

VAN DE REDAKTIE	robert van den broek	3
SOFTWARE BIBLIOTHEEK	theo verberkt	5
BLOKEDITER	cees van dijk	6
DAI EN DE VUT	ruud muller	10
RANGOON	ruud muller	11
HIGH SPEED DATA LOADER	henk rison	13
"X" - BUS CARD	henk rison	20
DIRSORT	cees van dijk	21
"FWP" DUBBEL KOLOM	ton van es	25
MAASGRAFIEKEN	cees van dijk	28
TCCR-50	inno broekman	31
MDCR	theo verberkt	35
BESTELBILJET	voor iedereen	

ADRESSEN:

Voorzitter : Kees Jagerman, Ilperveldstraat 77
1024 FJ Amsterdam
Tel. 020 - 367156

Secretaris a.i. : Theo Verberkt, Van Buerenstraat 13
+ Hardware en Software , 5256 KL Oudheusden
Tel. 04162-2667

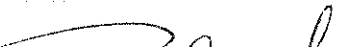
lijk meer behoefte aan al lekker met vakantie. Nog 3 weken slechts en dan ga ik gestrekt. Maar dat even terzijde en eerst naar wat ernstige zaken. Mijn bezoek op 5 mei aan Prodata in Brussel heeft nog al wat informatie opgeleverd. De belangrijkste is, dat uiteindelijk het doek voor onze "old grande" is gevallen. Jammer dat dit tijdstip is gekomen. Het hing al tijden in de lucht, we wisten het eigenlijk wel. Maar toch was er nog steeds dat kleine beetje hoop dat ons bezig hield of er eventueel een vervanger of een verbeterde modernere versie zou verschijnen. Helaas, geen directe vervanger maar een grote 16 bitter die kwa prijs (prestatie genoeg) nu niet direct binnen onze budget valt. Het doet mij wel een beetje zeer dat het eind inzicht is. Bij jullie ook?. Hopelijk valt onze GG niet direct als een plomppudding uit elkaar. Er zijn nog genoeg leuke dingen om te ontwikkelen.

Deze DAITA is erg dun wat het aantal artikelen aangaat. Goede artikelen overigens. Vrienden, hebben wij onze inspiratie verloren of zijn we nog niet uit de winter slaap. Gelukkig hebben onze vaste programmamakers ons dit keer weer niet in de steek gelaten. De belangrijkste deze keer: een bijdrage van Cees van Dijk voor de 1541, een BLOKEDITOR en een DIRSORT. Van Ton van Es een verbetering van FWP met betrekking de dubbele pagina opmaak. Theo Verberkt leverde het programma van de MDCR. Deze is zo dik dat we deze apart moeten bestellen. Henk Rison heeft het gevolg van de HIGH SPEED DATA LOADER afgemaakt. Vele kennen dit verhaal reeds uit DAINAMIC. Ruud Muller doet een duit in het zakje door alvast de goedheiligman te spelen. Lees zijn verhaal over de uitverkoop van zijn programma,s. Prima, alleen de prijslijst is nog niet aangepast. Ook het verhaal van het spel RANGOON is van Ruud. Dit is een echt mooi spel. Inno Broekman heeft een interessant verhaal over een de TCCR-50 recorder ingezonden. Helaas kan ik de beloofde uitbreiding van de LICHTKRANT niet nakomen opdat mijn machientje bij Theo op de operatietafel ligt.

De prachtige grafiek op de voorpagina is van Cees van Dijk. Het programma waarmee dit is gemaakt heet MAASGRAFIEKEN en is in dit nummer te vinden.

Het Fido gebeuren. Van twee kanten heb ik vernomen dat het programma van Gruiters onder handen wordt genomen. Mogelijk horen wij hier spoedig meer over. Inverband met het reorganiseren en udaten van FIDO-KAMPEN-1 zal deze van 10 t/m 17 juni as. niet bereikbaar zijn.

Zo dit was het weer. Stuur bijtijds de copij voor het volgende nummer in. Inzenden graag vóór 6 september as. In ieder geval nog veel plezier met deze inhoud. En voor iedereen een hele fijne en zonnige vakantie toe gewenst en dan tot 20 september.



Misschien verduidelijkt het een en ander.

pen 1:	Chassis-aarde, wordt optioneel toegepast.
pen 2: TxD	(Transmitted Data) Lijn voor uit te zenden data.
pen 3: RxD	(Received Data) Lijn voor te ontvangen data.
pen 4: RTS	(Request To Send) De DTE zet dit signaal 'aan', wanneer hij klaar staat om te zenden naar de modem (DCE).
pen 5: CTS	(Clear To Send) De DCE (modem) zet deze pen op 'aan' zodat de DTE weet dat hij kan gaan zenden. Deze pen wordt 'aan' gezet wanneer de Data Set Ready (pen 6) en de Request To Send (pen 4) eerder 'aan' zijn gezet.
pen 6: DSR	(Data Set Ready) De modem geeft de DTE te kennen dat hij gereed staat om te ontvangen. Het betekent echter nog niet dat er al data door de modem wordt ontvangen. Er is een verbinding tot stand gebracht met een communicatiekanaal, bijvoorbeeld doordat het uitzenden van de antwoordtoon is beëindigd.
pen 7: GND	(Ground) De retourlijn of signaal-aarde. Benodigd bij de datalijnen TxD en RxD.
pen 8: DCD	(Data Carrier Detect) De modem (DCE) reageert met een 'aan'-signaal wanneer op de telefoonlijn bruikbare signalen aanwezig zijn. De modem zal een 'uit' genereren, wanneer op deze lijn geen signaal of een signaal van te slechte kwaliteit aanwezig is.
pen 9:	Onbenoemd, wordt echter soms voor positieve DCE testspanning gebruikt.
pen 10:	Idem, is echter gereserveerd voor negatieve Data Set testing.
pen 11:	Niet van toepassing.
pen 12: SDCD	Bedoeld om secundaire draaggolf te detecteren.
pen 13: SCTS	Voor secundaire Clear To Send.
pen 14: STxD	Secundaire lijn voor uit te zenden data.
pen 15: TC	(Transmitter Signal Element Timing) klok behorend bij signaal elementen afkomstig van de zender.
pen 16: SRxD	Secundaire lijn voor te ontvangen data.
pen 17: RC	(Receiver Signal Element Timing) klok behorend bij signaalelementen afkomstig van de ontvanger.
pen 18:	Niet van toepassing.
pen 19: SRS-T	(Secondary Request To Send).
pen 20: DTR	(Data Terminal Ready) Signaal om in 'aan' toestand aan te geven, dat de DTE aanwezig is of klaar staat voor de modem. Het signaal van de DTE naar de modem gegeven.
pen 21: SQ	(Signal Quality Detector) De modem kan de signaalkwaliteit aan de DTE blijven melden.
pen 22: RI	(Ring Indicator) de modem kan hiermee de DTE melden dat hij wordt 'opgebeld'.

* Viditel-pakket monitor-mlp/monitor-basic telesoftware tranciever	F 40,--	F 45,--
* DAI-dres	F 30,--	F 35,--
* Perfect Editor II op KEN-DOS	F 45,-- F 55,--	F 50,--
* Boekhouding (huishoud) boodschappenlijst	F 40,--	F 45,--
* Logic simulator 3D mathematic graphics	F 40,--	F 45,--
* DBMS (data base management)	F 30,--	F 35,--
* 6800 Cross-assembler	F 30,--	F 35,--
* RS232 communication programma Terminal-emulator	F 30,--	F 35,--
* Games tape 1:		
The vally (uitleg + spel) Advancer 1 en 2	F 30,--	F 35,--
* Postzegel-verzameling	F 30,--	F 35,--
* Programmotheek tapecontrol/demo's	F 30,--	F 35,--
* RAMBLE DAI-DOS KEN-DOS floppy		F 45,--
* SHIFT-OTHELLO-DRAWING	F 20,--	F 25,--
* DCR-DIRECTORY LISTING Gotische maandkalender FGT	F 30,--	F 35,--
* VLAGGEN-GRATEX-ANAGRAM	F 20,--	F 25,--

Voor bestelling en van software kunt U zich wenden tot :

Theo Verberkt,
Van Buerenstraat 13
5256 KL Oud-Heusden
Tel:04162-2667.

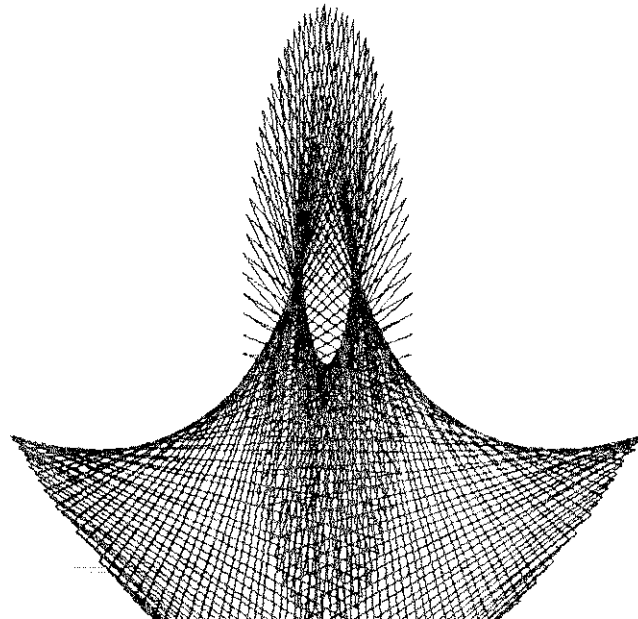
verschijnt in links op het scherm, terwijl in het rechterveld de ASCII-betekenis gegeven wordt. Beide kunnen via cursorbesturing eenvoudig gewijzigd worden. Het gewijzigde blok kan weer teruggeschreven worden op de schijf. Bij een file kan op simpele wijze elk volgend blok van de file opgeroepen worden.

Het programma is vooral nuttig bij bestudering van het systeem en bij operaties in de directory (inhoudsopgave op spoor 18). Het wijzigen in files kan een foutmelding tot gevolg hebben als men niet tevens de checksum aanpast.

Het programma kan bijv als volgt op de schijf gezet worden:
Blokeditor/O (FDD- monitor in ml)
Blokeditor/E (het basic-programma)
zodat de zaak gestart wordt met UEL "Blokeditor".

Veel plezier ermee.

