

SINCLAIR

M P U L S

het
SINCLAIR

Gebruikers
Groep

Nummer 10

2e kwartaal 1986

COLOFON

SINCLAIR IMPULS

REDAKTIE ADRES:
REDAKTIE SINCLAIR IMPULS
Wilhelminalaan 42
2625 KH DELFT

ABONNEMENTEN:

Jaarabonnement f 25,00
Abonnementen in het lopend
jaar: f 5,00 minder per
verschenen nummer.

Abonnementen kunnen alleen
worden aangegaan tot einde
van een jaar en door over-
making van het abonnements-
geld op giro: 5693776
t.n.v. Stichting IMPULS
te Den Haag
o.v.v. Donatie + jaar

ADVERTENTIES

Nieuwe advertenties kunnen
alleen worden opgenomen in-
dien deze niet strijdig
zijn met de wet en de be-
palingen van SGG en St.Impuls.

ADVERTENTIETARIEVEN

f 150,00 per blz. per nummer
f 500,00 per blz. per jaar

(bedragen excl. B.T.W.)

HCC-SINCLAIR GEBRUIKERS GROEP

Postbus 142 (let op komen-
1740 AC SCHAGEN de wijzi-
tel.: 02240-13866 ging!)

BESTUUR:

Vit.: Piet van Wees
Secr.: T. A. P. Molenaar
Penn.: George Burghgraefe
Public.: Ed Weijgers
Softw.: Robert Noteboom
Hardw.: Jack Raats
Organ.: Rob van Staalduinen

Jaargang 4, nummer 2 (10)

Verschijningsfrequentie
1x per kwartaal.

SINCLAIR IMPULS is een blad
voor en door gebruikers van
SINCLAIR-computers en een
uitgave van de HCC-SINCLAIR
Gebruikers groep.

REDAKTIE MEDEWERKERS:

Ed Weijgers
Piet Zwager
Albert Hoekman
Rob van Staalduinen

INLEVERING KOPIJ:

De kopij kunt u opsturen naar
het redactie-adres of op de
bijeenkomsten aan de balie af-
geven. Vergoedingen voor ge-
plaatste artikelen worden door
de redactie bepaald en u voor
plaatsing medegedeeld.

SOFTWARE:

Voor het samenstellen van
diverse softwarebanden kunt
u zelf gemaakte software op-
sturen naar:

SINCLAIR IMPULSSOFT

Postbus 142
1740 AE SCHAGEN

Ook hiervoor wordt een vergoe-
ding bepaald.

STICHTING IMPULS

Postbus 212
1740 AE SCHAGEN
Tel.: 070-998791

BESTUUR:

Vit.: Piet van Wees
Secr.: Erik Visser
Penn.: Rob van Staalduinen
Leden: Ed Weijgers
Jack Raats
Piet Zwager

BELANGRIJKE MEDEDELING HCC-SGG

Vlak voor het ter perse gaan van deze IMPULS werd bekend dat
Erik Visser als secretaris van de HCC-SINCLAIR-GEBRUIKERSGROEP
is opgevolgd door dhr T A P Molenaar in Leidschendam. Wij wen-
sen de nieuwe secretaris veel succes. Redactie

=====

INHOUDSOPGAVE VAN DEZE SINCLAIR IMPULS 10 TWEEDE KWARTAAL 1986

=====

- 02 Colofon
- 03 Inhoudsopgave
De landelijke HCC-SinclairGebruikersGroepsdagen in De Bron
De jaarlijkse HCC-Microcomputerdagen in De Jaarbeurs
- 04 Donateurs let op! Penningmeester Stichting Impuls
- 05 Van de redactie rEd
-
- 06 Handig "boot"-programma voor
gebruikers van het BETADISKINTERFACE R van Staalduinen
- 07 Rectificatie op "Kort en klein" uit IMPULS 09 RvS
- 08 Een spatiebalk voor de gewone Spectrum J Klerken
- 09 Het inbrengen van gegevens in tekstfiles I Breeden
- 11 De "ongebruikte" systeemvariabele op 23681 E Weijgers
- 12 Programmalistings via microdrive in tekstfiles EdW
- 14 Muziek-grafiek J Bokma
- 17 Spectrumprinten issue 2 en 3 A Hoekman
- 19 Aanvulling op High-Resolution voor de ZX-81 E Huisman
- 20 Een diskdrive voor de Spectrum P Schim van der Loeff
- 21 Tekens over elkaar drukken EdW
- 22 Tapes kopiëren
Printen alsof het gedrukt staat RvS
- 29 De ZX-Microdrives deel 1 J Bredenbeek
- 32 Programmaversnellings technieken bij BASIC EdW
- 35 QUILL versus TASWORD? J Bredenbeek
- 36 Naschrift SvdL
- 37 Zelfbouw RS232-modeminterface H Koevoets
- 40 "Toolkit 7" in BASIC voor regelmanipulaties EdW
- 43 Tekens verhogen en verbreden
Simpel tekenprogramma EdW
- 44 Spectrum-"bugs" J Bredenbeek
-
- 47 Overzicht van artikelen die te koop zijn via Stichting Impuls

=====

LANDELIJKE SINCLAIRGEBRUIKERSGROEPSDAGEN IN DE BRON TE UTRECHT

=====

De zaterdagse bijeenkomsten van onze HCC-SinclairGebruikersGroep in De Bron, Vader Rijndreef 7 in Utrecht, worden gehouden op:

23 augustus	29 november
11 oktober	17 januari '87

Was U daar nog niet? Van 11 tot 16 uur bent U van harte welkom. Voor leden van de HCC en voor donateurs van de Stichting Impuls.

=====

DE JAARLIJKSE HCC-MICROCOMPUTERDAGEN IN DE JAARBEURS TE UTRECHT

=====

vrijdag 21 en zaterdag 22 november 1986

Dit mag U niet missen! Noteer het dus maar alvast in Uw agenda.

=====

DONATEURS LET OP !

Dit is het laatste nummer, dat u krijgt toegezonden voor de bijdrage van 1984/1985.

Eindigt uw donateursnummer op de omslag van deze Impuls op een 3, dan zult u na dit nummer geen verdere exemplaren meer ontvangen, behalve als u
VOOR 1 AUGUSTUS 1986
nog f 12.50 stort op:

POSTGIRO 5693776

of op

BANKREKENING 45.40.87.446

ten name van:

Stichting Impuls te Den Haag.

Gaarne wel even vermelden:

Aanvulling donatie 1986.

Na 1 aug. a.s. zullen alle "3"-tjes worden gewist en geldt het tarief zoals u dit kunt vinden aan de binnenkant van de omslag!!

Rob van Staalduinen,
penningmeester.

RED

VAN DE REDACTIE

RED

Het nummer dat hier voor U ligt is het tweede van dit jaar 1986, zodat het er eindelijk naar uitziet dat wij nu ons voornemen om vier nummers per kalenderjaar te laten verschijnen daadwerkelijk kunnen uitvoeren. Deze verheugende gang van zaken is te danken aan het feit dat ik er niet helemaal alleen meer voor sta.

Piet Zwager heeft het klaarmaken voor de drukker, onder meer bestaande uit het knippen en precies op A3-vellen plakken, alsook de contacten met de drukker op zich genomen. Zijn eersteling, de moeilijkste bevalling dus, was de vorige IMPULS 9, en verwacht mag worden dat het steeds beter zal gaan.

Met de medewerking van Albert Hoekman en Rob van Staalduinen bestaat de redactie dus thans uit vier man.

Verder zijn er enkele deskundigen die ons medewerking toezegden bij de beoordeling van artikelen die op hun terrein liggen. Maar denkt U nu alstublieft niet dat we U niet meer nodig hebben. De oproepen die ik in IMPULS 8 tot U richtte wil ik hier even dringend herhalen:

WIE KAN DE REDACTIE BIJSTAAN?

Laat ons dus weten op welke terreinen Uw specialiteiten liggen!

STUUR BIJDRAGEN VOOR IMPULS NAAR DE REDACTIE!

Natuurlijk het liefst op file, maar als het niet anders kan ook graag de stof voor een artikeltje op papier inzenden. En nog steeds! neem contact op over nog niet geplaatste stukjes. Zo heb ik op de HCC-dagen van het afgelopen jaar een aardig programma op een cartridge gezet die later vastliep!

STUUR UW PROGRAMMA'S NAAR IMPULSOFT

Over IMPULSOFT gesproken: Onder de nieuwe softwaremanager Robert Noteboom vindt een reorganisatie plaats. Alle bestaande bandjes worden nauwkeurig beoordeeld en er worden nieuwe samengesteld met alleen nog maar de beste programma's. In het volgende nummer verschijnt er een compleet overzicht van.

Zoals U wellicht al merkte hebben we het indelingssysteem van de IMPULS met hoofdstukken en paragrafen naar onderwerpen overboord gegooid. We vonden het te star, en we dachten niet dat die oorspronkelijke opzet om de bladen uit elkaar te halen en op te bergen in een multiband nog door iemand in praktijk gebracht werd. Behalve voordeel voor ons, bij de samenstelling, krijgt U daarvoor artikelen op plaatsen waar eerst al die inhoudsopgaven voor elk hoofdstuk stonden. Eentje voorin elke IMPULS en in het vierde nummer een jaaropgave lijkt ons ruim voldoende. Mogelijk zou daarbij de indeling naar onderwerpen toch overzichtelijker zijn.

De allereerste redacteur van ons blad, Erik Visser, heeft in "De Bron", na zich vele jaren ingezet te hebben voor de Sinclair GG, afscheid genomen als secretaris. Wij bedanken hem en wensen hem het allerbeste toe. Hij kreeg als afscheidscadeau een zeer fraai toetsenbord voor zijn Spectrum, dus we horen vast nog van hem.

HANDIG "BOOT"-PROGRAMMA VOOR GEBRUIKERS VAN HET BETA-DISK-IF

Na wat maanden gewerkt te hebben met disks ontstaat er eigenlijk van zelf een onoverzichtelijke situatie in de catalogus van de diverse disks. Basic, data, code en magic button programma's staan op een gegeven moment kris kras door elkaar en het is vaak zoeken naar de preciese naam van een programma om te laden. Nu zijn er diverse index programma's denkbaar, echter dit programma voldoet aan mijn wensen die er als volgt uitzien:

- het programma moet op iedere schijf te zetten zijn en niet steeds hoeven worden aangepast als er een nieuw programma op de schijf wordt gezet of verwijderd.
- het programma hoeft alleen de basic programma's (of voorprogramma's van machinetaalprogramma's) en de met de magic button geSAVEde programma's te tonen, omdat dat de programma's zijn waarmee ik opstart.
- de programma's moeten duidelijk op het scherm staan en makkelijk te laden zijn.
- het "boot"- programma moet makkelijk op een nieuwe schijf te zetten zijn.
- op een simpele wijze moet ik ook kunnen zien wat er op de schijf in drive "B" staat en er moet dan ook meteen overgeschakeld worden naar deze drive: maar de mogelijkheid van terugschakelen moet aanwezig zijn.

Het volgende programma voldoet aan al deze wensen.

Om het programma te kunnen gebruiken, mag de schijf niet worden beveiligd, omdat de "CAT" tijdelijk op de schijf wordt gezet.

Na een aantal seconden verschijnt er op het scherm een overzicht van wat er zich voor basic en magic-button programma's op de schijf bevinden. Voor iedere programma-naam staat een nummer. Dit tikt u gewoon in (voor de getallen 1 t/m 9 staat een 0 die u ook dient in te tikken) waarna de computer het betreffende programma automatisch gaat laden (ook als het een magic-button programma betreft).

Hierop zijn twee uitzonderingen: Als u ingeeft 01, dan zal de computer een overzicht geven van hetgeen er staat op de disk in drive "B" en wordt drive "B" ook de te gebruiken drive. (als u bij dit overzicht weer de keuze 01 ingeeft komt u weer bij drive "A").

