer werkt warm, er wordt een elektrische lading door de naaldjes gevoerd waardoor ze warm worden. Nu kan er op twee manieren zichtbare informatie afgegeven worden:

- Door het hameren tegen een carbonlint waarbij het carbon door de warme naaldjes op het (gewone) papier gesmolten word.
- Door het hameren tegen warstegevoelig (thermo) papier waarbij ieder naaldje een zwart puntje (dot) produceerd waardoor de informatie zichtbaar gemaakt wordt.

De inktlint printer dankt zijn populariteit voornamelijk aan zijn eenvoud. Hij is snel, een inktlint gaat lang mee en een velletje papier is goedkoop en volop te verkrijgen.

Een thermoprinter echter gebruikt dure, slechts eenmaal te gebruiken carbonlinten of het ook niet echt goedkope thermische papier dat bovendien door zijn uiterlijk en kwetsbaarheid niet geschikt is voor directe verzending (het vergeeld snel). Een bijkomend nadeel is bovendien zijn trage werking.

Nu hoor ik u denken, wat moet ik er dan mee? En dan zijn we meteen bij de kern van dit artikel. De thermo printer is niet dood hij leeft!!!

Voor de juiste doeleinden gebruikt, de goede eigenschappen van dit type ten volle benuttend is het resultaat een afdruk met een prijs prestatie verhouding die er absoluut zijn mag!!

De impact van de warme naaldjes op thermo papier leveren "dots" op die allemaal even goed van kwaliteit zijn. Een zwart vlak wordt zo ook een zwart vlak! Met andere woorden, de Thermo printer in door zijn prijs prestatie verhouding de laserprinter voor de arme man!!! En dat zijn Spectrumgebruikers allemaal.

Om dit artikel te rechtvaardigen heb ik uiteraard een aantal printproeven gemaakt en vergelijkingen getrokken met de drie printer types, te weten de:

-EPSON LX-800	een inktlint printer van Fl 850.-
-TXP-1000	een thermische printer van Fl 150.-
-Laserwriter-II	een laserprinter van Fl 7000.-

Met alle drie de machines heb ik een aantal gelijksoortige graphische afbeeldingen geprint die daarna 2x gekopieerd werden (een kopie van een kopie) en gecontroleerd op kwaliteitsverlies.

Het zal nauwelijks een verrassing zijn te vermelden dat de laserwriter-II het beste resultaat gaf. Ook geen verrassing leverde de uitslag van de Epson LX-800 op, een totaal onbruikbare afdruk, vervaagd en rafelig.

De relevantie van deze tests was toch wel degelijk de thermo printer TXP-1000 van General Electric. De titel "laserprinter voor de arme man" maakte hij volkomen waar...!! Zwart bleef zwart en de afdruk bleef ook na het kopiëren scherp en uiterst bruikbaar voor bv DTP.

KONKLUSIE:

Wie zich wel eens bezighoud met Desktop publiching of aanverwante werkzaamheden (er is nu per slot van rekening een goed DTP programma door Willem Lageman uitgebracht) zal zeker niet

bekocht zijn met de aanschaf of gebruik van een thermische printer, laten we maar eens naar de kosten kijken.

- Het aanbod op de tweede hands markt is goed en de prijzen vaak nog beter (100.- tot 150.-)
- Thermisch papier is welliswaar duurder dan gewoon papier maar toch best betaalbaar (15.- tot 20.- per rol van 50m)
- Het eindresultaat mag gezien worden, ook na het noodzakelijke kopiëren voor opslag en verwerking.
- En het was al opgemerkt maar de prijs prestatieverhouding is echt te gek en zeg nou zelf, bij een Spectrum van 200.- met een Disciple van 325.- of een Plus-D van 200.- past ook uitstekend de prijs van een thermische printer van 150.-, zelfs om hem er als tweede printer naast te hebben!

Voor diegene die nog ergens een thermoprinter op zolder hebben staan en voor hen die dit artikel wel aanspreekt maar nog niet zoveel ideeën voor het gebruik hebben volgen hier nog een paar ideetjes.

- affiches/folders
- Persoonlijke berichten (geboorte, jubilea, verhuiskaarten enz)
- verenigings organen en periodieken
- Scripties en rapporten met grafieken enz.
- Financiële verslagen, al of niet gelardeerd met grafieken.
- DTP presentaties.

Zo, dat was het. Aan de slag maar met het tweede leven van de vaak ten onrechte verguisde thermische printer.

Carl Faas.



DISKFILER

PROGRAMMA VOOR HET OPTIMAAL REORGANISEREN
EN BEHEERSEN VAN UW DISKETTE-BESTANDEN.

CAPACITEIT TOT LIEFST 2000 FILES!

LEEST DIRECTORIES UIT, DOORZOEKT
SUPERSNEL BESTANDEN, VERANDERD,
SORTEERT EN BEWAARD.

BULK ERASE OPTIE.

GESORTEERD LIJSTEN PRINTEN.

2 DRIVE OPTIE.

ZEER GEBRUIKSVRIENDELIJK.

TE BESTELLEN DOOR STORTING VAN FL 12,50 OP
GIRO [REDACTED], T.N.V. P. FAAS, RED.NWSBR, [REDACTED].

DISCIPLE HANDBOEK

NEDERLANDS
TALIG

FL. 12.50

TE BESTELLEN DOOR STORTING
VAN FL 12.50 OP GIRO [REDACTED]
TNV. P. FAAS, RED. NWSBR.
[REDACTED]