C.W.A. van Dijk
Watermunt 5
8265 EL Kampen



```

20 REM LEES HSCI UIT VIDEO RAM
25 YD=(23-Y)*#86
30 AD=#BFE7-YD-X(K)*2:B=PEEK(AD):GOSUB 50:BH=B SHL 4:B=PEEK(AD-2):GOSUB 50:
   G=BH+B:RETURN

40 REM MAAK GETAL UIT ASCI
50 IF B<#30 THEN B=-#100:RETURN
60 IF B<#3A THEN B=VAL(CHR%(B)):RETURN
70 IF B<#41 THEN B=-#100:RETURN
80 B=VAL(CHR%(B-#11))+10:RETURN

89 REM *****
90 PRINT CHR%(12);:CURSOR 5,3:PRINT "CHAR.DEL:  ANDER VELD"
100 CURSOR 5,2:PRINT "SHIFT ^: VOLGEND BLOK"
110 CURSOR 5,1:PRINT "SHIFT ";CHR%(140);": SCHRIJF BLOK"
120 CURSOR 5,0:PRINT "TAB: NIEUW SPOOR/BLOK";
160 N=1:IN%=" "
170 OPEN4=#403:PRNT15=#40C:PUT4=#42D:GET4=#412:INP15=#41E:CLOSE4=#424
180 RESTORE:FOR K=1 TO 16:READ X(K):NEXT
190 DATA 2,4,7,9,12,14,17,19,22,24,27,29,32,34,37,39
200 CURSOR 5,22:INPUT "SPOOR:";IN:IF IN>35 OR IN<1 THEN 220:GOSUB 1920:SP%=
   IN%
210 INPUT " - BLOK:";IN:IF IN<0 THEN 220:GOSUB 1920:BL%=IN%:PRINT :GOTO 230
220 PRINT "FOUTE INV0ER":GOTO 200
230 IF SP%=" 0" THEN 200
240 OPEN%="#":CALLM OPEN4,OPEN%
250 OP%="01 4 0"+SP%+BL%:CALLM PRNT15,OP%

260 REM VOLGEND SPOOR&BLOK
270 OP%="B-P 4 0"
280 CALLM PRNT15,OP%:CALLM GET4,IN%:GOSUB 1900:SP1%=IN%:CALLM GET4,IN%:
   GOSUB 1900:BL1%=IN%

300 REM BLOK OP SCHERM
305 IN%=" "
310 CALLM PRNT15,OP%
320 CURSOR 0,20
330 FOR I=0 TO 15:K=0:PRINT HEX%(I);" ";
340 1 FOR J=44 TO 59:CALLM GET4,IN%:IN=ASC(IN%):PRINT RIGHT%("00"+HEX%(IN),
   1 2);
350 2 IF CURX MOD 5=1 THEN PRINT " ";
360 2 X=CURX:CURSOR J,CURY
370 2 IF IN<32 OR IN>127 THEN PRINT ".,:GOTO 390
380 2 PRINT IN%;
390 CURSOR X,CURY:NEXT:PRINT :NEXT
400 CALLM CLOSE4
410 CURSOR 0,20
420 POKE #74,0:POKE #75,#FF

```

```

1180,1180:IF UIT=0 THEN 1010:ON UIT GOTO 1500,1550
1040 PRINT CHR$(G);:X=CURX:CURSOR X(K),CURY:PRINT RIGHT$("DD"+HEX$(G),2);
1042 IF X(XMX+1 THEN 1044:IF CURY>YMN THEN 1043:CURSOR XMN,YMX:GOTO 1010
1043 CURSOR XMN,CURY-1:GOTO 1010
1044 CURSOR X,CURY:GOTO 1010
1180 XMN=2:XX=40:CURSOR X(K),CURY:GOTO 1210

1199 REM ***ASCII-VAK LINKS*****
1200 CURSOR 2,20:XMN=2:XX=40:YMN=5:YMX=20
1210 IF CURX>40 THEN 1010:GOSUB 1800:REM K BEREKENEN
1215 IF CURX MOD 5=1 THEN CURSOR CURX+1,CURY
1220 G=GETC:IF G=8 THEN 1390:IF G=9 THEN 1600:IF G<16 THEN 1220
1230 UIT=0:IF G>15 AND G<24 THEN ON G-15 GOSUB 1260,1280,1300,1340,1400,1410,
1390,1390:IF UIT=0 THEN 1210:ON UIT GOTO 1500,1550
1235 IF (G>47 AND G<58) OR (G>64 AND G<71) THEN 1240:GOTO 1210
1240 PRINT CHR$(G);:X=CURX:Y=CURY:CURSOR K+43,Y:GOSUB 25:IF G<32 OR G>127
THEN 1241:PRINT CHR$(G);:GOTO 1242
1241 PRINT " ";
1242 IF X(XMX+1 THEN 1244:IF CURY>YMN THEN 1243:CURSOR XMN,YMX:GOTO 1210
1243 CURSOR XMN,CURY-1:GOTO 1210
1244 CURSOR X,CURY:GOTO 1210

1250 REM CURSORBEWEGINGEN
1260 IF CURY=YMX THEN CURSOR CURX,YMN:RETURN:REM OP
1270 CURSOR CURX,CURY+1:RETURN
1280 IF CURY=YMN THEN CURSOR CURX,YMX:RETURN:REM NEER
1290 CURSOR CURX,CURY-1:RETURN
1300 IF CURX=XMN THEN 1320:REM LINKS
1310 CURSOR CURX-1,CURY:IF CURX MOD 5=1 THEN 1310:RETURN
1320 IF CURY<YMX THEN CURSOR XMN,CURY+1:RETURN
1330 CURSOR XMN,YMN:RETURN
1340 IF CURX=XX THEN 1360:REM RECHTS
1350 CURSOR CURX+1,CURY:RETURN
1360 IF CURY>YMN THEN CURSOR XMN,CURY-1:RETURN
1370 CURSOR XMN,YMX:RETURN
1380 CURSOR X(K),CURY:RETURN
1390 XMN=44:XX=59:CURSOR K+43,CURY:GOTO 1010
1400 UIT=1:RETURN
1410 UIT=2:RETURN

1500 REM SH CURUP: VOLGENDE BLOK
1510 N=N+1:Y=20:K=1:GOSUB 20:NSP$=STR$(G):K=2:GOSUB 20:NBLK$=STR$(G)
1515 SP$=LEFT$(NSP$,LEN(NSP$)-2):BL$=LEFT$(NBLK$,LEN(NBLK$)-2)
1520 IF SP$=" 0" THEN CURSOR 33,3:PRINT "LAATSTE BLOK";:GOTO 1000
1530 CURSOR 5,22:PRINT "SPOOR:";SP$;" - BLOK:";BL$;" #";MID$(STR$(N),1,
LEN(STR$(N))-3);" ";
1540 GOTO 240

```



```

1800 REM MAAK PLAATSCODE K UIT CURSOR
1810 X=CURX:Y=CURY:IF X<44 THEN 1830
1820 K=X-43:RETURN
1830 K=0
1840 K=K+1:IF K>16 THEN 1850:IF X(K)<=X THEN 1840:K=K-1:RETURN
1850 K=16:RETURN

1900 REM ASCIL NAAR STRING
1910 IN=ASC(IN%)
1920 IN%=LEFT$(STR$(IN),LEN(STR$(IN))-2)
1950 RETURN

2000 REM LEES VIDEO-RAM /SCHRIJF BLOK
2005 CURSOR 33,3:PRINT "VIDEO-RAM UITLEZEN";
2010 CALLM OPEN4,OPEN%
2020 OP%="B-F 4 0":CALLM PRNT15,OP%
2030 FOR Y=20 TO 5 STEP -1:YD=(23-Y)*#86
2040 FOR K=1 TO 16:GOSUB 30:IF G<0 THEN 2100:IN%=CHR$(G):CALLM PUT4,IN%:NEXT:
NEXT
2080 OP%="U2 4 0"+5P%+BL%:CALLM PRNT15,OP%
2090 OP%="ID":CALLM PRNT15,OP%:GOTO 2110
2100 CURSOR 33,1:PRINT "LEESFOOT; GEEN ASCII":CALLM CLOSE4:RETURN
2110 CALLM CLOSE4:CURSOR 33,3:PRINT SPC(26);:RETURN

```

CROSS-REFERENCE

20:	1510,	1260:	1030,1230,
25:	1240,	1280:	1030,1230,
30:	2040,	1300:	1030,1230,
50:	30,	1310:	1310,
90:	10,1610,	1320:	1300,
200:	220,230,	1340:	1030,1230,
220:	200,210,	1360:	1340,
230:	210,	1390:	1220,1230,
240:	1540,1560,	1400:	1030,1230,
390:	370,	1410:	1030,1230,
1000:	1520,1550,	1500:	1030,1230,
1010:	1030,1042,1043,1044,1210,1390,	1550:	1030,1230,
1020:	1020,	1600:	1020,1220,
1043:	1042,	1800:	1010,1210,
1044:	1042,	1830:	1810,
1180:	1020,1030,	1840:	1840,
1210:	1010,1180,1230,1235,1242,1243,1244,	1950:	1840,
1220:	1220,		

sinds enige tijd speelt de vraag, hoe we verder moeten met de DAI. Het is wel duidelijk dat het aantal hobbyisten dat nu nog een spiksplinternieuwe DAI zal aanschaffen uiterst klein, zometer nul zal zijn. Tegenwoordig is voor hetzelfde geld meer functionaliteit te koop (vergelijk DAI-DOS maar eens met de ATARI 520ST of een nieuwe MSX-2 machine).

Het is weliswaar nog geen reden om onze ouwe trouwe DAI te verschroten, maar het risico dat de zaak langzaam maar zeker doodbloedt is wezenlijk. Toch biedt juist een club met hobbyisten, zoals de DAIGG de mogelijkheid om het leven van de DAI niet alleen te rekken, maar zelfs een onbezorgde en levendige oude dag te bezorgen. De aangewezen weg is om juist voor hobbyisten aantrekkelijke hardware en software te maken, en dat tegen een voor hobbyisten aantrekkelijke prijs (min of meer gratis dus). De daad bij het woord voegend heb ik daarom besloten om voor alle software die ik tot nu toe voor de DAI heb geschreven (en in de club-bibliotheek voorhanden) geen stuiver extra te vragen. U kunt vanaf nu bij Theo de volgende software tegen kostprijs (prijs van diskette of (MDCR-)cassette) bekomen:

- OTHELLO het bekende REVERSI of OTHELLO-spel met 4 spel-niveaus, en op een zeer overzichtelijk speelveld (naar een programma uit de HCC 61).
- SHIFT een grafisch schuif-spelletje (reeds gepubliceerd in DAITA 13)
- DRAWING een programma dat in interactie met de gebruiker lijnen trekt, blokken vult, cirkels tekent en tekst op het graphische scherm zet (zowel wit-op-zwart als zwart-op-wit)
- VLAGGEN een programma dat de vlaggen van een groot aantal landen tekent.
- ANAGRAM een woordspelletje, waarbij de DAI de letters door elkaar gooit, en U ze weer op de juiste volgorde moet zetten.
- RANGOON een grafisch kaartspel, waarvan U een beschrijving vindt in dit nummer van DAITA. Maar pas op, U kunt aan dit spel helemaal verslaafd raken !!

Ik hoop dat ik daarmee een voorzet heb gegeven die andere DAI-bezitters nog vele uren spel-plezier zal verschaffen, en natuurlijk ook dat het voorbeeld navolging zal vinden. Lang Leve de DAI.

Nog meer uitverkoop.

Zo mag ik het wel noemen. Henk Risson laat ons weten dat er toch nog een 60-tal 1541 floppy drive,s verkocht zijn. Allerdings lovenswaardig. Hij heeft er nog enkele over en die wil hij graag opruimen. Het is mogelijk deze als bouw pakket te betrekken. Informeer naar prijs en prestatie op telefoon num-

04401 49962

twee namen heeft geleid tot een meer exotische naam.

U heeft al een voorproefje gehad van het spel, op de voorpagina van DAITA-14 stonden namelijk twee plaatjes van een "tableau" zoals het op het scherm van de PC verschijnt tijdens het spel. In dit geval op een "echte" PC (een IBM-compatible dus) waarop ik het spel ondertussen ook heb "opgebracht". Hieronder volgt een gedeelte uit de handleiding van het spel (ook bijgeleverd in het programma RANINFO) om een idee te krijgen wat het programma doet.

Uitleg van het spel RANGOON.

RANGOON is een patience-spel dat wordt gespeeld met 52 kaarten. Na het schudden van de kaarten worden deze met de beeldzijde zichtbaar uitgelegd in vier rijen van elk dertien kaarten. Vervolgens worden de vier heren eruit gehaald en vooraan neergelegd; schoppen-heer komt aan de linkerkant van de bovenste rij, harten-heer op de rij eronder, ruiten-heer daar weer onder, en tenslotte klaver-heer op de onderste rij.