De tweede uitzondering is de keuze 00. Dan zal de computer op de betreffende disk een eventueel al aanwezig "boot"-programma wissen en het nieuwe "boot"- programma op die disk zetten.

Het programma kan maximaal 60 programma's lezen en laden, maar als je je bedenkt, dat er maximaal 128 programma's op een disk passen en veel programma's een of meerdere code-delen achter zich aan slepen, is dit "boot"- programma in bijna alle gevallen goed bruikbaar.

Ik hoop dat veel Beta-Disk gebruikers plezier mogen hebben van dit programmaatje.

-----Rob van Staalduinen-----

SP

LISTING "BOOT"-PROGRAMMA

SP

```

1 DIM A$(32): DIM B$(62,9): L
ET B=1: LET D=15363: RANDOMIZE U
SR D: REM : OPEN #5,"CAT"W
2 RANDOMIZE USR D: REM : CAT
#5
3 GO SUB 15: RANDOMIZE USR D:
REM : OPEN #5,"CAT"R
4 INPUT #5:A$: LET B$(61)=A$(
8 TO 1): FOR C=1 TO 3: INPUT #5:A
$: NEXT C: FOR C=1 TO 64: INPUT
#5:A$: IF A$(1)<>"A" OR B>60 THE
N GO TO 6
5 LET A=0: GO SUB 17: LET A=1
5: GO SUB 17: NEXT C
6 GO SUB 15: RANDOMIZE USR D:
REM : ERASE "CAT"#
7 PAPER 0: INK 7: BORDER 0: C
LS : PRINT TAB 10: BRIGHT 1: INV
ERSE 1:"* ":B$(61, TO 8):" *": I
NVERSE 0: BRIGHT 0: FOR C=1 TO 6
0: PRINT AT C+1-(20 AND C>20)-(2
0 AND C>40),(10 AND C>20)+(10 AN
D C>40):("0" AND C<=9):C: INVERS
E 1:B$(C, TO 8): INVERSE 0: NEXT
C
8 INPUT : PRINT #1: INVERSE
1: BRIGHT 1:"WELK PROGRAMMA NOET
IK LADEN ?": INVERSE 0: BRIGHT

```

```

0: GO SUB 18: LET A=VAL INKEY**
10: GO SUB 18: LET A=A+VAL INKEY
$: GO TO B+(A<B)+2*NOT A
9 IF B$(A,9)="*" THEN RANDOMI
ZE USR D: REM : GO TO B$(A)CODE
10 CLS #: RANDOMIZE USR D: REM
: LOAD B$(A)
11 RANDOMIZE USR D: REM : ERAS
E "boot"
12 CLEAR : RANDOMIZE USR 15363
: REM : SAVE "boot" LINE 1
13 RANDOMIZE USR 15363: REM :
MOVE
14 RUN
15 RANDOMIZE USR D: REM : CLOS
E #5
16 RETURN
17 LET B$(B)=(A$(3+A TO 10+A)
AND A$(12+A)="B" AND A$(3+A TO 1
0+A)<>"boot ")+(A$(3+A TO 10
+A)+"") AND A$(12+A TO 16+A)="C
">192)): LET B=B+(B$(B)O B$(62)):
RETURN
18 IF INKEY$<>"*" THEN GO TO 18
19 IF INKEY$<"0" OR INKEY$>"9"
THEN GO TO 19
20 PRINT #1:INKEY$: RETURN

```

SP

RECTIFICATIE KORT EN KLEIN.

SP

Helaas zijn er door wat heen en weer geschuif met de artikeltjes een aantal kleine foutjes ontstaan in programmaatjes in het vorige nummer van de Impuls. De eerste fout is een opmerking op bladzijde 41 links onderaan:

"Overeenkomstig resultaat levert POKE 57500, lengte": dit dient aangevuld te worden met: "indien u gebruik maakt van het programma Beta Basic."

Een tweede foutje is wat ingrijpender. Doordat er een fout zit in regel 10 van het COPY-programa op bladzijde 43, zal dit niet werken. Regel 10 dient er daarom als volgt uit te zien:

```

10 PRINT AT 0,0:"REGEL ":INT (
t/4)+1:" IS GOED!": NEXT T: RAND
OMIZE USR s: CLS : PRINT "DATA O
.K."""gebruik: COPY $ voor tek
st en"" COPY # voor pla
atjes"

```

Tevens is het raadzaam om in regel 9 CLEAR s te vervangen door CLEAR s-1 omdat achteraf blijkt, dat bij sommige machines het programma hierop nog wel eens wil vastlopen.

-----Rob van Staalduinen-----

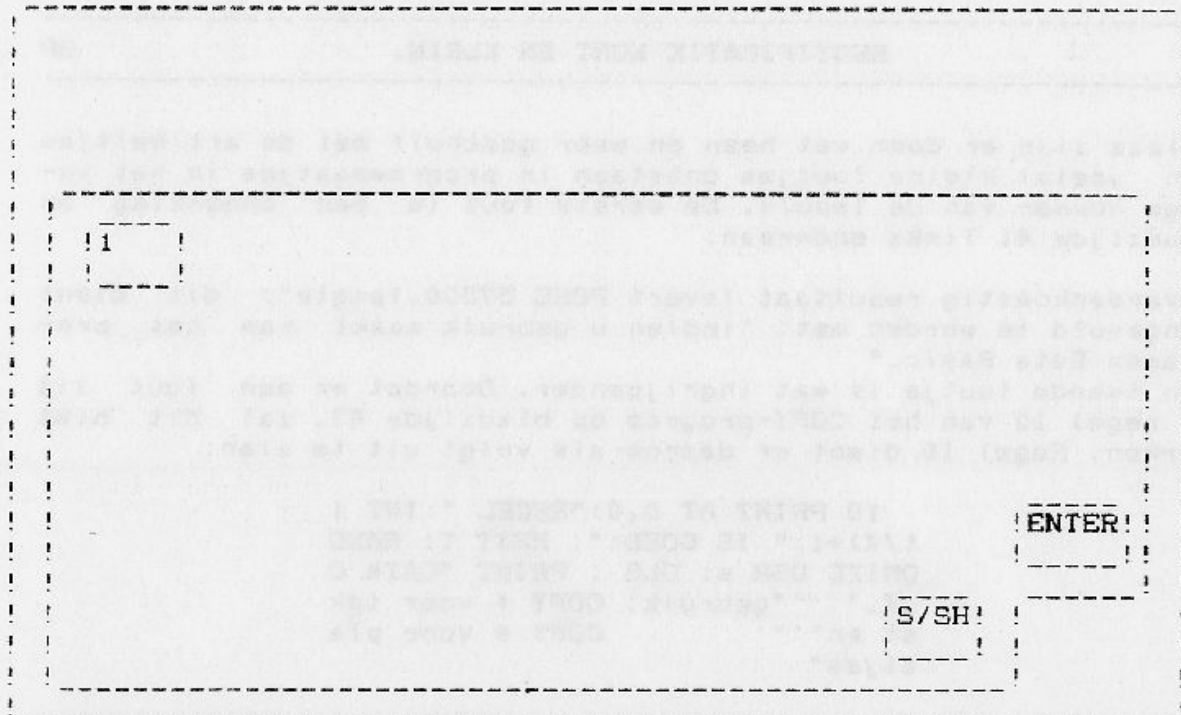
SP

EEN SPATIEBALK VOOR DE GEWONE SPECTRUM

SP

Bij alle besprekingen van de Spectrum kun je er donder op zeggen dat er staat dat het toetsenbord zeer ongeschikt is voor tekstverwerking. Naast andere bezwaren is er dan eentje erg voornaam: het ontbreken van een spatiebalk. Je moet inderdaad met je vingers aan de rechter buitenkant een beetje zoeken naar de spatietoets, en die nog zorgvuldig indrukken wil je resultaat hebben. Maar hier is een betrekkelijk eenvoudige, zelf te (laten) maken oplossing voor. Die bestaat hierin, dat je van een plaatje triplex - liefst van mooi hout - ter grootte van ruim het oppervlak van de Spectrum (toetsenbord en de wat verhoogde strook erboven) een raam uitzaagt. Doe dit zo, dat alle toetsen en de bijschriften bij de toetsen vrij blijven, behalve de break/space-toets. Je kunt het op het tekeningetje hieronder zien.

Wanneer je nu tegen de bovenkant van dit raam een smal strookje triplex of iets anders lijmt, dan voorkom je dat het raamwerkje naar beneden zou kunnen schuiven. Verder kun je tegen de onderkant, langs de smalle strook triplex die in feite de spatiebalk wordt, ook nog een strookje lijmen opdat het raam niet vervormt. Als je zover bent hoef je alleen nog onder het bovenste (dichte) gedeelte van het raam enkele smalle strookjes dubbelzijdig plakband aan te brengen, en het geheel op de juiste plaats vast te drukken. Mocht het niet meteen helemaal goed zitten, dan kun je het altijd nog losmaken (daarom is een strookje van enkele millimeters plakband al voldoende) en de vorm en/of positie nog wat wijzigen. Om de aanslag precies op het juiste moment te krijgen, kun je onder het raam, op de plaats waar dat op de spatietoets drukt, een vulplaatje aanbrengen van kurk of rubber of zoiets. Een bijkomende aardigheid is, dat de Sinclair Spectrum er zo een stuk gezelliger uitziet. Voor wie aan de slag gaat! veel succes!



SP

HET INBRENGEN VAN GEGEVENS IN TEKSTFILES

SP

Het komt wel eens voor dat dat je een programma listing in een tekstfile van je tekstverwerker zou willen inbrengen. Dat is niet zo eenvoudig omdat bij een Sinclair een programma file iets heel anders is dan een tekstfile. Om te voorkomen dat de hele listing zou moeten worden overgetyped heb ik een klein hulp programmaatje gemaakt waarin de print stream wordt omgezet, zodat alle uitvoer die normaal naar de printer zou gaan, nu naar een tekstfile gaat. Op die manier is het mogelijk om met de LLIST- en LPRINT commando's, listings of programma resultaten in een tekstfile in te brengen. Het is zo zelfs mogelijk om spreadsheet resultaten in een tekst te verwerken. Ook is het nu eenvoudig om tekstfiles die u in Basicode formaat hebt gekregen, zoals ik reeds eerder heb beschreven, in uw tekstverwerker in te brengen.

De zo gemaakte files zijn in een formaat, dat geschikt is voor de tekstverwerkers 'tasword two' en 'spectral writer'. Ik geloof dat er nauwelijks andere gebruikt worden op de Spectrum.

Het geheel bestaat uit een kort BASIC programma, waarin met twee POKE opdrachten de printstream wordt omgezet zodat die naar de printerbuffer wijst, en dat vervolgens een machinetaal programma in de printerbuffer laadt. (Die wordt dan immers toch niet meer gebruikt.)

Dit machinetaal programma zorgt er voor dat de 'LLIST'- en 'LPRINT' commando's zodanig werken dat de teksten in ASCII-formaat vanaf de 'RAMTOP' worden neergezet. 'RAMTOP' wordt ingesteld op adres 44999, zodat de textfile vanaf adres 45000 komt te staan. Met het 'CLEAR' commando mogen echter ook andere waarden gekozen worden. De tekstfile begint dan na het door u gekozen adres.

Denk er aan dat het COPY commando niet gebruikt mag worden omdat dit commando de printerbuffer leeg maakt. Gebruik in plaats van het COPY commando:

```
FOR A=0 TO 21: FOR B=0 TO 31: LPRINT SCREEN* (A,B):: NEXT B:
LPRINT: NEXT A
```

De gebruiksaanwijzing is als volgt.

```
1: LOAD "list-tas"
2: LOAD "" (het programma, waarvan u de listing of het
resultaat in tasword- formaat wilt hebben)
3: LLIST of RUN
4: SAVE "naam" CODE 45000, PEEK 23353 + 256 * PEEK 23354
5: LOAD "tasword"
6: LOAD of MERGE vanuit het tasword- menu de textfile "naam"
(die u bij punt 4 ge- SAVED hebt).
```

Het is mogelijk meerdere listings achter elkaar te maken door na punt 3 een ander programma te laden en wederom het LLIST commando te geven. De nieuwe listing komt dan achter de oude.