Nu kan het eigenlijke spel beginnen. Het doel is om de ideale situatie te bereiken die als volgt wordt gekenmerkt: op de bovenste rij bevinden zich alle schoppens, en wel in de volgorde: heer, vrouw, boer, tien, negen, acht, zeven, zes, vijf, vier, drie, twee, aas. Op de tweede rij bevinden zich de hartens, op de derde de ruitens, en op de onderste de klaveren.

Hebt U de hierboven vermelde situatie bereikt, dan bent U UIT !!

Om dit hoogste ideaal te bereiken zullen de kaarten verlegd moeten worden, aangezien de kans dat ze meteen goed liggen astronomisch klein is. Het verleggen van de kaarten is aan slechts één regel gebonden: een kaart mag slechts dan achter zijn voorganger gelegd worden als de plaats daarachter "vrij" is. Vier van zulke vrije plaatsen zijn ontstaan door het naar voren leggen van de vier heren. Het verleggen van een kaart vult een gat, maar tegelijkertijd ontstaat een nieuw gat op de plaats waar die kaart vandaan komt. Dat gat kan dan weer worden gevuld, enzovoort.

Een voorganger is de naast hogere kaart van dezelfde kleur, bijvoorbeeld: de voorganger van schoppen-acht is schoppen-negen, die van ruiten-tien is ruiten-boer, die van harten-aas is harten-twee. De heren zelf hebben geen voorganger; er zijn geen kaarten die een aas als voorganger hebben.

Het spel eindigt als alle vier de gaten of achter een aas of achter een ander gat liggen, zodat er geen enkele kaart verlegd kan worden. Hoewel de spelregels simpel zijn komt er toch nog aardig wat geluk en/of vaardigheid bij kijken om de ideale eindstand te bereiken. In de meeste gevallen zult U al eerder zijn blijven steken; een score van een op de honderd is echter al heel verdienstelijk !!

Om het geheel nog een extra dimensie te geven (tenslotte moet men het spel al heel aardig beheersen om tenminste een op de honderd keer UIT te komen) is een puntentelling ingevoerd. Voor elke kaart die op de ideale positie ligt wordt 1 punt gegeven, voor elke kaart die achter zijn voorganger ligt eveneens 1 punt. Bent U helemaal UIT, dan levert dat de maximale score van 100 punten op (52 punten voor de juiste plaats en 48 punten voor de juiste voorgangers). Zoals

- een serie van 5 spellen
- een serie van 20 spellen
- een serie van 100 spellen

Het spel op de DAI

Het spel leent zich uitermate goed om op de DAI gespeeld te worden. Het programma neemt een groot aantal van de taken van de speler over, op één punt na: het NADENKEN over de te volgen strategie moet de speler zelf doen. Het schudden, het uitleggen, het bijhouden van de punten (ook tijdens het spel) wordt door het programma verzorgd, verder zal het programma het verleggen van de kaarten overnemen indien er nog maar één geldige verlegging mogelijk is. Daarnaast is een drietal hulpfuncties ingebouwd om het de speler wat te vergemakkelijken:

- het ongedaan maken van de laatste verlegging.
- het laten zien welke kaarten verlegd kunnen worden.
- het laten zien hoe een gat gevormd zou kunnen worden.

Tijdens het spel wordt onder het tableau met de kaarten een viertal lijnen met tekst getoond. Hierin wordt de volgende informatie gegeven:

- rechter-kolom:
 - de score van het huidige spel:
 - de positie-score
 - de volgorde-score
 - de score (som van de bovenstaande twee)
 - de score van vijf spellen terug.
 - de totaal-score van (de laatste) 5 spellen
 - de totaal-score van (de laatste) 20 spellen
 - de totaal-score van 100 spellen
- midden-kolom:
 - de mogelijke verleggingen
 - ev. waarschuwings-boodschap
 - spelnummer
 - help-boodschap
- linker-kolom:
 - de scores van de voorgaande vier spellen

U kunt de score natuurlijk met de hand bijhouden, maar ook voor dit karweitje is er programmatuur op uw DAI beschikbaar. Het spelprogramma RANGOON gebruikt een kladblad om de huidige scores te laten zien, en om de nieuwe scores toe te voegen; op het kladblok is er ruimte voor 100 spellen. Tussen het spelen door wordt het kladblad als een INT-file onder de naam RANPAG.INT opgeslagen. Het kladblok zelf wordt beheerd door het programma RANADMI, van waaruit andere programma's voor andere deeltaken zorgen, zoals het programma RANREC voor het bijhouden van de persoonlijke records.

Welke programma's er nog meer zijn en wat ze doen, wordt U duidelijk gemaakt door het programma RANINFO dat onder andere ook de bovenstaande informatie

geen DAI-VC1541 interface maar willen toch graag een HSL bouwen, kunnen jullie niet de software publiceren volgt hier het complete software programma.

Deze listing is NIET hetzelfde als wat er in onze DAI - VC1541 DOS staat, maar een op zich zelf staand programma. Aan het einde van de listing staat "ENTRY FROM DAI POWER ON/RESET". Dit stukje wordt alleen gebruikt wanneer de interface of HSL beschikt over een hardware Power On initialisatie. Mocht U hier over beschikken, in b.v. een DCR, dan moet U dit stukje programma nog toevoegen.

Het schema voor de Power On initialisatie hardware zal ik ook publiceren.

Het is de bedoeling dat deze software in een Eprom komt welke op de "X" bus wordt geplaatst.

Bent U in het bezit van een DCR dan moet U dit programma combineren met Uw DCR software, assembleren en opnieuw in een Eprom programmeren.

Ik raad U aan hiervoor een andere Eprom te kopen en de oude niet te wissen totdat U er zeker van bent dat alles werkt.

Mocht U niet beschikken over een "X" bus interface print dan vindt U hierbij het schema hoe U er een kunt bouwen. Dit is geen groot probleem, het gaat op een stukje Veroboard van 7x5 cm.

Mede gelet op de feiten dat Eproms van het type 2764 en 27128 momenteel goedkoper zijn dan de 2716 en 2732, en de meeste programma's 8k byte of groter zijn, heb ik een gewijzigde HSL gemaakt.

Hiermede is het mogelijk zelf een keuze te maken uit drie Eprom type's afhankelijk van de lengte van het programma.

Hier zijn ook beperkingen aan verbonden n.l:

Een programma moet in een halve of hele Eprom passen.

U kunt maar twee Eproms gebruiken voor maximaal 4 programma's.

Gebruikt U twee 27128 dan kunt U hierin 4 programma's opbergen van elk 8k byte groot.

De situatie verandert natuurlijk wanneer U 27256 gebruikt.

Hierin kunt U een 32k byte programma opbergen, maar deze Eproms zijn momenteel nog duur.

Beschrijving schema.

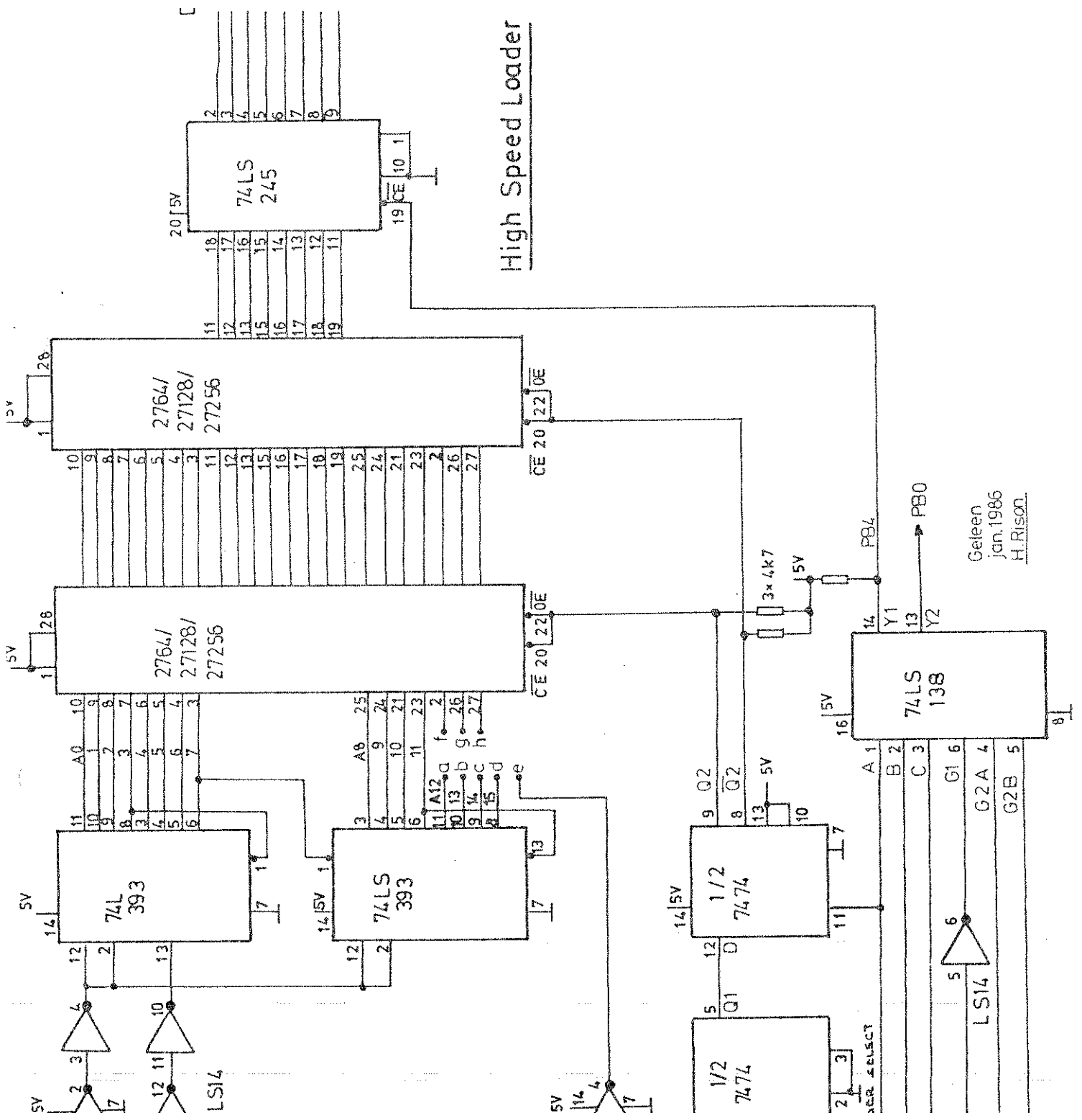
In deze versie is de prom van het type 82S23 komen te

Selectie Tabel.

Eprom type	Geheugen Inhoud	Door verbinding	Selectie commando
2764	2 x 8K	a - f	HSL en HSL2
2764	4 x 4K	e - f	HSL, HSL1, HSL2 en HSL3
27128	2 x 16K	a - f b - g	HSL en HSL2
27128	4 x 8K	a - f e - g	HSL, HSL1, HSL2 en HSL3
27256 27256	2 x 32K	a - f b - g c - h	HSL en HSL2
27256	4 x 16K	a - f b - g e - h	HSL, HSL1, HSL2 en HSL3

Ik hoop dat ik U hiermede van genoeg informatie heb voorzien om een HSL te bouwen.
Het programmeren van de Eproms moet U zelf verzorgen.
Denk aan de eerste bytes zoals beschreven in mijn vorige artikel.
De praktische realisatie van dit project laat ik aan U over.

Geleen, febr.1986
H.Rison



High Speed Loader

Geleen
jan. 1986
H. Risson

```

0000 @=0117 RDIFP EQU 117H ;flag set while running input
0000 @=0131 OTSW EQU 131H ;output switching
0000 @=0132 EFEPT EQU 132H ;encoding input pointer
0000 @=0134 EFECT EQU 134H ;encoded input count
0000 @=0135 EFSW EQU 135H ;encoded input switching
0000 @=01B0 FILES EQU 1B0H ;number following commands
0000 @=0296 INSW EQU 296H ;input switching
0000 @=0297 TABLE EQU 297H ;start address table
0000 @=0299 RSP EQU 299H ;rescued stackpointer
0000 ; during comm.execution
0000 @=02C4 KBRFL EQU 2C4H
0000 @=02E3 DINC EQU 2E3H
0000 ; DAI ROM ROUTINES
0000 ; *****
000 @=C14D EXIT EQU 0C14DH ;POP H,D,B,PSW,RETURN
0000 @=DD45 EXIT1 EQU 0DD45H ;POP H,PSW,CMC,RETURN
0000 @=C88F ENDCOM EQU 0C88FH ;jump if spec. action
0000 @=CA34 LOOKC EQU 0CA34H ;find instruction in table
0000 @=D1E2 INO EQU 0D1E2H ;if input from keyboard
0000 @=D695 MPT31 EQU 0D695H ;output one character
0000 @=DA0B SYNERR EQU 0DA0BH ;run syntax error
0000 @=DAFF PMSGR EQU 0DAFFH ;print message
0000 @=DD5E CRLF EQU 0DD5EH ;print carr.return linefeed
0000 @=DDB4 INSER EQU 0DDB4H ;input from RS232
0000 @=DDD2 GIGNB EQU 0DDD2H ;get char.from line neglect
0000 ; ;TAB and SPACE
0000 @=DE0D NUMER EQU 0DE0DH ;check for number
0000 ; GIC CONTROLLER FPI (8255)
0000 ; *****
0000 @=FE00 GICPA EQU 0FE00H ;I/O port-A
0000 @=FE01 GICPB EQU 0FE01H ;I/O port-B
0000 @=FE02 GICPC EQU 0FE02H ;I/O port-C
0000 @=FE03 GICCMD EQU 0FE03H ;command word 8255 (GIC)
000 ;
0000 ; ORG 0F000H
F000 ; *****
F000 ; ENTRY POINTS
F000 ; *****
F000 C321F0 REMS JMP REM ;Execute REM statement
F003 C329F1 CRDL JMP HSL ;Command read data from EPROM
F006 ; DS 1BH
F021 ; EXECUTE REM STATEMENT
F021 ; *****
F021 F5 REM PUSH PSW
F022 C5 PUSH B
F023 D5 PUSH D
F024 E5 PUSH H
F025 210A00 IXT H 0AH

```



```

F035 323401 STA EFECT ;set EFECT
F038 60 MOV H,B
F039 69 MOV L,C
F03A 223201 SHLD EFEPT ;BC in EFEPT
F03D 3E01 MVI A 1H
F03F 323501 STA EFSW ;input from string
F042 4F MOV C,A
F043 CDA7F0 CALL FIND ;execute command
F046 AF XRA A
F047 323501 STA EFSW ;input from keyboard
F04A 329A02 STA 29AH
F04D 0D DCR C
F04E CA0BDA JZ SYNERR ;then "SYNTAX ERROR"
F051 C34DC1 JMP EXIT ;pop all:ret
F054 ;REPLACED DATA INPUT ROUTINE (DINC)
F054 ;*****
F054 E5 RDINC PUSH H
F055 211701 LXI H RDIFF ;(RDIFF)=#FF
F058 7E MOV A,M ;get flag running inputs
F059 23 INX H
F05A B6 ORA M ;compare with flag running
programs
F05B 6F MOV L,A ;00 when both 0, else FF
F05C 3A3101 LDA OTSW ;get output direction
F05F E6FE ANI OFEH ;mask bit 0
F061 B5 ORA L ;result is 0,2 or FF
F062 E1 POP H
F063 C280F0 JNZ CHID ;when running something
F066 CD80F0 CALL CHID ;get inputs
F069 F5 PUSH PSW
F06A FE0D CPI ODH ; "CR" ?
F06C CA8BF0 JZ IFCR
F06F FE09 CPI 9H ; "TAB" ?
F071 C27EF0 JNZ RDINC1 ;JMP else
F074 3E0C MVI A OCH
F076 CD95D6 CALL MPT31 ;clear screen
F079 3E2A MVI A 2AH
F07B CD95D6 CALL MPT31 ;print "*"
F07E F1 RDINC1 POP PSW
F07F C9 RET
F080 ;CHECK INPUT DIRECTION
F080 ;~~~~~
F080 3A9602 CHID LDA INSW ;get DINC flag
F083 E601 ANI 1H
F085 CAE2D1 JZ INO ;check for new key pressed?
F088 C3E302 JMP DINC ;get input from RS232
F08B ;INPUT IS CARRIAGE RETURN
F08B ;~~~~~

```

```

FOA7 2A9702 FIND LHL D TABLE ;get start addr. cmd. table
FOAA 1E01 FIND1 MVI E 1H
FOAC 7C MOV A,H
FOAD B5 ORA L
FOAE C8 RZ ;if zero end of cmd. table
FOAF C5 PUSH B
FOB0 CD34CA CALL LOOKC ;find comm. in table
FOB3 DABEF0 JC FIND2 ;when found
FOB6 C1 POP B ;else
FOB7 7E MOV A,M ; (
FOB8 23 INX H ;addr. in table where
FOB9 66 MOV H,M ;instr.string can be found
FOBA 6F MOV L,A ; )
FOBB C3AAFO JMP FIND1 ;check next one
FOBE 5E FIND2 MOV E,M ; (get addr.cmd. after
FOBF 23 INX H ; string in DE
FOC0 56 MOV D,M ; )
FOC1 CDCDF0 CALL NUCCOM ;check for more cmds or number
FOC4 79 MOV A,C
FOC5 C1 POP B
FOC6 4F MOV C,A
FOC7 21A7FO LXI H FIND
FOCA E5 PUSH H ;return addr.FIND on stack
FOCB EB XCHG ;get addr.DL,command in HL
FOCC E9 PCHL ;perform command
FOCD ; CHECK FOR NUMBER AND/OR MORE COMMANDS
FOCD ; ~~~~~
FOCD CDD2DD NUCCOM CALL GIGNB ;get char. from line,neglect
TAB/SPACE
FOD0 CD0DDE CALL NUMER ;check if char.is number
FOD3 DAD9FO JC NUCCOM1
FOD6 3E2F MVI A 2FH
FOD8 0D DCR C
FOD9 0C NUCCOM1 INR C ;points to next char.
FODA D630 SUI 30H
FODC 32B001 STA FILES ;store HEX val. of nr in FILES
FODF CDD2DD NUCCOM2 CALL GIGNB ;get char.neglect tab/space
FOE2 0C INR C ;points to next char.
FOE3 FE3A CPI 3AH
FOE5 C8 RZ ;return when ":"
FOE6 FE0D CPI 0DH
FOE8 C8 RZ ;return when "CR"
FOE9 3AB001 LDA FILES ;get FILES
FOEC B7 ORA A
FOED F2DFF0 JP NUCCOM2 ;get next char.until FILES=FF
FOF0 0E01 MVI C 1H
FOF2 117EFO LXI D RDINC1 ;alternative return addr.FOCC
FOF5 C9 RET

```

```

F0FF 217DF1 LXI H COUNTAB ;start addr. Count. table
F102 229702 SHLD TABLE ;store it in TABLE
F105 219602 LXI H INSW ;addr. DINC-flag
F108 3E02 MVI A 2H
F10A B6 DRA M
F10B 77 MOV M,A ;set flag for(0296) DR-02
F10C AF XRA A
F10D 32C402 STA KBRFL ;reset KBRL
F110 2154F0 LXI H RDINC ; [
F113 22E102 SHLD 2E1H ;new DINC into address
F116 3EC3 MVI A 0C3H ; 02E0-02E2
F118 32E002 STA 2E0H ; ]
F11B 21B4DD LXI H INSR ; [load "JUMP DDB4" (INSER)
F11E 22E402 SHLD 2E4H ; into 02E3-02E5
F121 3EC3 MVI A 0C3H ; input from RS232
F123 32E302 STA DINC ; ]
F126 C34DC1 JMP EXIT ;POP H,D,B,PSW,RETURN
F129 ;READ DATA FROM HIGHSPEED DATA LOADER
F129 ;*****
F129 F5 HSL PUSH PSW
F12A D5 PUSH D
F12B E5 PUSH H
F12C 3AB001 LDA FILES ;EPROM-bank number
F12F B7 DRA A
F130 F237F1 JP HSL10 ;If number give
F133 3C INR A
F134 32B001 STA FILES ;Else default is HSL0
F137 FE04 HSL10 CPI 4H
F139 F20BDA JP SYNERR ;Syntax error if nr >4
F13C CDA6F1 CALL SETGIC ;Init GIC
F13F 3AB001 LDA FILES ;Get EPROM nr
F142 07 RLC
F143 07 RLC ;Into hinibble: PB2,3
F144 C610 ADI 10H ;Add HSL select: PB4=1
F146 3201FE STA GICPB ;Address EPROM
F149 3E01 MVI A 1H ;Reset HSL :PC0=1
F14B 3203FE STA GICCMD
F14E 3D DCR A
F14F 3203FE STA GICCMD ;Cancel reset: PC0=0
F152 ;LOAD MEM.ADDR. AND PRGRM LENGHT IF EPROM MOUNTED
F152 ;~~~~~
F152 CD7FF1 CALL HSL30 ;move character into A reg.
F155 FEFD CPI OFDH ;check for empty EPROM, wrong
F157 C28FF1 JNZ HSL40 ;coded and/or EPROM bank
F15A ;get startaddress + length
F15A CD7FF1 CALL HSL30
F15D 6F MOV L,A ;Lo-byte start addr.
F15E CD7FF1 CALL HSL30

```

```

F175          ;close HSLX
F175          ;~~~~~
F175 210000   LXI H      OH
F178 2201FE   SHLD      GICPB      ;reset addr.and count bits
F17B E1       POP H
F17C D1       POP D
F17D F1       POP PSW
F17E C9       RET              ;return to monitor prgrm.
F17F          ;read byte from HSL
F17F          ;~~~~~
F17F 3E03 HSL30 MVI A      3H
F181 3203FE   STA      GICCMD      ;Set clock: PC1=1
F184 3A00FE   LDA      GICPA      ;Get byte
F187 F5       PUSH PSW
   188 3E02   MVI A      2H
F18A 3203FE   STA      GICCMD      ;clear clock:PC1=0
F18D F1       POP PSW
F18E C9       RET
F18F          ;ERROR ROUTINE
F18F          ;*****
F18F 210000 HSL40 LXI H      OH
F192 2201FE   SHLD      GICPB      ;Disable HSL:PB+PC=0
F195 CDFFDA   CALL     PMSGR      ;print error message
F198 C7DC     DW       ODCC7H    ;"NOT AVAILABLE"
F19A C35EDD   JMP      CRLF      ;carr.return linefeed
F19D          ;TABLE WITH COMMANDS
F19D          ;*****
F19D 03 COMTAB DB          3H
F19E 48534C DB          'HSL'    ; command HSL
F1A1 29F1     DW          HSL
F1A3 00       NOP
F1A4 00       NOP
F1A5 00       NOP
   1A6          ;set GIC for HSL
F1A6          ;~~~~~
F1A6 3E98 SETGIC MVI A      98H
F1A8 3203FE   STA      GICCMD      ;Select GIC-mode
F1AB 326100   STA      MEMGIC     ;Keep memory
F1AE C9       RET
F1AF          ;ENTRY FROM DAI POWER ON/RESET
F1AF          ;*****
F1AF          ;ORG OF2F2H
F1AF          ;JMP INIT ;init Highspeed Data Loader
F1AF          END

```

adres lijnen, data lijnen en diverse controle signalen uitkomen.