Wilt u na stap 4 weer een nieuwe file maken, dan moet u eerst de commando's 'POKE 23353, 0 : POKE 23354,0' geven. De nieuwe uitvoer komt dan weer aan het begin van de tekstfile. (Vanaf RAMTOP.)

Ook is het mogelijk de resultaten van het programma achter de listing te krijgen door het programma te runnen. Alle LPRINT opdrachten zorgen voor uitvoer naar de text- file. Ook de PRINT opdrachten hebben diezelfde uitwerking, indien u eerst het commando 'OPEN #2,"P" geeft. (U kunt dit weer ongedaan maken met 'CLOSE #2' of 'OPEN #2,"S".)

Programma: list-tas

```
10 CLEAR 44999: LET a=PEEK 23631+256*PEEK 23632+15: POKE a,0:
POKE a+1,91: LOAD "LIST-TAS"CODE
```

Bytes: LIST-TAS CODE 23296,59

HISOFT GEN83 ASSEMBLER

Copyright HISOFT 1983

All rights reserved

Pass 1 errors: 00

adres:hex-code:	label:	mnemonic:	commentaar:
5B00	10	ORG 23296	:PRINTERBUFFER
5B00 FE0D	20	CP #0D	:ENTER?
5B02 200F	30	JR NZ,TOKEN	:NEE, DAN VERDER
5B04 ED5B395B	40	LD DE,(CURR)	:LAADT DE RELATIEVE PR INTPOSITIE
5B08 3E3F	50 LUS	LD A,#3F	:..OM TE KIJKEN OF BEG IN VAN VOLGENDE
5B0A A3	60	AND E	:..REGEL BEREIKT IS
5B0B CB	70	RET Z	
5B0C 3E20	80	LD A,#20	:ZONIET,DAN SPATIE AFD RUKKEN
5B0E CD1C5B	90	CALL LETTER	
5B11 18F5	100	JR LUS	
5B13 FE45	110 TOKEN	CP #A5	:IS HET EEN TOKEN?
5B15 3805	120	JR C,LETTER	:..NEE, DAN VERDER
5B17 D6A5	130	SUB #A5	
5B19 C3100C	140	JP #0C10	:ZOEK AFZONDERLIJKE LE TTERS VAN TOKEN
5B1C EDEB395B	150 LETTER	LD DE,(CURR)	:RELATIEVE PRINTPOSITI E
5B20 2AB25C	160	LD HL,(RAMTOP)	:BEGIN VAN TEXTFILE
5B23 23	170	INC HL	
5B24 19	180	ADD HL,DE	:ABS. PRINTPOSITIE
5B25 77	190	LD (HL),A	:PLAATS LETTER
5B26 13	200	INC DE	:VOLGENDE POSITIE..
5B27 ED5B395B	210	LD (CURR),DE	:..VAST KLAARZETTEN
5B2B FE20	220	CP 32	:IS ER EEN SPATIE AFGE DRUKT?
5B2D 2805	230	JR Z,SPATIE	
5B2F FDCB0186	240	RES 0,(IY+1)	:SISTEEMVARIABLE FLAGS. GEEF AAN: GEEN SPATIE.
5B33 C9	250	RET	
5B34 FDCB01C6	260 SPATIE	SET 0,(IY+1)	:GEEF AAN: SPATIE. Dit is om dubbele spaties te voorkomen bij het afdrukken van Tokens.
5B38 C9	270	RET	
5B39 0000	280 CURR	DEFW #0000	:RELATIEVE PRINTPOSITI E
50B2	290 RAMTOP	EQU 23730	:DAARACHTER KOMT TEXTF ILE

Pass 2 errors: 00

Table used: 85 from 215

Checksum: 6480 (dec.)

Mocht u niet zo'n fraaie assembler hebben, dan kunt u de de hexadecimale waarden in de kolom 'hex-code' (dat is de tweede kolom. De eerste is het adres in hexadecimale vorm) invoeren met hulp van het volgende programma.


```

1 REM Hexadecimaal invoer programma
10 POKE 23658,8: REM Hoofdletters
20 LET b$="0123456789ABCDEF"
25 LET poging=1
30 INPUT "Wat is het beginadres van de machinecode? (dec.)"
:begin
40 LET checksum=0: LET loper=begin
50 PRINT "Adres:";TAB 9;"Hex:";TAB 16;"Hex-code"
100 LET dec=loper: LET h$=""
110 LET hulp=INT (dec/16): LET mod=dec-hulp*16: LET h$=b$(mod+1)+h$: LET dec=hulp: IF dec>0 THEN GO TO 110
120 PRINT loper;TAB 9:h$,
130 LET a$="Voer Hexadecimaal getal in. (Om te stoppen: all
een Enter.) "
200 INPUT (a$);h$
210 IF h$="" THEN GO TO 400: REM klaar
220 IF LEN h$<>2 THEN BEEP 1,0: LET a$="Je moet groepen van twee tekens invoeren. Opnieuw: "; GO TO 200
230 LET dec=0
240 FOR n=1 TO 2: REM twee hex-tekens,een voor een
245 LET dec=dec*16
250 IF h$(n)>"F" THEN BEEP 1,0: LET a$=h$+" is niet hexadecimaal. Opnieuw: "; GO TO 200
260 IF h$(n)>"A" THEN LET dec=dec+CODE h$(n)-55: GO TO 300
270 IF h$(n)>"9" THEN BEEP 1,0: LET a$=h$+" is niet hexadecimaal. Opnieuw: "; GO TO 200
280 IF h$(n)>"0" THEN LET dec=dec+VAL h$(n): GO TO 300
290 BEEP 1,0: LET a$=h$+" is niet hexadecimaal. Opnieuw: "; GO TO 200
300 NEXT n
310 IF poging=2 THEN IF dec<>PEEK loper THEN BEEP 1,0: INPUT (h$+" Is dat wel goed?(J/N) ");j$: IF j$(1)="N" THEN LET a$="Opnieuw: "; GO TO 200
320 POKE loper,dec: LET loper=loper+1: LET checksum=checksum+dec
330 PRINT h$
340 GO TO 100
400 PRINT : PRINT "Checksum is ";checksum
410 INPUT "Klopt dat? (J/N) ";j$
420 IF j$(1)="N" THEN LET poging=2: GO TO 40
425 IF j$(1)<>"J" THEN GO TO 410
430 INPUT "Onder welke naam moet het ge SAVE d worden? ";j$
440 SAVE j$CODE begin,loper-begin
450 PRINT "Spoel het bandje terug en speel af voor VERIFY "
460 VERIFY j$CODE

```

IVO BREEDEN

BEATRIXLAAN 70

4213 CK

DALEM

SP

DE "ONGEBRUIKTE" SYSTEEMVARIABLE OP 23681

SP

Kortgeleden kwam mij een wonderlijk programmaatje onder ogen. De essentie ervan wil ik U niet onthouden. Wie weet hoe dit werkt?

FOR A=64 TO 71: POKE 23681,A: LPRINT "HOE KOMT DIT NOU?": NEXT A

21E BLZ 10.16

EdW

SP

PROGRAMMALISTINGS VIA MICRODRIVE IN TEKSTFILES

SP

De mogelijkheid om een listing in een tekstfile te zetten heeft, blijkens mijn ervaring daarmee in IMPULS, de volgende voordelen:

- uitgaande van een werkend programma: geen fouten;
- geen problemen om 64 tekens per regel af te drukken;
- herindeling mogelijk, bijvoorbeeld om teksten tussen de regels te plaatsen, de regelnummers voor een kantlijn te halen, afgebroken opdrachten op 1 regel te zetten, enz.

Van Ivo Breeden kreeg ik een korte MC-routine hiervoor, maar ook daarbij treden de spatiefouten op die ik al noemde in IMPULS 09: "SPECTRUMGEKKIGHEID". Om deze en enige andere redenen schreef ik zelf het navolgende programma, in BASIC, dus trager in executie. Het bestaat uit twee delen, het eerste noemde ik "list TO file":

```
0 LET d=PEEK 23766: POKE 23658,8:
INPUT " LLIST rnr",r:
OPEN #3:"m":d:" LLIST ": LLIST r: LPRINT : CLOSE #3:
OPEN #4:"m":d:" LLIST ": LOAD *"m":d:"file TO text"
```

Maak regel 0 van regel 1 met POKE 1+PEEK 23635+256*PEEK 23636,0. SAVE dit eenregelige programma zonder LINE, en op die cartridge ook het tweede deel "file TO text" met LINE 300: de listing werd met dit programma zelf verkregen en daarna heringedeeld, o.a. om op de pagina hiernaast te passen. Nu is het klaar voor gebruik.

MERGE "list TO file" zodra het gewenste programma in Uw SPECTRUM zit en geef RUN. Dan wordt U om het regelnummer gevraagd voor de LLISTing die, na Uw antwoord, naar een hulpfile " LLIST " gaat. Tenslotte wordt "file TO text" geLOAD. Alles via dezelfde drive.

Uit " LLIST " wordt telkens een regel in l\$ gelezen, getoond, en naar het geheugen gePOKEd. Hierbij worden KEYWORDS geëxpandeerd volgens de DATAregels (met hun CODEs als regelnummers), GRAPHICS en UDGs worden vraagtekens, en "control codes" worden genegeerd. Dan worden er spaties toegevoegd tot een veelvoud van 64 tekens.

Opgemerkt zij dat in " LLIST " de BASIC-regelnummers al vertaald zijn in spaties en cijfers en dat er ook geen FPRs in voorkomen. De spaties in de DATAregels zijn zo gekozen dat de genoemde fouten bij dit programma niet optreden. Om de snelheid te verhogen zijn de FOR-NEXT-lussen voor in de eerste drie regels opgenomen. Verdere versnelling is mogelijk met PRINT l\$ in regel 400 ipv 1.

Als l\$ leeg is (door LPRINT in deel 1) of als de 352 regels (het maximum voor SPECTRAL WRITER) bereikt of overschreden zijn, laat regel 400 het aantal der tekstregels zien (een surplus wordt afgebroken). Daarna wordt " LLIST " gewist en wordt de naam en het drivenummer gevraagd. Tenslotte wordt de listing als tekstfile voor SPECTRAL WRITER of TASWOORD op de gewenste cartridge gezet.