Deze connector wordt de "X" bus connector genoemd en is zeer geschikt voor uitbreidingsmogelijkheden.

Een van deze mogelijkheden is een Eprom kaart waarbij de vrije adres ruimte van #F000 tot #F7FF gebruikt wordt voor een besturings programma b.v. voor DCR of High Speed Loader.

Door middel van z.g. bank switching kan men hier ook grotere bestanden in onder brengen.

Wij zullen echter alleen het schema voor de besturing van een Eprom van het type 2716, welke 2Kbyte kan bevatten, behandelen.

Beschrijving schema.

Het schema is zeer eenvoudig. Het bestaat uit twee IC's n.l de Eprom 2716 welke aangesloten wordt op de adres lijnen A0 t/m A10, de data lijnen D0 t/m D7 en \overline{OE} .

De selectie van de Eprom vindt plaats d.m.v een TTL bouwsteen van het type 74LS151.

Als de signalen A12 t/m A15 "H" zijn (#FO) en A11 is "L" gaat de uitgang (het signaal W) op pin 6 laag. Dit is het \overline{CE} signaal voor de Eprom en maakt deze actief (Mits \overline{OE} actief is).

De adres counter kan nu het adres bereik van 0 tot 7FF doorlopen. Dit is tot alle bits van A0 t/m A10 "H" zijn. Gedurende deze tijd kan men de 8 data bits via D0 t/m D7 uitlezen.

Wordt nu echter A11 "H", adres #F800, dan schakelt de uitgang van de 74LS151 om op "H" omdat de ingangs signalen niet meer aan de juiste voorwaarden voldoen en maakt de Eprom inactive.

De constructie van deze kaart is niet moeilijk. De connector is meestal in een goede elektronika zaak verkrijgbaar evenals een stukje gaatjesboard twee IC voeten en een condensator.

Als U klaar bent met de bouw meet dan eerst alle signalen van de connector naar de IC voeten door en zorg ervoor dat er nergens sluiting is.

Een voortijdig einde van Uw DAI is het laatste wat ik U zou wensen.

Geleen, jan 1986

H.Rison

```

9      REM **** SCHUIF HEAP OP, MAAK RUIMTE VOOR MONITOR ***
10     CLEAR #2600-PEEK(#29B)-(PEEK(#29C) SHL 8):POKE #29B,0:POKE #29C,6:POKE
      #29D,0:POKE #29E,#20
20     CLEAR #2000
30     IF PEEK(#400)=#C3 THEN 100
40     DIM BOOT(8,0):LOADA BOOT "BOOT":CH#="FDD-MONITOR"
50     BOOT=VARPTR(BOOT(0,0)):CALLM BOOT,CH#

99     REM *****
100    REM *** HOOFDPROGRAMMA
101    REM *****
110    GOSUB 1000:REM INITIALISEREN
120    GOSUB 1000:REM LEES DIRECTORY
130    M=0:X=0:Y=21
140    GOSUB 1300:REM DISPLAY
150    GOSUB 1500:REM KEY&CURSOR
160    IF G=72 THEN PRINT CHR$(12):GOSUB 1200:GOTO 140:REM HERSTEL
170    IF G=83 THEN GOSUB 2500:GOTO 140:REM SCHRIJF NIEUWE DIR
180    IF G=78 THEN GOSUB 2800:GOTO 140:REM NAAM
190    IF G=87 THEN GOSUB 2000:GOTO 140:REM WISSEL
200    IF G=86 THEN GOSUB 3000:GOTO 140:REM VEEG
210    IF G=65 THEN GOSUB 2200:GOTO 140:REM SORTEER ALFABETISCH
220    IF G=85 THEN PRINT CHR$(12):END:REM UIT
230    IF G=20 OR G=21 THEN 140
290    GOTO 150

997    REM
998    REM *****
999    REM
1000   REM *** LEES DIRECTORY IN DIR#
1010   PRINT CHR$(12);:CURSOR 10,20:INPUT "breng het juiste schijfje in
      <RETURN>";CH#
1020   CURSOR 10,16:PRINT "DIRECTORY LEZEN":CURSOR 10,12:PRINT "SPOOR 18 -
      BLOK";
1030   OPEN#=#":CALLM OPEN4,OPEN#
1040   OP#="U1 4 0 18 0":CALLM PRNT15,OP#
1050   POS144#="B-P 4 144":CALLM PRNT15,POS144#
1060   DIR$(0)=" "
1070   CALLM GET4,IN#:IF IN#(>)CHR#(#A0) THEN DIR$(0)=DIR$(0)+IN#:GOTO 1070
1080   BUF=BUFORG
1090   SP#=" 1"
1100   OP#="U1 4 0 18 "+SP#:CALLM PRNT15,OP#
1110   POS1#="B-P 4 1":CALLM PRNT15,POS1#
1120   CALLM GET4,IN#:SP#=STR$(ASC(IN#)):SP#=LEFT$(SP#,LEN(SP#)-2)
1130   POS0#="B-P 4 0":CALLM PRNT15,POS0#
1140   CURSOR 26,12:PRINT SP#;" ";
1150   FOR I=0 TO 255:CALLM GET4,IN#:POKE BUF+I,ASC(IN#):NEXT
1160   IF PEEK(BUF)(>0) THEN BUF=BUF+256:GOTO 1100

```

```

1220 CURSOR 17,8:PRINT N;
1230 CH$="":FOR I=2 TO 31:CH=PEEK(BUF+I):IF CH=#AO THEN CH=#1A
1240 CH$=CH$+CHR$(CH):NEXT
1250 IF PEEK(BUF+2) (80 THEN 1260:DIR$(N+1)=CH$:N=N+1
1260 BUF=BUF+#20
1270 IF PEEK(BUF) (>#4B THEN 1220
1280 CALLM CLOSE4
1290 RETURN

1300 REM **** DISPLAY DIR$
1310 PRINT CHR$(12);
1320 PRINT TAB(12);DIR$(0,0):PRINT
1330 FOR I=M+1 TO M+20
1340 1 A=I:GOSUB 1900:IF F=1 THEN PRINT A$;CH$;
1350 1 A=I+20:GOSUB 1900:IF F=1 THEN PRINT TAB(20);A$;CH$;
1360 1 A=I+40:GOSUB 1900:IF F=1 THEN PRINT TAB(40);A$;CH$;
1370 1 PRINT
1380 NEXT
1390 CURSOR 0,0:PRINT "(SH)Cur ";CHR$(10);" Alf ";CHR$(10);" Herst ";
CHR$(10);
1400 PRINT " Naam ";CHR$(10);" Schr.DIR ";CHR$(10);" Uit ";CHR$(10);" Veeg ";
CHR$(10);" Wis1";
1410 RETURN

1500 REM **** CURSORROUTINE
1510 POKE #74,1:POKE #75,#B9:CURSOR X,Y
1520 G=GETC:IF G<16 THEN 1520
1530 ON G-15 GOTO 1550,1570,1590,1610,1630,1650,1670,1670
1540 GOTO 1670
1550 Y=Y+1:IF Y>21 THEN Y=2
1560 CURSOR X,Y:GOTO 1510
1570 Y=Y-1:IF Y<2 THEN Y=21
1580 CURSOR X,Y:GOTO 1510
1590 X=X-20:IF X<0 THEN X=40
1600 CURSOR X,Y:GOTO 1510
1610 X=X+20:IF X>59 THEN X=0
1620 CURSOR X,Y:GOTO 1510
1630 M=M-60:IF M<0 THEN M=N-(N MOD 60)
1640 GOTO 1670
1650 M=M+60:IF M>N THEN M=0
1660 GOTO 1670
1670 POKE #74,1:POKE #75,#5F
1680 RETURN

1900 REM **** CHECK DIR$(A) EN MAAK A$
1910 F=1:IF A>N THEN F=0:GOTO 1950
1920 CH$=DIR$(A):IF LEN(CH$)=0 THEN F=0:GOTO 1950
1930 CH$=MID$(CH$,3,14)

```

```
2040 H%=DIR%(NO)
2050 FOR I=NO TO N-1:DIR%(I)=DIR%(I+1):NEXT
2060 FOR I=N-1 TO NR STEP -1:DIR%(I+1)=DIR%(I):NEXT
2070 DIR%(NR)=H%
2080 RETURN
```

```
2200 REM **** SORTEER ALFABETISCH
2210 CURSOR 0,0:PRINT SPC(59);:CURSOR 5,0:PRINT "SORTEREN: ";N;
2220 FOR A=1 TO N-1
2230 1 C=A
2240 1 FOR B=A+1 TO N
2250 2 IF MID%(DIR%(B),3,16)<MID%(DIR%(C),3,16) THEN C=B
2260 1 NEXT B
2270 1 H%=DIR%(A):DIR%(A)=DIR%(C):DIR%(C)=H%
2280 1 CURSOR 20,0:PRINT A;
2290 NEXT A
2300 RETURN
```

```
2500 REM **** SCHRIJF DIRECTORY
2510 CALLM OPEN4,OPEN%
2520 K=0
2530 CURSOR 0,0:PRINT SPC(59);:CURSOR 5,0:PRINT "SCHRIJF FILE #";
2540 DATA 4," 1",7," 4",10," 7",13," 10",16," 13",2," 16",5," 2"
2550 DATA 8," 5",11," 8",14," 11",17," 14",3," 17",6," 3",9," 6",12," 9"
2560 DATA 15," 12",255," 15",0
2570 CALLM PRNT15,P050%
2580 CALLM PUT4,CH18%:READ SP:SP%=CHR%(SP):CALLM PUT4,SP%
2590 K=K+1:IF K>N THEN FOR J=0 TO 29:CALLM PUT4,NUL%:NEXT:GOTO 2620
2600 FOR J=0 TO 29:CH%=MID%(DIR%(K),J,1):IF CH%=CHR%(#1A) THEN 2615:CALLM
PUT4,CH%:NEXT:GOTO 2620
2615 CH%=CHR%(#A0):CALLM PUT4,CH%:NEXT
2620 IF K MOD 8=0 THEN 2660
2630 CURSOR 20,0:PRINT K;
2640 CALLM PUT4,NUL%:CALLM PUT4,NUL%
2650 GOTO 2590
2660 IF K>=N THEN CALLM PUT4,NUL%:CALLM PUT4,FF%
2670 READ SP%
2680 OP%="U2 4 0 18"+SP%:CALLM PRNT15,OP%
2690 IF K<N THEN 2570
2700 CALLM PRNT15,OF10%
2710 RETURN
```

```
2800 REM **** NAAM VERANDEREN
2810 NO=M+X+(22-Y):PRINT NUL%;
2820 CURSOR 0,0:PRINT SPC(59);
2830 CURSOR 0,0:PRINT MID%(DIR%(NO),3,16);"wordt: ";
2840 INPUT NI%:IF NI%="" THEN 2890
```



```

3030 CURSOR 0,0:PRINT "moet ";MID$(DIR$(NO),3,16);
3040 INPUT " geveegd worden (J/N)";CH$
3050 IF CH$="" THEN 3080:IF LEFT$(CH$,1)(">")="j" AND CH$(">")="J" THEN 3080
3070 FOR I=NO TO N-1:DIR$(I)=DIR$(I+1,0):NEXT:N=N-1
3080 RETURN

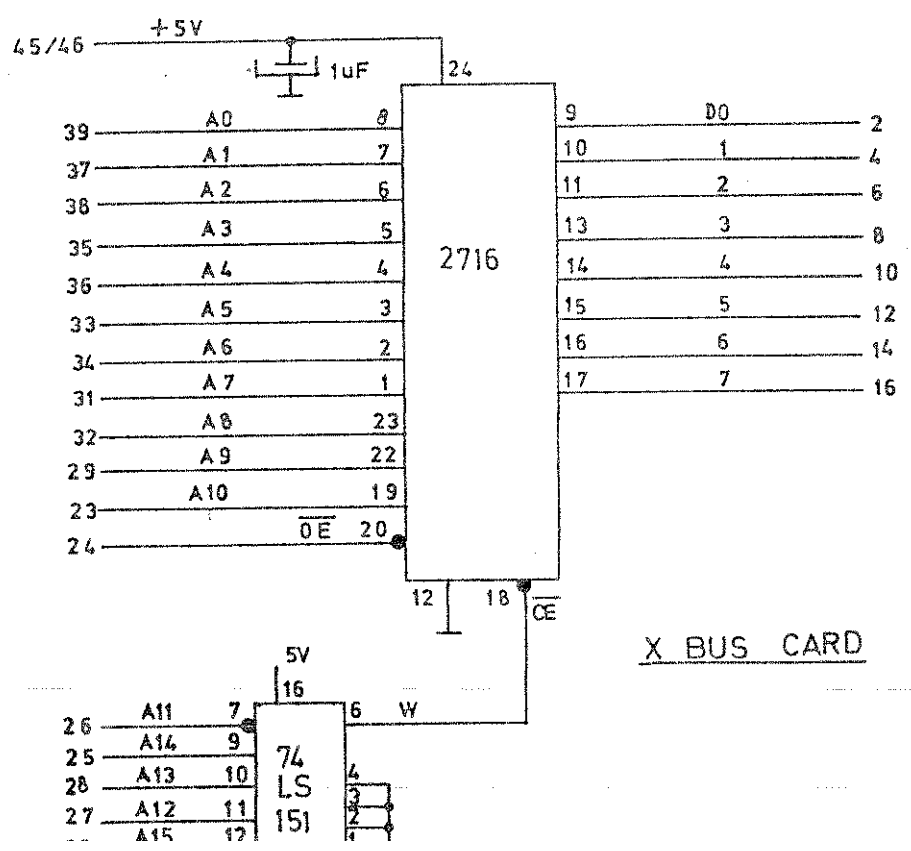
```

```

10000 REM *** INITIALISEREN
10010 DIM DIR$(144)
10020 OPEN4=#403:PRNT15=#40C:PUT4=#42D:GET4=#412:CLOSE4=#424:ERROR=#F2E8
10030 COLOR T B 0 15 5
10040 IN$=" ":NUL$=CHR$(0):OFIO$="IO":CH18$=CHR$(18):FF$=CHR$(#FF)
10060 BUFORG=#4000
10090 RETURN