Voor TASWOORD kunt U de getallen in de regels 5 en 400 met 2048 verminderen omdat daarbij slechts 320 tekstregels mogelijk zijn.


```

0 FOR p=1 TO LEN c$: POKE a+p, CODE c$(p): NEXT p:
  LET a=a+p-1: NEXT 1
1 FOR a=a+1 TO 64*INT ((a+63)/64): POKE a, 32: NEXT a:
  LET a=a-1: INPUT #4: LINE 1$: PRINT 1$
2 FOR 1=1 TO LEN 1$: LET c=CODE 1$(1):
  IF c>164 THEN RESTORE c: READ c$: GO TO 0
3 IF c>127 THEN LET a=a+1: POKE a, 63: GO TO 5
4 IF c>31 THEN LET a=a+1: POKE a, c
5 NEXT 1: GO TO 400 OR LEN 1$ AND a<62465
165 DATA "RND"
166 DATA "INKEY$"
168 DATA "FN "
170 DATA "SCREEN$ "
172 DATA "AT "
174 DATA "VAL$ "
176 DATA "VAL "
178 DATA "SIN "
180 DATA "TAN "
182 DATA "ACS "
184 DATA "LN "
186 DATA "INT "
188 DATA "SGN "
190 DATA "PEEK "
192 DATA "USR "
194 DATA "CHR$ "
196 DATA "BIN "
198 DATA " AND "
200 DATA ">="
202 DATA "LINE "
204 DATA " TO "
206 DATA " DEF FN "
208 DATA " FORMAT "
210 DATA " ERASE "
212 DATA " CLOSE #"
214 DATA " VERIFY "
216 DATA " CIRCLE "
218 DATA " PAPER "
220 DATA " BRIGHT "
222 DATA " OVER "
224 DATA " LPRINT "
226 DATA " STOP "
228 DATA " DATA "
230 DATA " NEW "
232 DATA " CONTINUE "
234 DATA " REM "
236 DATA " GO TO "
238 DATA " INPUT "
240 DATA " LIST "
242 DATA " PAUSE "
244 DATA " POKE "
246 DATA " PLOT "
248 DATA " SAVE "
250 DATA " IF "
252 DATA " DRAW "
254 DATA " RETURN "
167 DATA "PI"
169 DATA "POINT "
171 DATA "ATTR "
173 DATA "TAB "
175 DATA "CODE "
177 DATA "LEN "
179 DATA "COS "
181 DATA "ASN "
183 DATA "ATN "
185 DATA "EXP "
187 DATA "SQR "
189 DATA "ABS "
191 DATA "IN "
193 DATA "STR$ "
195 DATA "NOT "
197 DATA " OR "
199 DATA "<="
201 DATA "<>"
203 DATA " THEN"
205 DATA " STEP "
207 DATA " CAT "
209 DATA " MOVE "
211 DATA " OPEN #"
213 DATA " MERGE "
215 DATA " BEEP "
217 DATA " INK "
219 DATA " FLASH "
221 DATA " INVERSE "
223 DATA " OUT "
225 DATA " LLIST "
227 DATA " READ "
229 DATA " RESTORE "
231 DATA " BORDER "
233 DATA " DIM "
235 DATA " FOR "
237 DATA " GO SUB "
239 DATA " LOAD "
241 DATA " LET "
243 DATA " NEXT "
245 DATA " PRINT "
247 DATA " RUN "
249 DATA " RANDOMIZE "
251 DATA " CLS "
253 DATA " CLEAR "
255 DATA " COPY "
300 CLEAR 4E4: LET a=4E4: GO TO 1
400 LET 1=a-4E4: PRINT '1/64: IF 1>22528 THEN LET 1=22528
500 CLOSE #4: LET d=PEEK 23766: ERASE "m":d:" LLIST ":
  INPUT "tekstnaam", 1$, "drivenummer", d:
  SAVE *"m":d:1$CODE 40001, 1: VERIFY *"m":d:1$CODE

```

SP

MUZIEK - GRAFIEK

SP

- Dit programma tekent een grafiek die bestaat uit 32 'balken' welke bij geluid hoger worden en bij het ontbreken van geluid weer naar beneden zakken. Het programma bestaat uit twee delen, een BASIC gedeelte (dat U toelaat om de elementen waaruit de grafiek wordt opgebouwd te veranderen, de afstand in verticale richting tussen twee elementen te veranderen en een normale of een gekleurde achtergrond te verkrijgen) en een MACHINE-TAAL gedeelte, welke het inlezen van geluid en het tekenen van de grafiek verzorgt.

Programma 1

- Eerst typt u het BASIC programma in dat gebruikt wordt om het machinetaal gedeelte aan te sturen (Listing 2). Nadat u dat naar band heeft weggeschreven met SAVE "Muziek-GFX" LINE 300 en geverifieerd heeft typt u na NEW Programma 1 in en voert u na RUN de bytes uit Tabel 1 in.

```
10 CLEAR 31999: LET c=8936: LET x=32000: LET a=0
20 INPUT "Byte:":b: PRINT TAB a/b: POKE x,b: LET c=c-b: LET a=a+4: LET x=x+1:
40 IF x<32086 THEN GO TO 20
50 IF c THEN PRINT "Checksum error": STOP
60 SAVE "Muziek-GB"CODE 32000, 86: VERIFY "Muziek-GB"CODE
```

Listing 2

```
10 REM ** MUZIEK-GRAFIEK **
20 LET s=1: LET m=s: LET n=s: LET b$="01111110": LET a$="baaaa aab": FOR f=USR "a"+1 TO USR "a"+6: POKE f,126: POKE f+8,66: NEXT f: POKE f+2,126: POKE f+7,126
30 CLS: PRINT AT 1,9:"MUZIEK-GRAFIEK": OVER 1:AT 1,9:"-----":AT 3,5:"c 1985 Jack The Hacker"
40 PRINT "" (R) - Grafiek"" (S) - Verander Stap"" (E) - Verander Element"" (N) - Normale Achtergrond"" (G) - Gekleurde Achtergrond"" (M) - Mode"" (H) - Halt"
50 PRINT AT 20,1: PAPER 1:"Stap:":s: Element:":a$:" Mode:":m
60 POKE 22914+n*64,71: POKE 22914+64*NOT n,120
70 PAUSE NOT PI: LET i$=INKEY$: BEEP .01,40: IF i$="M" THEN LET m=NOT m
80 IF i$="N" THEN LET n=1
90 IF i$="S" THEN INPUT "Stap [1-8]:" : LINE s$: IF s$>"0" AND s$<"9" THEN LET s=VAL s$
100 IF i$="H" THEN CLS: STOP
110 IF i$="E" THEN GO SUB 250
120 IF i$="G" THEN LET n=NOT PI
130 IF i$<"R" THEN BEEP .1,7: GO TO 50
140 REM ** GRAFIEK **
```

- De a's en b's in a\$ zijn Graphic A en B. (Dit is ook zo na de INPUT in reg. 260).

- De twee POKE's in regel 60 POKEN in het ATTR-geheugen dat begint op 22528. POKE 22914,120 heeft dus als resultaat dat op 12,2 (nl. 22528+32*12+2=22914) Een wit blokje (PAPER 7, INK 0, BRIGHT 1 nl. (0+7*8+64=120) verschijnt.

- POKE 32067,VAL ("BIN"+b\$) in regel 160. VAL ("BIN"+b\$), eerst tussen de haakjes, stel b\$="01111110" , wordt dus VAL ("BIN 01111110" = 126.

- PRINT #0:AT 0,0: = Gebruik regel 22 en 23. (Regel 150)
- INPUT , = Wis onderste twee regels. (Regel 160)

- POKE 23693,71 (reg. 300) Deze POKE verandert de kleuren op het beeldscherm (Na CLS). Dit commando is


```

150 PRINT #0:AT 0,0:"Verbindt d
e EAR ingang met een muzieksign
aal. 'Q' = Menu.": PAUSE 200
160 INPUT ,: POKE 32067,VAL ("B
IN "+b$): POKE 23681,s: POKE 320
32,32+m*8
170 CLS : IF n THEN GO TO 190
180 PRINT ,,: FOR f=7 TO 1 STEP
-1: PRINT INK f,,,,,: NEXT f
190 RANDOMIZE USR 32000
200 GO TO 30
250 REM ** VERANDER ELEMENT **
260 INPUT AT 0,20:"BBBBBBBB"
Element (Binair):" : LINE b$: I
F LEN b$<>8 THEN GO TO 260
270 FOR f=1 TO 8: IF b$(f)<>"0"
AND b$(f)<>"1" THEN GO TO 260
280 LET a$(f)=CHR$ (144+(b$(f)=
"0")): NEXT f: RETURN
290 REM ** Auto Start **
300 POKE 23693,71: BORDER NOT P
I: POKE 23658,8: CLEAR 31999: LO
AD ""CODE: RUN

```

te vergelijken met PAPER 0
:INK 0:BRIGHT 1. (nl. 71 =
64+8*0+7, zie ook handboek
bij ATTR.)
- POKE 23658,8 (reg.300)
Dit commando zorgt ervoor
dat de computer in
CAPSLOCK komt te staan.

c 1985 By Jack The Hacker.

Tabel 1 Bytes Checksum:8936

32000 -	243	58	72	92	15	15	15	246
32008 -	8	211	254	33	0	91	6	32
32016 -	54	0	35	16	251	17	0	91
32024 -	14	0	219	254	203	119	26	71
32032 -	32	18	183	40	33	197	205	170
32040 -	34	193	54	0	58	129	92	237
32048 -	68	128	24	17	58	129	92	128
32056 -	254	176	48	10	71	197	205	170
32064 -	34	193	54	126	120	18	19	121
32072 -	198	8	79	32	205	62	251	219
32080 -	254	31	56	193	251	201		

Assembly listing van Muziek-Grafiek (c) 1985 By Jack The Hacker

LINE	LABEL	OPERAND	COMMENTAAR
0000		ORG 32000	- Startadres = 32000
	:	START	
0010	START	DI	- Interrupts uit
0020		LD A,(23624)	- A-reg bevat de BORDER kleur * 8
0030		RRCA	- Delen door 8 (= 2^3)
0040		RRCA	
0050		RRCA	
0060		OR 8	- Set bit 3 (inlezen van geluid gaat nu beter) en schrijf dit naar 254.
0070		OUT (254),A	
	:	CLRBF	
0080	CLRBF	LD HL,23296	- De Printbuffer wordt als tabel voor de y-posities gebruikt.
0090		LD B,32	
0100	CLR	LD (HL),0	- Maak al de 32 posities nul.
0110		INC HL	
0120		DJNZ CLR	

```

: DRAW
0130 DRAW LD DE,23296 - Het DE-register wordt als
0140 LD C,0 - tabelwijzer gebruikt en C voor de
x-positie.
0150 DRAW1 IN A,(254) - Lees de byte van poort 254 in.
0160 BIT 6,A - Test bit 6. (Wel of geen geluid.)
0170 LD A,(DE) - Y-positie in het A-register
0180 LD B,A - en in het B-register
0190 JR NZ,UP (*) - Was bit 6 een dan naar UP.

: DOWN
0200 DOWN OR A - Is de Y-pos. nul ?
0210 JR Z,EXIT - Zo ja dan naar EXIT
0220 PUSH BC - Het BC-register op de stapel.
0230 CALL 22AAH - Bereken het adres in het beeldscherm
- geheugen dat overeenkomt met de
PLOT-coördinaten in het BC register.
0240 POP BC - Haal het BC-register van de stapel.
0250 LD (HL),0 - Wis het oude streepje uit.
0260 LD A,(23681) - Haal de stapgrootte op.
0270 NEG - Maak deze negatief
0280 ADD B - Voeg er B aan toe (is dus gelijk aan
B-A !)
0290 JR EX0 - Spring naar EX0

: UP
0300 UP LD A,(23681) - Haal de stapgrootte op.
0310 ADD B - Voeg er B aan toe
0320 CP 176 - Vergelijk A met 176
0330 JR NC,EXIT - A > 175, dan naar EXIT.
0340 LD B,A - Nieuwe Y-positie in B
0350 PUSH BC - Zie DOWN
0360 CALL 22AAH
0370 POP BC
0380 LD (HL),126 - Plaats een streepje op het beeld.
(126=01111110).
0390 LD A,B - A=B
0400 EX0 LD (DE),A - Berg nieuwe y-positie op in de tabel

: EXIT
0410 EXIT INC DE - Maak tabelwijzer een hoger.
0420 LD A,C - A=C
0430 ADD B - Tel B op by de X-positie (8 pixels).
0440 LD C,A - C=A
0450 JR NZ,DRAW1 - Herhaal dit 32 maal.
0460 LD A,251 - Lees poort 64510 in. (254+251*256=
0470 IN A,(254) 64510 is rij 'Q' t/m 'T').
0480 RRA - Schuif de Byte naar rechts
0490 JR C,DRAW - Bit 0 =1 dan Carry, G niet ingedrukt
,doorgaan met programma.
0500 EI - Zet interrupts weer aan.
0510 RET - Keer terug naar BASIC.
9999 END

```

- Opmerkingen:

- Het adres 23681, wat gebruikt wordt voor het opslaan van de stapgrootte, is een ongebruikte systeemvariabel.
- Op sommige computers doet het programma precies het omgekeerde, de grafiek stijgt bij geen geluid en daalt als er wel geluid aanwezig is. In het BASIC programma kunt u dit

aanpassen door toets 'M' (= MODE) in te drukken.

(De JR NZ,UP wordt dan door een POKE in JR Z,UP veranderd.)

- De maximale hoogte van de grafiek kan veranderd worden door de CP 176 bij up te veranderen in CP max+1.
- Het machinetaal gedeelte is relocatable d.w.z. het kan ook op een ander adres geplaatst worden (en zo gecombineerd worden met andere programma's).

Literatuur:

- zakboekje voor de ZX-Spectrum
- The Spectrum MACHINE CODE Reference Guide (ROM disass.)

JOHN BOKMA

KALMOES 47

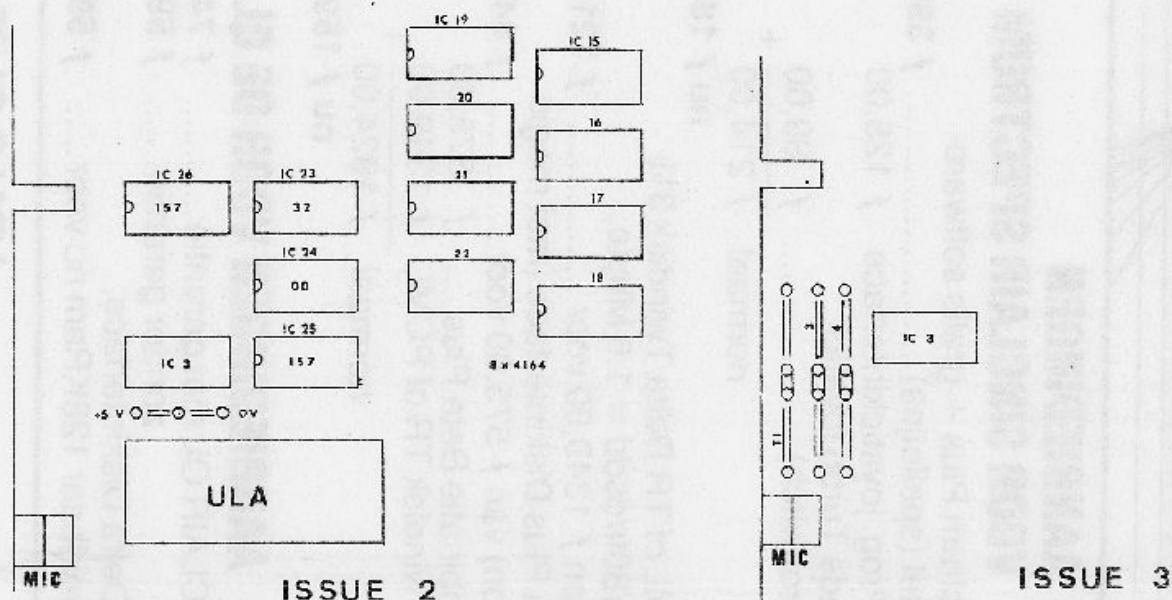
NAALDWIJK

SP

SPECTRUMPRINTEN ISSUE 2 EN 3

SP

Bij "GEHEUGENUITBREIDING VOOR UW SPECTRUM", IMPULS 09-34, kwamen de verschillen in de opbouw tussen de printen van Issue 2 en 3 van de Spectrum aan de orde. Ter verduidelijking hiervan treft U hier twee tekeningen aan, waarin die verschillen goed te zien zijn.



NEDERLANDS GROOTSTE SINCLAIR SPECIALIST brengt o.a.:

AANBIEDINGEN

VOOR SINCLAIR SPECTRUM

Spectrum Plus + gratis software-
pakket (spelletjes) f 399,00
DK Prog. joystickinterface f 125,00
Arcade Turbo joystick
(microswitch) f 89,00
normaal f 214,00 +
nu f 185,00

PCML of TR Delta Twindisk 3 1/2
inch/80tr/dsdd = 1,6 Mbyte,
nu van f 1349,00 voor f 1215,00
Beta Plus Diskinterface (met magic
button) van f 575,00 voor f 449,00
Combinatie Beta Plus f 575,00
en Twindisk TR of PCML f 1349,00
normaal f 1924,00 +
nu f 1599,00

AANBIEDINGEN VOOR DE QL

SINCLAIR QL met garantie f 749,00
zonder garantie f 599,00
TR Delta Diskinterface,
uitbreidbaar 128K Ram nu voor f 599,00

PCML Diskinterface
nu van f 1349,00 voor f 1215,00
In combinatie met TR DELTA of PCML
Diskinterface nu naar keuze de TR
of PCML TWINDISK voor f 1049,00

OPRUIMING

SINCLAIR SPECTRUM en ZX 81 boeken
voor de halve prijs, OP = OP.
Aanbiedingen gelden zolang de voorraad
strekt.

INFORMATIE EN BESTELLEN
KAN OOK TELEFONISCH TUSSEN
10.00 EN 16.00 UUR
☎ 040-456660



KOMIN B.V.

Showroom en kantoor: De Greetstraat 15a
5622 GJ Eindhoven. Tel. 040-456660
Telex 59032 hotel nl. Viditel pag. * 624822 #
ABN Veldhoven, Rekening Nr. 52.82.75.615

VRAAG ONZE GRATIS CATALOGUS!

BESTELLEN BIJ VOORUITBETALING BANK OF CHEQUE MET VERMELDING BESTELNR. VERZENDKOSTEN f 6.50 PRIJZEN INCL. BTW

In het vorige nummer ontbreekt helaas een deel van mijn artikel, doordat ik niet de gehele tekst aan de redactie heb overhandigd. Om dit nu goed te maken is hier het vervolg:

Op een gewone ZX-81 is het niet mogelijk om met deze High Resolution te werken. Het is namelijk nodig om een nieuw beeldscherm geheugen te creëren. Dit geheugen moet dan echter wel de eerste negen adreslijnen A0 t/m A8 met de ROM gemeen hebben. We kunnen dit bereiken door in het geheugengebied van 8-16k extra geheugen bij te plaatsen. Hier is gekozen voor het 8k*8-IC 5565 P15 van Toshiba. Dit is een statisch geheugen-IC van zeer laag vermogen.

Om deze uitbreiding technisch te realiseren werd een print ontworpen welke aangesloten kan worden op de ROM-voet in de ZX-81. Hierop gaan de ROM en het nieuwe geheugen-IC geplaatst worden.

Op deze print is tevens rekening gehouden met de mogelijkheid om een muziek-IC, de Philips SAA 1099, te plaatsen, waarvoor de ondersteunende statements zijn toegevoegd aan de Extended Basic. Alle geluidseffecten zijn, behalve met de geluidsstatements, ook door middel van de IN- en OUT-statements te verkrijgen, voor nog ingewikkelder geluidspatronen bijvoorbeeld.

LEVEL 3.0 BEVAT DE VOLGENDE NIEUWE EXTENDED BASIC STATEMENTS:

DIS	HR/DIS	SKAR	HR/CLS	HR/DRAW
HR/UNDRAW	HR/CIRCLE	HR/UNCIRCLE	HR/FILL	HR/MOVE
SDP	SKS	LDP	LKS	SPEED
HR/COPY	PLAY	SOUND	NOISE	MODE

BESCHRIJVING DER SOUND-STATEMENTS UIT "EXTENDED BASIC LEVEL 3.0"

MODE:K,F,R,X,Y, = GELUID! K=kanaalnummer (1-3); F=kanaal aan (1) of uit (0); R=ruis aan (1) of uit (0); X=volume rechts; Y=volume links.

NOISE:K = GELUID! Zet de ruisgenerator aan van kanaal K (1-3) mits dit kanaal door MODE is ingeschakeld

PLAY:K,O,N,D, = GELUID! Geeft een toon uit kanaal K in oktaaf O met een toonhoogte in dit oktaaf N (1-255) en een tijdsduur D. Er wordt een tabel bijgeleverd waar voor elke hele en halve noot in een oktaaf de juiste waarde voor N gegeven wordt.

SOUND:X = GELUID! Zet het totaal geluid aan (X=1) of uit (X=0) van de door MODE ingestelde kanalen.

Meer informatie over dit programma is verkrijgbaar bij de auteur dhr. F.G. Beniest, Linnaeusstraat 16, Den Haag, tel. 070-998237. De print met de aansluitkabel voor de ROM-voet, het geheugen-IC, verdere onderdelen en de software kunt U voor ongeveer fl. 50,-- bij hem bestellen (zonder het muziek-IC dat ca. fl. 25,-- kost).

E.A.J.Huisman Karveel 49-67 8242 VT Lelystad Tel. 03200-42138

SP

EEN DISKDRIVE VOOR DE SPECTRUM

SP

Het werken met een diskdrive lijkt voor menigeen een schier onbereikbaar ideaal. Tja ... met heel veel geld ...

Toch lijkt dat voor de wat smallere beurs ook haalbaar te gaan worden.

Ik kocht enige tijd geleden een THURNALL-diskdrive. Een prima ding, maar met één groot bezwaar: er zijn speciale diskettes nodig die niet meer te krijgen zijn. Thurnall is - zoals zo vaak gebeurt in computerland - over de kop. Meer dan een jaar heb ik mijn files en programma's heen en weer geschoven tussen slechts twee diskettes van nog geen 180 kb ieder! Als ik weer ruimte nodig had (dagelijks dus) moest ik eerst plaats maken door files naar een cassette weg te schrijven. Een oproep in de "Sinclair Gebruiker" (ook alweer wijlen) en de "HCC-Nieuwsbrief" leverde welgeteld twee telefoontjes op.

Een van iemand die mij vertelde dat hij ook geen diskettes had (ik was niet ad rem genoeg om hem te zeggen dat ik blij was dat de 13.999.999 andere Nederlanders mij niet hebben opgebeld voor die boodschap!). Het andere telefoontje was van iemand die wilde weten of ik verder tevreden was. Hij kon namelijk de andere Thurnall-drive kopen. Ik kon hem alleen waarschuwen voor het gebrek aan diskettes. Daarmee dacht ik dat mijn oproepen tevergeefs waren geweest, en ik ploeterde geërgerd voort.

Enige tijd later belde die laatste, inmiddels Thurnall-eigenaar, op met een vraag. Hij had geen manual, maar had de drive wel uit elkaar geplozen. Deze TH-student, G.W. van der Vegt, deed de software, een vriend van hem de hardware. Mijn rol daarin beperkte zich tot het opdoen en inbrengen van praktijkervaring. Die had ik gedurende meer dan een jaar namelijk wel opgedaan. Ik diende ook wensen in vanuit die optiek.

Kort en goed: inmiddels is een aantal mensen bezig naar dat toch wel goede idee achter de Thurnall een eigen disk-interface voor Shugart-compatible drives te ontwerpen. Leerzaam omdat het inzicht in de werking van een diskdrive geeft en aangenaam omdat daardoor de kosten van een disk voor de Spectrum tot aanvaardbare proporties kunnen worden gereduceerd. We denken dat het mogelijk is het interface voor minder dan f 200,- te bouwen.

Indien hiervoor belangstelling is, wil de Stichting Impuls wellicht behulpzaam zijn bij de aanschaf van de moeilijkste onderdelen, zoals de print en de disk-control-chip WD 1770. Er wordt gedacht aan het gezamenlijk bouwen van de interfaces in Utrecht, op onze landelijke Sinclairgebruikersdagen in De Bron. Daarbij kunnen wellicht ook individuele wensen aan het project worden toegevoegd, zoals een printerinterface, joystickinterface of monitoruitgang, printersoftware of een 64-tekens-per-regel-routine in EPROM.

Laat U alvast aan Rob van Staalduinen van de Stichting weten of U belangstelling hebt? Het adres vindt U in de colofon voorin.

SP

TEKENS OVER ELKAAR DRUKKEN

SP

Om een teken van een trema, accent, onderstreping of iets dergelijks te voorzien moet er tweemaal op een plaats worden gedrukt. Deze constatering is simpeler dan de realisatie ervan. Het lijkt mij zinvol U te vertellen hoe dat met een AVT(FAX)-100 gaat, omdat andere printers soortgelijke mogelijkheden hebben, hoewel de codes waarschijnlijk anders zijn; neem daar Uw handleiding voor.