```

Behoort bij het verhaal van de x-bus op blz. 20.



X BUS CARD

goed liep, moesten er van tevoren wat handelingen gedaan worden.

Er moest een MOVE gedaan worden in UT en het aantal regels enkelkolom moest aangevuld worden met CR's tot 2 maal het aantal regels per pagina dubbelkolom. Ook in het basic programma kon men (hoefde niet) nog een pointer aanpassen. Na een "RUN" doet het programma het dan ook prima, met dien verstande dat de laatste pagina er niet erg elegant uitziet.

Als er b.v. voor de laatste pagina nog 90 echte regels (dus geen CR's) overblijven zet het programma 60 regels in de eerste kolom en 30 in de tweede. Alles bij elkaar een leuke aanleiding om een programma te schrijven (waarbij de prima routine van Frank druyff natuurlijk onontbeerlijk is) dat al het bovenstaande automatisch en elegant doet. Met elegant bedoel ik dan, dat ook de regels op laatste pagina netjes verdeeld worden, dus bij b.v 90 regels enkelkolom 45 regels dubbelkolom. Met als resultaat het hierbij afgedrukte programma.

De regels 110 t/m 210 is de reeds eerder in DAINamic gepubliceerde routine met hier en daar een verandering. Deze subroutine wordt eigenlijk 1 of 2 maal aangeroepen.

De eerste keer wordt het begin van de verschoven text als begin meegegeven en als eind het adres aan het eind van de hele pagina's. Dit alles alleen als er hele pagina zijn, want anders wordt deze eerste aanroep overgeslagen. Deze eerste eventuele aanroep wordt gedaan in regel 370.

De tweede keer is het begin en het einde het beginadres en eindadres van de text op de laatste pagina. Deze tweede aanroep gebeurt in regel 230 . waarbij natuurlijk ook het aan-

regel 590 en is als volgt:

```
ORG      0B300H
PUSH B
LXI B    0000H   SAVE BASIC POINTER
LXI D    0000H   NIEUW ADDRES
LXI D    0000H   BEGIN TE VERSCHUIVEN
                BLOK
LXI H    0000H   EIND TE VERSCHUIVEN
                BLOK
CALL     0DE4FH  MOVE BLOK
POP B
RET
```

De regels 240 en 250 zorgen ervoor dat de hele text zoveel verplaats wordt, dat in het begin een vrije regels ontstaat om de besturingstekens voor de printer op te zetten. #1B en #4D is voor mijn printer "print ELITE" en #F betekent "condensed". Het geheel afgesloten met een CR (#D).

Hier moet u dus uw eigen controls invoeren en niet vergeten het einde van de text evenzoveel te verplaatsen. In mijn geval ST=ST+4.

In de regels 260 en 270 wordt het totaal aantal regels (T) in de text bepaald.

De variabele F in de regels 270 en 280 geeft aan na hoeveel regels er een CR moet komen te staan om ook de rechter-kolom in de dubbelkolom-text op de eerste regel van de eerste pagina blanco te houden. Het textgedeelte na deze plek wordt daarom eventueel 1 byte gEMOVED en een CR wordt tussengevoegd.

Als de text nog geen hele pagina dubbelkolom is, heeft F natuurlijk een andere waarde (regel 270) dan wanneer er wel hele pagina(s) zijn (regel 280). De bepaling waar deze CR moet komen te staan en de MOVE gebeurt door de subroutine opde regels 460 - 510.

Regel 290 kijkt nog of de text een on-

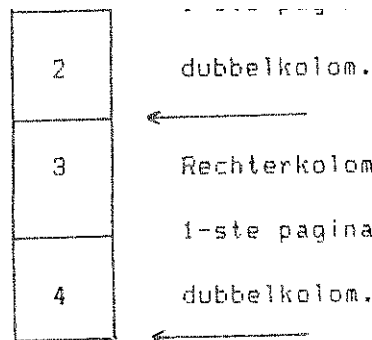
2. Hoeveel regels zijn er op de laatste pagina?
 (regel 310 : $L1=(T \text{ MOD } P1)/2$)

Indien er meerdere pagina's zijn, wordt in de regels 330 t/m 360 het eindadres na het totaal aantal hele pagina's bepaald (E) en het beginadres van de laatste pagina (E1).

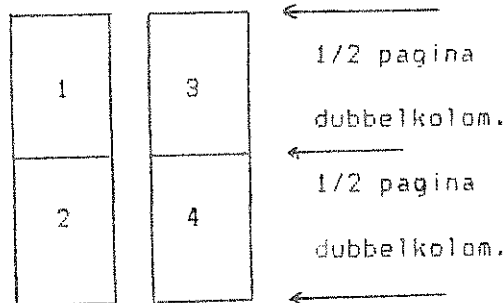
Met dit programmaatje heb ik grote test-texten "gedubbeld". Een stuk text van b.v. 23,8 Kb neemt na dubbelen meer ruimte in dan in buffer 1 voorhanden is. Als men na het "dubbelen" dan ook FWP weer opstart, staat de FWP byte-teller dan ook ergens in 64- a 65000. Deze uit buffer 1 in buffer 2 overlopende text kan men echter prima printen.

Men kan deze lange dubbelkolom-text ook SAVEN, maar dit heeft geen zin, omdat deze door de te grote lengte niet meer te LOADEN is:

Een ding dient nog besproken te worden. Door de toegevoegde tussen-spacities neemt een dubbelkolom-text meer ruimte in dan de oorspronkelijke hoeveelheid enkelkolom. Theoretisch is het dus mogelijk, dat als men de enkelkolom niet voldoende geMOVED heeft omdat er veel onvolledige zinnen inzitten, dat dan de dubbelkolom-text de enkelkolom-text inhaalt en overschrijft, met alle onprettige gevolgen vandien. Om die mogelijkheid tot een minimum te beperken, worden telkens als een 1/2 pagina dubbelkolom is op-



gaat over in



Als er een halve pagina is verwerkt, dan zijn de blokken 1 en 3 verbruikt en blok 2 wordt dan tegen blok 4 aangeschoven. Dit gebeurt in regel 190 en in subroutine in regel 520. De vlag VLI zorgt ervoor dat dit maar een maal per pagina gebeurt.

Verder wordt er in het programma gebruik gemaakt van de TIMER op de adressen #1BE en #1BF (regel 100 en 380 t/m 400). Na afloop wordt dan ook verteld hoelang het "dubbelkolommen" heeft geduurd. Een text van 8 a 9 kB duurt ca 2,5 minuten; bij een grote text kan dit oplopen tot 5 a 6 minuten.

PROGRAMMA :

```

100 CLEAR 200:GOSUB 530:P1=2*P:VL1=0:A=1:POKE #1BE,#FF:POKE #1BF,#FF:GOTO 240
110 R=0:BR=BL-1
120 BR=BR+1:IF PEEK(BR)<>13 THEN 120:R=R+1:IF R<P THEN 120:BR=BR+1:TUS=BR
130 BC=PEEK(BL):BL=BL+1:IF BC=13 THEN 150
140 POKE D,BC:D=D+1:A=A+1:GOTO 130

```

```

280 F=P:GOSUB 460
290 IF T MOD 2<>0 THEN GOSUB 450
300 EEN=NW:TW=D:DR=ST:GOSUB 420:ST=ST+OFS:ED=ST
310 VL=0:X=INT(T/P1):L1=(T MOD P1)/2
320 IF X=0 THEN E1=NW:GOSUB 230:GOTO 380
330 T1=0:L3=L1*2+1:K=ST
340 IF PEEK(K)=13 THEN T1=T1+1
350 IF T1=L3 THEN E=K+1:E1=E:BL=NW:GOTO 370
360 K=K-1:GOTO 340
370 GOSUB 110
380 TD!="#FFFF-(256*PEEK(#1BF)+PEEK(#1BE)):TI!=TD!/3000
390 MIN=INT(TI!):SEC!=FRAC(TI!)*60:SEC!=INT(10*SEC!)/10
400 PRINT "DE TIJD IS":MIN:" MINUTEN EN":SEC!:" SECONDEN"
410 POKE D,0:D=D+1:POKE #A4,D IAND #FF:POKE #A5,D SHR 8:END
420 PK=EEN:Z=0:GOSUB 440:PK=TW:Z=3:GOSUB 440
430 PK=DR:Z=6:GOSUB 440:CALLM #B300:RETURN
440 POKE #B302+Z,PK IAND #FF:POKE #B303+Z,PK SHR 8:RETURN
450 POKE ST,13:ST=ST+1:T=T+1:RETURN
460 T1=0:FOR K=D TO ST:IF PEEK(K)=13 THEN T1=T1+1
470 IF T1=F THEN GOSUB 490:RETURN
480 NEXT
490 IF PEEK(K+1)=13 THEN RETURN
500 EEN=K+2:TW=K+1:DR=ST:GOSUB 420:T=T+1:ST=ST+1
510 POKE K+1,13:RETURN
520 * EEN=BR+BL-TUS-1:TW=BL:DR=TUS:GOSUB 420:VL1=1:BL=EEN:RETURN
530 FOR K=0 TO 14:READ P:POKE #B300+K,P:NEXT
540 ST=PEEK(#A4)+256*PEEK(#A5)-1:D=#3000:PRINT CHR$(12)
550 OFS=#1000:NW=#3000+OFS
560 INPUT "Hoeveel regels dubbelkolom per pagina ":P:PRINT :T=0
570 INPUT "Hoelang is een regel enkelkolom maximaal ":EK:PRINT
580 INPUT "Hoeveel spaties wil je tussen de kolommen ":SP:PRINT :RETURN
590 DATA #C5,1,0,0,#11,0,0,#21,0,0,#CD,#4F,#DE,#C1,#C9
600 REM FWP DUBBELKOLOM AUTOMATISCH V.14
610 REM DATUM : 18 MEI 1986

```

GEBRUIK :

Als men in FWP een text heeft ingetypt, getrimd op de gewenste lengte en vervolgens heeft uitgevuld, gaat men via naar basic. Na IMP INT laadt men het programma en RUNT het.