Bij mijn printer gaat het niet met "backspacing" (code B), omdat daardoor alleen maar het voorafgaande teken uit de printerbuffer verdwijnt, zodat het eerste teken niet eens afgedrukt wordt. (Ik vraag mij af wat je er dan eigenlijk aan hebt; weet U dat soms?) Ook bij printers die wel een stapje terug doen functioneert dat bij SPECTRAL WRITER en TASWORD niet als de regel te vol is voor de twee extra tekens die ieder speciaal teken nodig heeft, namelijk een bs-GR (backspace-GRAPHIC) met daarna het overdrukteken.

Het lukt wel door de printer een hele regel terug te laten gaan. Hiertoe staan mij drie zog "skip operations" ter beschikking: elk behoeft een voorinstelling van vijf besturingscodes, welke ik in de BASICregel met de printerinitialisatie opgenomen heb, en een aanroep van twee codes, die ik aan een terug-GR heb gekoppeld. Voor 1/6" terug met "skip 1" kreeg ik bv 27-70-45-0-12 en 27-53.

Nu lijkt het eenvoudig, maar dat is het (nog) niet, want wilt U over een regel heen printen door de volgende met die terug-GR te beginnen, dan komen Uw trema's ed op papier op verkeerde tekens. Dit kan voorkomen worden door die terug-GR als laatste teken aan de eerste regel toe te voegen, mits daarvoor nog plaats over is.

Een regel inlassen met enkel twee terug-GR's erin werkt altijd. Hierdoor springt de printer twee regels terug, zodat de volgende regel over de eerste heen komt. Van beide regels zijn nu alle 64 kolommen bruikbaar, elk overdrukteken staat recht onder het bijbehorend teken, en scherm- en papierkolommen komen weer overeen.

Ook als voor een andere GR in een regel de ruimte ontbreekt, en de volgende er niet mee mag beginnen, omdat alle plaatsen nodig, of op scherm en papier gelijk moeten zijn, dan kan die GR worden opgenomen in zo'n tussenregel, nu met slechts een terug-GR erin.

Het is wellicht verstandig altijd zo'n tussenregel te gebruiken, in plaats van verschillende methodes naargelang van de situatie.

VOORBEELDEN, SUGGESTIES EN TENSLOTTE ...

Wilt U een regel met trema's ed ook nog onderstreept hebben, dan gebruikt U twee overdrukregels, ieder voorafgegaan door een tussenregel. Deze overdrukprocedure kunt U naar believen herhalen.

Na een spatieuze regel kan toch van LF of lettertype veranderd worden in een tussenregel met een terug-GR en de betreffende GR.

Met de drie "skips" ingesteld op 1/6", 1/8" en 1/72" teruggang, kunnen de tussenregels aangepast worden aan iedere gebruikte LF, door middel van combinaties van de bijbehorende drie terug-GR's. Het is nu zelfs mogelijk om accenten wat hoger, en niet tegen de letters aan te plaatsen, want elke printhoogte wordt bereikbaar. Denk dan wel aan volgregeels om hierna weer normaal uit te komen.

Omdat de regelnummering de tekstlengte op papier niet meer zonder tellen en rekenen aangeeft, gebruik ik voor de IMPULS liever mijn pen voor eventueel noodzakelijke trema's, onderstreping ed.

SP

TAPES KOPIEREN

SP

Er bestaat al een groot aantal programma's om cassettes te vermenigvuldigen, maar dit korte en simpele programmaatje slaat mijns inziens toch alles.

Behalve dat u alle soorten van computerprogramma's kunt overzetten op een nieuwe tape, zelfs programma's met een lengte van de gehele 64 k, kunt u ook muziek kopiëren op uw computer. Ja echt... gewone muziekcassettes met echte muziek!

Ook is niet van belang uit hoeveel delen het programma bestaat want het programma gaat continu door tot u een reset geeft of de stekker er uit trekt.

Het enige nadeel is, dat u over 2 cassetterecorders moet beschikken.

Tik het volgende programma in, geef RUN, waarna het programma u zal vertellen wat te doen.

```
1 LET A=1+PEEK 23637+256*PEEK
23638: POKE A,0: PRINT "Geef in
: 1 <ENTER> RUN 3 <ENTER>": STOP
2 CLS : PRINT "TAPE-BACKUP"
"Met dit programma kunt u via de
Spectrum met twee recorders
(een verbonden met de EAR en de
ander met de MIC) direkt kopiee
n maken van ieder bandje (zelfs
muziekbandjes).""Op het scherm
zult u niets zien maar het prog
ramma werkt nu al!""Dit progra
mma is alleen maar te stoppen do
or een RESET of het verwijdere
n van de spanning!""Denk wel
aan de -rechten!": FOR I=USR "A
" TO USR "A"+B: READ A: POKE I,A
: NEXT I: RANDOMIZE USR USR "A":
DATA 243,14,254,237,120,237,121,24,250
```

-----Rob-----

ZIE NR 11 VOOR RECTIFIC.

SP

PRINTEN ALSOF HET GEDRUKT STAAT

SP

Onder deze kop zal ik de komende nummers wat meer informatie geven over het gebruik van de verschillende opdrachten die nodig zijn om de printer diverse dingen te laten doen die hem/haar van huis uit zijn meegegeven. Omdat het bijna onmogelijk is alle gegevens van alle in Nederland verkrijgbare printers in deze lijst op te nemen heb ik gekozen voor een printer waar zeer veel mogelijkheden in zitten: de AVT-180, aangevuld met de specifieke gegevens van de AVT-100 die nog wat extra mogelijkheden biedt. In het schema dat is gepubliceerd in de vorige Impuls, kunt u zien welke opdrachten zullen worden behandeld en met de handleiding van uw printer er naast zult u bij eventueel afwijkende printer-codes toch wel het juiste gedeelte van de nu volgende omschrijvingen en voorbeelden voor uw situatie kunnen vinden, hetgeen zal worden vergemakkelijkt, omdat ik voor de koppen de Engelse termen voor de betreffende opdracht zal gebruiken.

Om het geheel wat overzichtelijk te houden zal ik de printercommando's indelen in 6 hoofdgroepen:

- | | |
|---------------------------|----------------------------------|
| 1 HORIZONTALE FUNKTIES | 4 GRAFISCHE FUNKTIES |
| 2 VERTIKALE FUNKTIES | 5 ZELFDEFINIEERBARE ROM FUNKTIES |
| 3 SCHRIFTSOORTEN FUNKTIES | 6 DIVERSE ANDERE FUNKTIES |

Hopelijk zullen deze artikelen een verduidelijking zijn op uw handleiding en mogen leiden tot een beter begrip van de werking en een effectiever gebruik van uw printer.

-----Rob van Staalduinen-----

1 HORIZONTALE FUNKTIES

* BS *

Naam: Back Space
 Code: CHR\$ 8;
 Functie: na het ingeven van deze code zal de in de buffer aanwezige tekst worden uitgeprint en zal de printkop 1 stap terug doen en het volgende karakter over het laatst uitgeprinte karakter heen printen.

```
10 REM BS test
20 LPRINT "printer XXXXX";
30 LPRINT CHR$ 8; CHR$ 8;
40 LPRINT "=====
```

Opm.: Sommige printers zoals de AVT-100 wissen het laatste karakter i.p.v. een stap terug te doen. Zie hiervoor de omschrijving bij ESC DEL.

* HT *

Naam: Horizontal Tab
 Code: CHR\$ 9;
 Functie: Verschuift de kop tot een volgende van tevoren met ESC D ingestelde (tab-)positie.

Opm.: Voor een voorbeeldprogramma zie ESC D.

* CR *

Naam: Carriage Return
 Code: CHR\$ 13;
 Functie: Brengt de printkop terug naar het begin van de regel.
 Opm.: Indien uw printer hierop is ingesteld, zal eventueel een Line Feed volgen na deze functie.

* CAN *

Naam: Cancel mode
 Code: CHR\$ 24;
 Functie: wist de nog uit te printen tekens in de printerbuffer.

```
10 REM CAN test
20 LPRINT "Cancel mode test"
30 LPRINT "CANCEL MODE"
40 LPRINT "CANCEL"; CHR$ 24;
50 LPRINT "MODE"
```

* ESC < *

Naam: Home Head
 Code: CHR\$ 27; "<";
 Functie: zet de printkop terug in de uitgangspositie.

```
10 REM ESC < test
20 FOR I=1 TO 5
30 LPRINT "SCHRIJFKOP"
40 LPRINT CHR$ 27; "<"
50 NEXT I
```

* ESC D+n1+n2+....+nk+0 *

Naam: Horizontal tab stop set
 Code: CHR\$ 27; "D"; CHR\$ nnn0
 Functie: hiermee stelt u de horizontale tabs in die u via CHR\$ 9 kunt aflopen.

Instell: meestal zijn in het totaal een achttal tab's in te stellen.

Deze instelling gebeurt meestal door bij n in te geven: of het aantal tekens op de horizontale lijn, of een lengte in n/72 inch.

Voor de instelling op tekens geeft u 1 maal de instructie ESC D in gevolgd door de tab's als CHR\$ + positie.

Sluit dan af met CHR\$ 0. Zie hiervoor listing 1.

Indien u gebruik moet maken van een afstand, dan moet de ESC D instructie bij iedere tab worden herhaald.

De tab dient dan berekend te worden d.m.v. de positie t.o.v. het totaal aantal punten per lijn (stel 576). U zoekt dan de positie en geeft in ESC D +CHR\$ (puntenpositie/128) + CHR\$ (restpunten). Zie hiervoor listing 2.

LISTING 1

```
10 REM ESC D+n1+... test
20 LPRINT "012345678901234567890
12345678901234567890123456789"
30 LPRINT CHR$ 27;"D";CHR$ 1;CHR$
  10; CHR$ 20; CHR$ 30; CHR$ 40;
  CHR$ 0;
40 FOR I=1 TO 5
50 FOR Q=1 TO 5
60 LPRINT CHR$ 9;"tab";Q;
70 NEXT Q: LPRINT: NEXT I
```

LISTING 2

```
10 REM ESC D+n1+n2 test
20 LPRINT "012345678901234567890
12345678901234567890123456789"
30 LPRINT CHR$ 27;"D";CHR$ 0; CHR$
  7; CHR$ 27;"D"; CHR$ 0; CHR$
  72; CHR$ 27;"D"; CHR$ 1; CHR$ 16
  ; CHR$ 27;"D"; CHR$ 1; CHR$ 88;
  CHR$ 27;"D"; CHR$ 2; CHR$ 32;
40 FOR I=1 TO 5
50 FOR Q=1 TO 5
60 LPRINT CHR$ 9;"tab";Q;
70 NEXT Q: LPRINT : NEXT I
```

* ESC Q + n *

Naam: Right Margin setting
Code: CHR\$ 27;"Q"; CHR\$ n;
Funktie: hiermee zet u de rechter kantlijn (bij een linker kantlijn van 0 dus het aantal tekens per regel).
Instell: geef bij n in het aantal tekens vanaf het begin van het papier waar de kantlijn moet komen te staan.

```
10 REM ESC Q+n test.
20 FOR I=1 TO 3
30 LPRINT "RECHTER KANTLIJN OP";
  I*10
40 LPRINT CHR$ 27;"Q"; CHR$ (I*1
  0);
50 LPRINT "123456789012345678901
234567890"
60 LPRINT "123456789012345678901
234567890"
70 LPRINT
80 LPRINT CHR$ 27;"Q"; CHR$ 80;
90 NEXT I
```

* ESC U+n *

Naam: Uni-directional print
Code: CHR\$ 27;"U"; CHR\$ n;
Funktie: na ingeven van deze code zal de printer alleen maar van links naar rechts printen, zodat de letters nauwkeuriger onder elkaar komen te staan.
Instell: Bij n=1 is deze code actief en u hebt deze weer op met n=0

```
10 REM ESC U test
20 LPRINT "BIDIRECTIONEEL"
30 LPRINT CHR$ 27;"I";: LET X=0
40 LPRINT "H H H H H H H H"
50 LPRINT "H H H H H H H H"
60 LPRINT "H H H H H H H H": IF
  X=1 THEN STOP
70 LPRINT "'UNIDIRECTIONEEL"
80 LPRINT CHR$ 27;"U"; CHR$ 1;
90 LET X=1: GO TO 40
```

* ESC l+n *

Naam: Left margin setting
Code: CHR\$ 27;"l";CHR\$ n;
Funktie: zet de linker kantlijn op een bepaalde positie
Instell: door bij n een waarde in te geven verschuift u de linker kantlijn n tekens. Listing 1.
Voor de AVT 100 kunt u de linker kantlijn instellen met ESC E en activeren met ESC 4. De berekening voor n1 en n2 is dezelfde als bij ESC D. Listing 2.

Het commando ESC l+n dient te worden ingevoerd, als de printerbuffer leeg is (desnoods eerst een LPRINT geven).

LISTING 1

```
10 REM ESC 1+n
20 LPRINT "0---5---0---5---0"
30 LPRINT CHR$ 27;"1"; CHR$ 5;
40 LPRINT "Print start"
```

LISTING 2

```

10 REM ESC E+n1+n2 + ESC 4
20 LPRINT "0---5---0---5---0"
30 LPRINT CHR$ 27;"E"; CHR$ 0; C
HR$ 36; CHR$ 27;"4";
40 LPRINT "Print start"

```

* ESC p+n *

Naam: Proportional spacing
Code: CHR\$ 27; "p"; CHR\$ n;
Functie: zet proportionele spa-
tie tussen tekens aan
(n=1) of uit (n=0).

```

10 REM ESC p test
20 LPRINT "GEWONE TEKST"
30 LPRINT "HHHHHHHHHiiiiiiiJJJJJJJJ"
40 LPRINT CHR$ 27;"p"; CHR$ 1
50 LPRINT "PROPORTIONELE TEKST"
60 LPRINT "HHHHHHHHHiiiiiiiJJJJJJJJ"

```

* DEL *

Naam: Delete mode
Code: CHR\$ 127;
Functie: wist het laatste teken
dat zich in de printer-
buffer bevindt.
Voor de AVT-100 is dit
CHR\$ 8.

```
10 REM DEL test
20 LPRINT "ABCDEX"; CHR$ 127;
30 LPRINT "FGHIJK"
```

2 VERTIKALE FUNKTIES

* LF *

```

Naam:      Line Feed
Code:      CHR# 10;
Functie:   de in de buffer aanwe-
           zige tekens worden uit-
           geprint, het papier
           wordt 1 regel opgevoerd
           en de eventueel ingege-
           ven SO-code wordt onge-
           daan gemaakt.

```

```
10 REM LF test
20 LPRINT "uw printer heeft nu";
30 LPRINT CHR$ 10;
40 LPRINT "het papier"; CHR$ 10;
50 LPRINT "2 regels opgeschoven"
```

* VT *

Naam: Vertical Tab
Code: CHR# 11;
Functie: verschuift het papier tot de volgende met ESC B ingestelde positie.

Opm.: Voor een voorbeeldprogramma zie ESC 8.

* FF *

Naam: Form Feed
Code: CHR# 12;
Functie: de in de buffer aanwezige tekens worden uitgeprint en het papier wordt 1 vel opgevoerd.

* ESC / +n *

Naam: VFU channel selection
Code: CHR\$ 27;"/"; CHR\$ n;
Functie: voert de met ESC b gezette verticale tabs uit.
Bij n geeft u de keuze van de gezette tab in.

Opm.: Voor een voorbeeldprogramma zie ESC b.
Voor de AVT-100 hebben de functies ESC 5, ESC 6 en ESC 7 een identiek effect.

* ESC 0 *

Naam: 1/8 inch line spacing
Code: CHR\$ 27;"0";
Functie: zet de regelopvoer op 1/8 inch.

```

10 REM ESC 0 test
20 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
30 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
40 LPRINT CHR$ 27;"0";
50 FOR I=1 TO 5
60 LPRINT "REGELAFSTAND 1/8 INCH
  -----"
70 NEXT I

```

* ESC 1 *

Naam: 7/72 inch line spacing
Code: CHR\$ 27;"1";
Functie: zet de regelopvoer op 7/72 inch.

```

10 REM ESC 1 test
20 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
30 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
40 LPRINT CHR$ 27;"1";
50 FOR I=1 TO 5
60 LPRINT "REGELAFSTAND 7/72 INCH
  -----"
70 NEXT I

```

* ESC 2 *

Naam: 1/6 inch line spacing
Code: CHR\$ 27;"2";
Functie: zet de regelopvoer op 1/6 inch. (standaard)

```

10 REM ESC 2 test
40 LPRINT CHR$ 27;"2";
50 FOR I=1 TO 5
60 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
70 NEXT I

```

* ESC 3+n *

Naam: n/216 inch line spacing
Code: CHR\$ 27;"3"; CHR\$ n;
Functie: zet de regelopvoer op n/216 inch, waar n een waarde wordt toegekend die de regelopvoerafstand bepaalt.

```

10 REM ESC 3+n test
20 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
30 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
40 FOR I=1 TO 5
50 LPRINT CHR$ 27;"3"; CHR$ (I*10);
60 FOR Q=1 TO 3
70 LPRINT "REGELAFSTAND ";I*10;"
  /216 INCH"
80 NEXT Q
90 NEXT I

```

* ESC A+n *

Naam: n/72 inch line spacing
Code: CHR\$ 27;"A"; CHR\$ n;
Functie: zet de regelopvoer op n/72 inch, waar n een waarde wordt toegekend die de regelopvoerafstand bepaalt.

```

10 REM ESC A+n test
20 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
30 LPRINT "REGELAFSTAND 1/6 INCH
  -----"
40 FOR I=1 TO 5
50 LPRINT CHR$ 27;"A"; CHR$ (I*10);
60 FOR Q=1 TO 3
70 LPRINT "REGELAFSTAND ";I*10;"
  /72 INCH"
80 NEXT Q
90 NEXT I

```

* ESC B +n1+n2+...nk+0 *

Naam: Vertikal tab setting
Code: CHR\$ 27;"B"; CHR\$ nnn0
Functie: hiermee stelt u de vertikale tabs in die u via CHR\$ 11 kunt aflopen.

Instell: zie hiervoor de omschrijving bij ESC D; waarbij u "regel" leest i.p.v. "tekens" op de horizontale lijn en als maximum 792/72 inch. Het aantal mogelijke instellingen verschilt van type tot type printer en varieert tussen de 8 en 16.

Standaard papier is 11 inch lang en bij een "gewone" instelling van 1/6 inch wordt het dus $11 \times 6 = 66$ regels.

LISTING 1

```
10 REM ESC B+n1+... test
20 LPRINT "START -----";
30 LPRINT CHR$ 27;"B"; CHR$ 1; C
HR$ 5; CHR$ 10; CHR$ 15;
40 LPRINT CHR$ 11;" 1 VT -----";
50 LPRINT CHR$ 11;" 5 VT -----";
60 LPRINT CHR$ 11;"10 VT -----";
70 LPRINT CHR$ 11;"15 VT -----";
```

LISTING 2

```
10 REM ESC B+n1+n2 test
20 LPRINT "START -----";
30 LPRINT CHR$ 27;"B"; CHR$ 0; C
HR$ 12; CHR$ 27;"B"; CHR$ 0; CHR
$ 60; CHR$ 27;"B"; CHR$ 0; CHR$
120; CHR$ 27;"B"; CHR$ 1; CHR$ 5
6;
40 LPRINT CHR$ 11;" 1 VT -----";
50 LPRINT CHR$ 11;" 5 VT -----";
60 LPRINT CHR$ 11;"10 VT -----";
70 LPRINT CHR$ 11;"15 VT -----";
```

* ESC C+n *

Naam: Form length / lines
Code: CHR\$ 27;"C"; CHR\$ n;
Funktie: met deze funktie bepaalt u uit hoeveel regels het te printen blad bestaat.

Standaard is dit 66 regels bij een regelafstand van 1/6 inch.

Instell: de berekening voor n is als volgt: het aantal regels per inch (bij 1/8 inch regelopvoer dus 8) maal de lengte van het papier in inches.

```
10 REM ESC C+n test
20 FOR I=1 TO 26 STEP 5
30 LPRINT CHR$ 27;"C"; CHR$ I;
40 LPRINT "DIT BLAD HEEFT EEN LENGTE VAN ";I;" REGELS !"
50 LPRINT CHR$ 12;: NEXT I
60 LPRINT "HET EINDE"
```

* ESC C+0+n *

Naam: Form length / inches
Code: CHR\$ 27;"C"; CHR\$ 0; CHR\$ n;
Funktie: evenals bovenstaande funktie bepaalt u ook hier de lengte van een bladopvoer; echter nu in inches, zodat wijzigingen van regelopvoerlengten geen invloed hebben op de bladlengte.

```
10 REM ESC C+0+n test
20 LPRINT CHR$ 27;"C"; CHR$ 0; C
HR$ 1;
30 LPRINT CHR$ 27;"N"; CHR$ 2;
40 FOR I=0 TO 20
50 LPRINT I
60 NEXT I
```

* ESC J+n *

Naam: Print out / paper feed
Code: CHR\$ 27;"J"; CHR\$ n;
Funktie: print de gegevens in de printerbuffer en voert daarna het papier n/216 inch op zonder de kop terug te zetten.

```
10 REM ESC J+n
20 FOR I=1 TO 5
30 LPRINT "START -----";
40 LPRINT CHR$ 27;"J"; CHR$ 20;
50 NEXT I
```

* ESC N+n *

Naam: Skip perforation sett
 Code: CHR\$ 27;"N"; CHR\$ n;
 Functie: na iedere bladopvoer zal deze functie er voor zorgen, dat nog n regels extra worden opgevoerd die meetellen bij het nieuwe blad.
 Opm.: het voorbeeldprogramma kunt u vinden bij ESC C+0+n.

* ESC O *

Naam: Skip perforation cancel
 Code: CHR\$ 27;"O";
 Functie: maakt functie ESC N+n ongedaan.

* ESC b+n+ml.....+mk+0 *

Naam: VFU position setting
 Code: CHR\$ 27;"b"; CHR\$ n;
 CHR\$ mmm0
 Functie: hiermee zet u tabs die onafhankelijk van elkaar bruikbaar zijn; dus niet zoals bij ESC B, waar u ze in serie moet aflopen. Binnen de tab's kunt u echter weer series samenstellen, die de printer zonder tussencommando's achter elkaar uitvoert. De verschillende tab's zijn op te roepen met ESC /+n.
 Instell: n staat voor het tab-nummer, waarvan u er in het totaal 8 kunt samenstellen. m staat voor het aantal regels dat binnen de tab opdracht moet worden opgeschoven en hiervan kunt u er 16 in 1 opdracht zetten, maar deze mogen de bladlengte niet overschrijden. De code dient te worden beëindigd met een CHR\$ 0.