TOM VAN ES - EVERTSESTRAAT 1 - 3601 JH - MAARSSEN.

TIP :

Wist u dat als men bij het printen met FWP het pagina-nummeren gebruikt, dat dan de plaats waar het pagina-nummer komt te staan, afhankelijk van de instelling van het aantal karakters per regel in het default-menu ?

```

30 IF F>Z2 THEN F=Z2
40 IF F<Z1 THEN F=Z1
50 RETURN
100 Y=W0:X=U0:GOSUB 20
110 W4=52*(W0-W1):X=54*(U0-U1)-C1*W4+R1:Y=53*(F-Z1)-51*W4+R2:W=1.0:IF P8=
0.0 THEN RETURN
120 W6=Y:U6=X:D=1.0:Y0=-51*W4+R2-1.0:Y9=56:IF X>59 THEN Y9=55*X+57
130 IF Y0<Y9 THEN RETURN
140 V2=(Y0-Y9)/EA:FOR K=0.0 TO EA:Y8=Y0-K*V2
150 1 Y=(R2-Y8)/51:X=(U6+C1*Y-R1)/54+U1:Y=Y/52+W1:GOSUB 20:F=(F-Z1)*53+Y8:IF
1 F<W6 THEN W=E0
160 1 IF F>=W6 THEN D=E0
170 1 IF W+D=E0 THEN 200
180 NEXT K
190 W=1.0
200 Y=W6:X=U6:RETURN
210 GOSUB 100:IF W=V9 THEN 330
220 IF T0=E2 THEN 280
230 W9=W0:W8=W9-W3
240 IF 52*(W9-W8)<EZ THEN 320
250 W0=(W8+W9)/E2:GOSUB 100:IF W<>V9 THEN W9=W0
260 IF W=V9 THEN W8=W0
270 GOTO 240
280 U9=U0:U8=U9-U3
290 U0=(U8+U9)/E2:GOSUB 100:IF W<>V9 THEN U9=U0
300 IF W=V9 THEN U8=U0
310 IF 54*(U9-U8)>=EZ THEN 290
320 W=V9:V9=1.0-W
330 IF W<>0.0 THEN DRAW XA,YA X,Y 15
340 XA=X:YA=Y:RETURN
350 FOR K1=E0 TO N1:U0=U1+K1*U3:W0=W1:GOSUB 100:XA=X:YA=Y:V9=W:FOR K2=E1 TO
N2:W0=W1+K2*W3:GOSUB 210:NEXT:NEXT
360 T0=E2:U3=(U2-U1)/N2:W3=(W2-W1)/N1
370 FOR K1=E0 TO N1:W0=W1+K1*W3:U0=U1:GOSUB 100:XA=X:YA=Y:V9=W:FOR K2=E1 TO
N2:U0=U1+K2*U3:GOSUB 210:NEXT:NEXT
380 PRINT :INPUT " ";A$:MODE 6A:INPUT "DO YOU WISH TO MAKE ANY CHANGES";A$:
PRINT :IF A$(">") THEN GOSUB 1060:GOTO 820
390 END

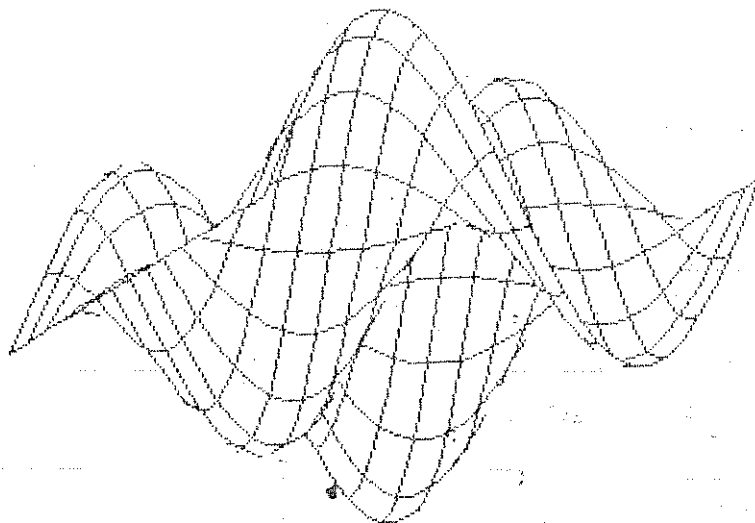
800 REM
810 GOSUB 1000
820 COLORG 0 15 5 12:MODE 6
830 S4=A1/(U2-U1):S2=B1/(W2-W1):S3=C2/(Z2-Z1):C1=0.519615:S1=0.3:S5=
0.577353:S9=A1-C1*B1+R1
840 S6=R2-S1*B1:U3=(U2-U1)/N1:W3=(W2-W1)/N2:S7=R2-S5*(A1+R1)
850 E0=0.0:E1=1.0:E2=2.0:EZ=0.5:T0=E1
860 GOTO 350
1000 A1=182.0:B1=182.0:C2=136.0:R1=96.0:R2=55.0:N1=15.0:N2=30.0:EA=10.0

```

```

1000 PRINT "ANSWER TO QUESTIONS WITH NO RETURN ARE:"
1060 PRINT "DO YOU WISH TO CHANGE THE FUNCTION";:INPUT A$:PRINT
1070 IF A$(">") THEN PRINT "INSERT YOUR FUNCTION AT LINE 10":PRINT "TYPE RUN
    TO CONTINUE":END
1080 PRINT "DO YOU WISH TO CHANGE"
1090 PRINT TAB(6);"THE ORIGINS OR THE LENGTHS OF THE AXES ";:INPUT A$:PRINT
1100 IF A$="" THEN 1130
1110 PRINT "DEF. SCREEN X- AND Y-COORDINATES OF ORIGIN ";R1;R2:INPUT R1,R2:
    PRINT
1120 PRINT "SCREEN LENGTHS OF X-,Y- AND Z-AXIS ";A1;B1;C2:INPUT A1,B1,C2:
    PRINT
1130 PRINT "DO YOU WISH TO CHANGE THE INTERVALS X, Y OR Z ";:INPUT A$:PRINT :
    IF A$="" THEN 1170
1140 PRINT "BEGIN AND END POINT OF X-INTERVAL ";U1;U2:INPUT U1,U2:PRINT
1150 PRINT "BEGIN AND END POINT OF Y-INTERVAL ";W1;W2:INPUT W1,W2:PRINT
1160 PRINT "BEGIN AND END POINT OF Z-INTERVAL ";Z1;Z2:INPUT Z1,Z2:PRINT
1170 PRINT "DO YOU WISH TO CHANGE THE NUMBER OF LINES OR THEN NUMBER OF
    PLOTSTEPS PER LINE (DEFAULT=15,30) ":INPUT A$
1180 IF A$="" THEN 1200
1190 INPUT "NUMBER OF LINES, NUMBER OF PLOTSTEPS PER LINE";N1,N2
1200 INPUT "DO YOU WISH TO OMIT THE HIDDEN LINES (0=NO, 1=YES)";PB:PRINT
1210 IF PB=1.0 THEN PRINT "DO YOU WISH TO CHANGE THE PRECISION OF THE
    VISIBILITY-DETECTION ROUTINE (DEFAULT= 10 STEPS)":INPUT A$:PRINT
1220 IF PB=1.0 THEN IF A$(">") THEN INPUT "NUMBER OF STEPS TO BE USED FOR
    VISIBILITY - DETECTION";EA:PRINT
1230 RETURN

```



audiocassetterecorder van Tandberg, de TCCR-530.

De TCCR-530 is uitgerust met een 8048 processor, die via via een RS-232-C 25-pins D-connector kan communiceren.

Hij accepteert ASCII-strings die afgesloten dienen te worden met <CR> oftewel CHR\$(13); bijvoorbeeld "PL"<CR> staat voor het commando "PLAY".

De TCCR-530 ziet er uit als een diplomatenkoffertje (breedte 44,5 cm, hoogte 14,2 cm, diepte 28 cm), heeft een enigszins oplopend bedieningspaneel, waarop onder andere een gecombineerde seconden- en minutenteller zichtbaar is.

De normale functies van een audiocassetterecorder zijn via wat klein uitgevallen toetsen bereikbaar en er is een RECAP-toets voor het terughalen van het laatste geluidsfragment.

Naast de RS-232 geeft het achterpaneel nog verbindingsmogelijkheden te zien voor een LINE-verbinding met radio of tuner/versterker naast een REMOTE CONTROL waarop bijvoorbeeld een infrarood- of een voetbesturing (niet meegeleverd), maar ook een met relais schakelende cassettecontrolepoort van een computer kan worden aangesloten.

De LINEverbinding is uiteraard ook geschikt om computerprogramma's van tape te laden, waarbij de speaker kan worden dichtgedraaid. Ook voor Basicode uiteraard.

Er is een speaker ingebouwd, maar die is via de speakerplug uitschakelbaar, zodat via externe speakers of een koptelefoon gewerkt kan worden.

Tot nog toe klinkt dat allemaal vrij normaal voor een audiocassetterecorder, maar het zou geen Tandberg zijn, als er niet veel meer te beleven zou zijn.

Allereerst zijn er wat fluitsignalen beschikbaar, die de gebruiker attent maken op zaken zoals bandeinde en cassette-inlegfouten, naast signalen die je waarschuwen als je geluid wilt opnemen zonder cassette of op een beschermde cassette. Comfortabele service van de processor, nietwaar?

Ik gaf al aan dat er een digitale teller op het frontpaneel te zien was. Deze telt seconden en minuten en schakelt na 59 minuten en 59 seconden weer over op 00.00, dat is pas "real time".

In beginsel aardig om de werkelijke bandlengte van je cassettes eens mee te controleren. (Blijkt aardig te kloppen overigens.)

Het model dat hier beschreven is, is een mono-uitvoering, maar er schijnt ook een stereoversie te bestaan.

In dat geval kan er een audiokanaal gebruikt worden voor bijvoorbeeld stuurpulsus voor een dia projector.

1. transmit data pin 2;
2. receive data pin 3;
3. signal ground. pin 7.

Om de correcte logische niveaus te bereiken kan het nodig zijn voor uw computer meer pinnen te moeten solderen dan dat hier staat aangegeven, maar de handleiding is daar duidelijk genoeg over.

Van de fabriek uit is de machine ingesteld op 1200 baud bij een karakterlengte van 7 bits en EVEN pariteit.

Klopt dat niet dat moet ofwel aan de computerkant een wijziging plaatsvinden ofwel aan de kant van de TCCR.

Voor dat laatste is er de PARAMETERmode, die via een combinatie van frontpaneeltoetsen bereikt kan worden.

Je kunt dan de BAUDrates veranderen (150, 300, 600, 1200, 2400, 4800 en 9600 Bd). Het recorderadres is instelbaar: het is mogelijk om 96 recorders te adresseren.

Tevens is de antwoordtijd van de TCCR instelbaar, wat voor sommige communicatieprotocollen wel zinnig kan zijn.

Het tellervenster geeft dan de betreffende coderingen op in bijv. hexadecimalen.

Er worden in de handleiding voorgaende voorbeelden gegeven om een en ander voor uw computermerk in te stellen.

Snapt de TCCR uw instellingen niet, dan genereert hij een vraagteken tot verdere stimulering van uw noeste arbeid.

BBC-gebruikers zijn in het voordeel, want voor hun merk wordt in de handleiding een en ander in soft- en hardware nader uitgewerkt.

De volgende tweelettercommando's zijn in gebruik en kunnen vanaf het toetsenbord ingevoerd worden of in de software opgenomen worden. Denk aan <CR> of CHR\$(13) als afsluiting van een stukje tekstfile.

De TCCR echoot uw invoer terug naar de computer, om te kennen te geven dat het complete commando is ontvangen.

ST =STOP	FL =PLAY	RE =RECORD
WF =WIND FAST	RF =REWIND FAST	RC =RECAP
CU =CUE	RV =REVIEW	EJ =EJECT
CC =CLEAR TAPE	MH =MOTOR HALT	MS =MOTOR START
	COUNTER	DK =DISABLE KEYBRD
ES =ENABLE SOUND	EK =ENABLE KEYBOARD	
DS =DISABLESOUND	RT =READ TAPECOUNTER	
RM =READ MODE	PS =PROGRAM SENSOR	GT =GOTO bandpositie
TO =naar positie	SY =synchroniseer	HS =HANDSHAKE
	teller	

Deze collectie commando's geeft wel aan, dat je alle kanten opkunt met de TCCR, je kunt zelfs software en tekst of

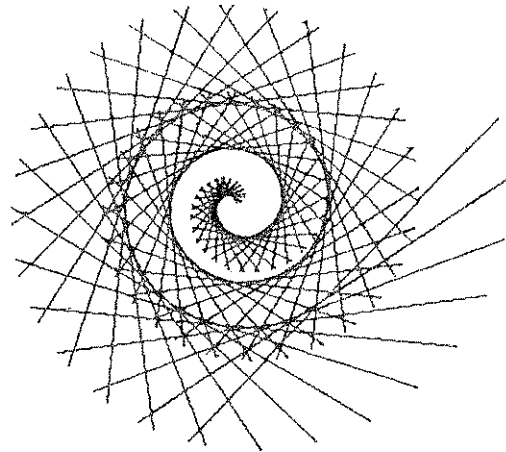
aanbiedingen wil laten weten, laat via zijn computer en zijn auto-dial-modem de clientelen rondbellen, en laat de ingesproken mededeling via de TCCR horen.

Misschien is dit laatste ook een idee voor uw vereniging, om bij wanbetalers het geheugen op te frissen.

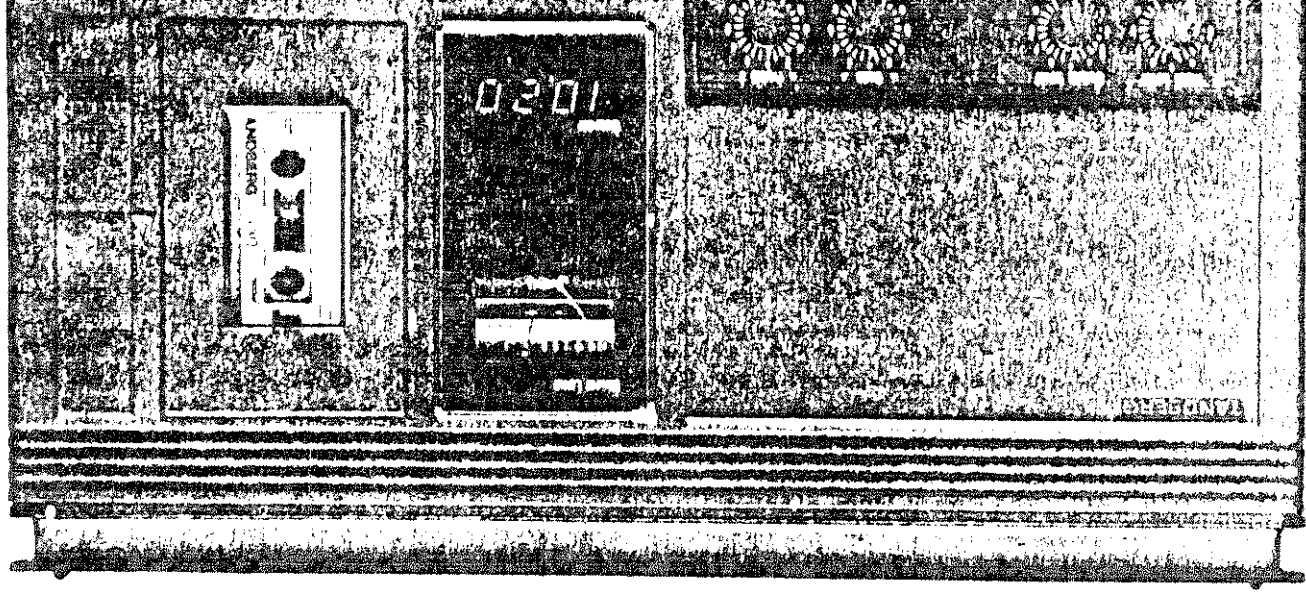
Het leren bedienen van de computer met de TCCR kan ook een toepassingsmogelijkheid zijn.

Er zijn misschien een paar nadeeltjes te noemen, namelijk de overigens uitstekende maar Engelstalige handleiding, het niet meeleveren van een microfoon, en het feit, dat je uitsluitend een cassette kunt verwijderen als het apparaat staat ingeschakeld.

Inno Broekman
tel. 02520-29279



230 V \pm 10%, 50 Hz 115 V \pm 10%, 60 Hz
35 watts maximum
10 - 45°C/50 - 115°F ambient (suitable for use in tropics)
Half track (IEC 574-10)
Compact cassette
1-7/8 ips. (4.75 cm/s)
\pm 1%
0.25% 0.08%
40 - 14,000 Hz \pm 3 dB
58 dB
4
Dual capstan closed loop
Microprocessor controlled
4-digit electronic counter ("real time")
8048
> 75 dB
delta-connector (MIN D) Minimum configuration (no handshake) same connector as RS 232
ASCII-strings terminated by a [CR] Half duplex. Baud rates from 150 to 9600 baud
Sensitivity < 65 mV Impedance \geq 100 kohms Sensitivity 0.2 mV/200 ohms
0.5 V/10 kohms 1.5 V/8 ohms 8 watts RMS in 4 ohms 13 watts music power
17 1/2" (445 mm) 5 1/2" (142 mm) 11 " (280 mm)
13% lbs (6 kg)



Het MDCR " Tape Operating System " of besturingsprogramma voor de digitale cassetterecorder is ondergebracht in een EPROM (2 Kbyte), die ingeplant is op de X-BUS van de DAI PC. Dit programma ondersteunt de MDCR bevelen en gebruikt daartoe enkele adressen in het adresbereik van 0 tot 300 hex.

Het zijn:

120H ERRCO

Dit adres bevat de foutcode d.w.z. de manier waarop eventueel optredende fouten verwerkt worden. Staat op dit adres 0 dan worden de normale foutmeldingen gebruikt. Is de inhoud verschillend van '0' dan wordt een code overeenkomstig, de optredende fout opgeslagen in de accumulator van microprocessor. Deze code kan voor verdere interpretatie van de fout opgevraagd en behandeld worden.

De verschillende codes worden samengesteld door aan enkele van de bits van de accumulator een speciale betekenis te geven:

bit 7 is steeds 0

bit 6 is '1' als de cassette niet uitgerust is met een plug die schrijven toelaat.

bit 5 is '1' als in de recorder geen cassette steekt.

bit 4 is steeds 0

bit 3 is steeds 0

bit 2 is steeds 0

bit 1 is '1' als begin of einde van de band bereikt wordt.

bit 0 is steeds 0

Merken we op dat indien geen fouten optreden de accumulator inhoud '0' is en dat tevens de Z flag van de microprocessor '1' is

11FH DCRSL

Op dit adres staat het nummer van de geselecteerde MDCR.

1B0H FILES

Op dit adres staat het aantal bestanden waarop het gekozen bevel dient uitgevoerd te worden. (REW5: er dienen 5 bestanden teruggespoeld te worden).

297H Op dit adres en het volgende staat het beginadres van de bevelentabel van het TOS programma.

299H Op dit adres brengt de TOS het adres van de stapelwijzer onderindien zulks nodig is.

296H De inhoud van dit adres wordt gewijzigd, zodat bij het interpreteren van de bevelen de 'omweg' gemaakt wordt over de bijkomende bevelentabel.

Tevens worden alle input/outputvctoren van het adressenbereik 2C5H tot 2E3H overschreven door deze nodig voor de besturing van de digitale cassette.

```

528      POP PSW
529      CALL    GIVERT      ;Sla fout op
530      XRA  A
531      JMP     PPCHDB
532  PPPHDB POP PSW
533      JMP     POPHDB
534  GIVERT  INR  A          ;Verhoog foutcode
535  GIVERR  PUSH H
        LHL  D      ERRCO      ;Plaats foutcode
        M  A          ;in ERROR%-variabe

```

BESTAND OPENEN OM TE SCHRIJVEN

```

257      ;
258      ;      Test of het data array dan wel een normaal programma
259      ;      oproepen van OPEN
260      ;      Lezen van status van 'drive'.
261      ;      Starten van cassette.
262      ;      Schrijven van startbyte (AA hex).
263      ;      Schrijven van filetype, lengte, en inhoud van blok.
264      ;      Schrijven van proefsom.
265      ;
266  DWOPEN  PUSH B
267          PUSH PSW          ;filetype
268          PUSH H           ;adre tekstbuffer
269          LHL  D      ERRCO
270          MVI M      OH
271  WRTINF  LHL  D      CURRNT
272          MOV  A,H          ;Bezig met BASIC-programma?
273          ORA  L
274          JNZ  SAVEAR      ;Ja: SAVEA
275          CALL PRTMES      ;INSERT WRITE-ENABLED
276          DW   WRTSTR      ;CASSETTE, TYPE SPACE
277          CALL HAALST      ;Wacht op spatie
278  SAVEAR  STC
279          CALL  OPEN        ;Initialiseer DCR
280          JNZ  WRTINF      ;Fout bij initialisatie
281          POP  H
282          CALL  FWWRDL
283          CALL  XDELAY      ;Wacht even
284          MVI  A      0AAH  ;Verzend start-byte
285          CL

```

Adres : _____

Postcode/woonplaats: _____

bestelt hierbij de volgende software:

_____	prijs	f	_____
_____	g g	g g	_____
_____	g g	g g	_____
_____	g g	<u>g g</u>	_____
	totaal:	f	_____

Het totaal-bedrag heeft hij overgeschreven op postrekening 5314900, t.n.v.
Th. Verberkt te Oud-heusden.

Handtekening:

BESTELLING DAI-GG-SOFTWARE

Te zenden aan Th. Verberkt, Van Buerenstraat 13, 5256 KL Oud-Heusden.

Ondergetekende : _____

Naam : _____

Adres : _____

Postcode/woonplaats: _____

bestelt hierbij de volgende software:

_____	prijs	f	_____
_____	g g	g g	_____
_____	g g	g g	_____
_____	g g	<u>g g</u>	_____