Opm.: de omschrijving hierboven lijkt wat ingewikkeld en dat is het ook. Het beste is wat te stoeien met het voorbeeldprogramma, totdat u het onder de knie heeft. Voor printers als de AVT-100 kunt u voor een vergelijkbaar resultaat (maar dan als enkelvoudige tab) gebruik maken van de functies ESC F, ESC G en ESC H. Deze functies kunnen ook een negatieve regel opvoer aan.

```
10 REM ESC b+n+ml+...+mk+0 test
20 LPRINT CHR$ 27;"C"; CHR$ 21;
30 LPRINT "PRINT START -----";
40 LPRINT CHR$ 27;"b"; CHR$ 1; C
HR$ 3; CHR$ 5; CHR$ 10; CHR$ 0;
50 LPRINT CHR$ 27;"b"; CHR$ 2; C
HR$ 4; CHR$ 6; CHR$ 14; CHR$ 0;
60 LPRINT CHR$ 27;"/"; CHR$ 1;
70 LPRINT CHR$ 12;
80 LPRINT "VERTIKALE TAB 1 - PRI
NT START"
90 FOR I=1 TO 3
100 LPRINT CHR$ 11;"TAB 1 NR.";I
110 NEXT I
120 LPRINT CHR$ 27;"/"; CHR$ 2;
130 LPRINT CHR$ 12;
140 LPRINT "VERTIKALE TAB 2 - PR
INT START"
150 FOR I=1 TO 3
160 LPRINT CHR$ 11;"TAB 2 NR.";I
170 NEXT I
```

* ESC j+N *

Naam: Reverse feed
 Code: CHR\$ 27;"j"; CHR\$ N;
 Functie: met deze code geeft u de printer de opdracht om het papier n/216 inch terug te draaien

```
10 REM ESC j+n test
20 LPRINT "REVURSE FIET"
30 LPRINT "'REVERSE FEED"
40 LPRINT CHR$ 27;"j"; CHR$ 108;
50 LPRINT "=====
```

Dit is het eerste deel van een serie over de ZX Microdrives. Deze serie is bedoeld om u wat meer inzicht te verschaffen in de werking van dit snelle opslagmedium. We behandelen o.a. de werking van de Microdrive, de nieuwe Basic-commando's en het werken met files op de Microdrive. Verder zullen we verder ingaan op de mogelijkheid van de Interface 1 om nieuwe Basic-commando's te creëren.

Hoewel deze serie natuurlijk bedoeld is voor Microdrive-bezitters raad ik u toch aan om dit eerste artikel te lezen als u geen Microdrive-bezitter bent, hierin behandelen we namelijk uitgebreid de in- en uitvoerorganen van de Spectrum.

DE OPEN- EN CLOSE COMMANDO'S

Op de bovenste rij van het toetsenbord van uw Spectrum (of Spectrum+) zult u vast wel een aantal commando's zijn tegengekomen waarvan de betekenis u een raadsel is. Dat zijn OPEN #, CLOSE #, MOVE, ERASE, CAT en FORMAT. Het handboek doet er ook nogal geheimzinnig over, er staat stevast "Does not work without Microdrive, etc.". Zonder Microdrive zouden we deze commando's dus niet kunnen gebruiken, maarrr... dat is niet helemaal waar! Twee van deze commando's, namelijk OPEN # en CLOSE #, werken wel degelijk op de gewone huis-tuin-en-keuken Spectrum. Alhoewel u geen wonderen mag verwachten van deze twee commando's blijken ze toch ook op de standaard Spectrum in bepaalde gevallen erg handig te zijn. Om hun werking te kunnen begrijpen moeten we echter wat dieper delven in de opbouw van uw Spectrum (+).

IN- EN UITVOER

Het eerste wat u met uw Spectrum deed toen u hem gekocht had was natuurlijk hem aansluiten op uw televisie. En als u toen de woorden PRINT "Hallo" had ingetikt was het, natuurlijk logisch dat op de televisie het woord "Hallo" verscheen. U gebruikte dus het toetsenbord als invoerapparaat en het beeldscherm (de televisie) als uitvoerapparaat voor uw computer.

Later kwam er misschien nog een printer bij, en had u dus keuze uit twee uitvoerapparaten. Om te kunnen kiezen tussen scherm en printer heeft Oom Clive twee extra commando's toegevoegd: LPRINT en LLIST (COPY laten we even buiten beschouwing) die hetzelfde doen als PRINT en LIST, maar nu gaat alle uitvoer (listings e.d.) naar de printer in plaats van het scherm.

We hebben nu dus al drie in- en uitvoerapparaten gezien: het toetsenbord, het scherm en de printer. Hoewel deze drie het meest voor de hand liggend zijn zijn er echter ook andere mogelijkheden voor in- en uitvoer. U zou bijvoorbeeld de adressen en telefoonnummers van uw kennissen in plaats van naar het scherm ook naar een of ander opslagmedium kunnen printen, of over een netwerk kunnen sturen. Evenzo zou u van dat opslagmedium of netwerk ook informatie kunnen opvragen. Met andere woorden: Invoer hoeft niet altijd van het toetsenbord te komen, en uitvoer hoeft niet altijd naar een beeldscherm of printer te gaan.

Bij het ontwerpen van de Spectrum heeft men hiermee rekening gehouden, en daarom is er in het RAM-geheugen een gebiedje dat "CHANNEL INFORMATION" genoemd wordt. Op de tekening van het geheugen in hoofdstuk 24 van het Spectrum handboek ziet u dat dit gebiedje vlak voor het begin van het BASIC-programma zit. Op

de Spectrum zonder Interface 1 zit dit gebied tussen de adressen 23734 en 23755.

We kunnen dit gebied weer onderverdelen in kleinere gebieden, die "CHANNELS" (kanalen) genoemd worden. Ieder kanaal heeft een naam bestaande uit 1 letter en bevat bepaalde informatie over een bepaald invoer/uitvoer apparaat. Alle toe- en afvoer van informatie van en naar die apparaten loopt via het kanaal van dat apparaat.

Op de standaard Spectrum zijn er vier kanalen, genaamd K, S, R en P. K staat voor Keyboard en verzorgt dus de invoer via het toetsenbord, maar ook de uitvoer naar het onderste gedeelte van het scherm. S staat voor Screen en verzorgt de uitvoer naar het bovenste gedeelte van het scherm. Kanaal R is niet vanuit Basic bereikbaar en wordt gebruikt voor interne doeleinden. Kanaal P tenslotte verzorgt de ZX printer.

Indien u een Microdrive en Interface 1 heeft zijn er nog vier andere kanalen mogelijk, namelijk M voor Microdrive, N voor netwerk en T en B voor RS 232 operaties.

De kanalen van de standaard Spectrum bestaan zijn ieder 5 bytes groot: Twee bytes welke het adres van de machinetaalroutine voor uitvoer bevatten, twee bytes voor het invoer-adres en een byte voor de naam van het kanaal.

Echter, de in- en uitvoer van de Spectrum gebeurt niet rechtstreeks via deze kanalen. Hiervoor worden namelijk zogenaamde "STREAMS" (stromen) gebruikt, welke gekoppeld kunnen worden aan een bepaald kanaal. Door deze stromen "stroomt" dus de in- en uitvoer van de Spectrum naar zijn randapparaten.

We kunnen deze stromen openen en sluiten, en daarvoor zijn dus die OPEN- en CLOSE commando's! Als we een stroom openen koppelen we hem aan een bepaald kanaal en kunnen we hem dus gebruiken voor in- en uitvoer. Als we hem sluiten wordt hij weer afgekoppeld en kunnen we geen informatie meer door deze stroom sturen of ontvangen.

Op de Spectrum wordt een stroom aangegeven met het teken "#" (SYMBOL SHIFT-3) gevolgd door een nummer tussen 0 en 15. Er zijn dus in totaal 16 stromen op de Spectrum, maar voor interne doeleinden zijn er nog drie extra stromen die alleen met behulp van machinetaal kunnen worden geopend.

Als we de Spectrum ingeschakeld hebben zijn er al 4 stromen geopend, namelijk 0 t/m 3. Ze zijn dan als volgt gekoppeld:

Stroom 0 en 1 aan kanaal K

Stroom 2 aan kanaal S

Stroom 3 aan kanaal P.

Met behulp van het OPEN-statement kunnen we ze echter aan andere kanalen koppelen, en ook kunnen we stromen 4 t/m 15 aan een van deze kanalen koppelen, alhoewel dit op de standaard Spectrum weinig nut heeft. Het OPEN-statement moet gevolgd worden door een getal tussen 0 en 15 (het stroomnummer), een komma, en tenslotte een string welke de naam van een kanaal bevat. Hoofd- en kleine letters worden niet onderscheiden.

Tik nu eens het volgende in:

```
OPEN #2,"k": PRINT "Hallo": PAUSE 0
```

U ziet nu het woord "Hallo" op het scherm verschijnen, maar.. op de onderste regel van het scherm, waarop u normaal niet kunt printen! De PAUSE 0 zorgt ervoor dat uw "Hallo" niet onmiddellijk wordt gewist door de "OK" melding.

Met het OPEN statement heeft u bewerkstelligd dat het PRINTen, dat via stroom 2 gaat, niet meer naar kanaal S gaat (het

bovenste schermdeel), maar naar kanaal K (het onderste schermdeel). Indien u een programmalisting hebt staan kunt u ook LIST intikken, de listing zal dan op het onderste schermdeel verschijnen. Dit zal echter een foutmelding veroorzaken als de listing groter wordt dan 23 regels.

De commando's PRINT en LIST gebruiken dus stroom 2, INPUT gebruikt stroom 1 en LPRINT en LLIST gebruiken stroom 3.

Dit kunt u echter veranderen door achter deze statements het #-teken gevolgd door een stroomnummer op te nemen. Bijvoorbeeld PRINT #0:"Hallo": PAUSE 0 print net zoals het vorige voorbeeld "Hallo" op het onderste schermdeel. PRINT #3:"Hallo" print "Hallo" af op de printer in plaats van op het scherm.

Als u probeert te PRINTen via een gesloten stroom, dus bijv. PRINT #4:"abc" krijgt u de foutmelding "O Invalid stream". Had u echter eerst OPEN #4,"S" ingetypt dan zou u gewoon "abc" op het bovenste schermdeel gekregen hebben.

Verder is het zo dat alleen kanaal K gebruikt kan worden voor invoer. Zou u bijvoorbeeld proberen INPUT #3:a\$ dan krijgt u de foutmelding "J Invalid I/O device" omdat stroom 3 normaal aan kanaal P van Printer gekoppeld is en dus niet voor invoer gebruikt kan worden.

Ook achter INKEY\$ kunt u een stroom opnemen, bijv. INKEY\$#1.

Ondertussen zit u nu nog steeds met uw listings die alleen naar het onderste schermdeel gaan, en ik kan me voorstellen dat u nu wel weer normaal wilt printen en list. Tik dus in: OPEN #2,"S" en alles werkt nu weer normaal. U had echter ook CLOSE #2 in kunnen typen. CLOSE sluit normaal een stroom, dus koppelt hem af van het kanaal waaraan hij was gekoppeld. Echter, stromen 0 t/m 3 worden niet echt gesloten, maar worden weer gekoppeld aan hun oorspronkelijke kanalen, dus stromen 0 en 1 aan kanaal K, stroom 2 aan kanaal S en stroom 3 aan kanaal P.

Als u een ZX-Printer hebt zult u vast wel eens een programma gemaakt hebben dat een keuzemogelijkheid tussen scherm en printer heeft. Dit zal wel eens tot lastige constructies leiden, zoals:

```
10 INPUT "Scherm (s) of printer?(p)?" :a$
20 IF a$="s" THEN PRINT "Hallo": GO TO 40
30 LPRINT "Hallo"
40 STOP
```

Met uw kennis van kanalen en stromen kunt u nu echter alles afdoen met slechts een PRINT-statement:

```
10 INPUT "Scherm (2) of printer (3)?" :a
20 PRINT #a:"Hallo"
30 STOP
```

Maar het kan nog eenvoudiger:

```
10 INPUT "Scherm (s) of printer (p)?" :a$
20 OPEN #2,a$
30 PRINT "Hallo"
```

: (rest van het programma)

```
9999 CLOSE #2: PRINT "Dat is alles!": STOP
```

Zo, dat was dan het eerste deel van de Microdrive-serie. Dit deel was, zoals u waarschijnlijk wel gemerkt hebt, ook interessant voor niet-Microdrive-bezitters. In het volgende deel gaan we echter wel in op de speciale Microdrive-commando's (zoals CAT, FORMAT e.d.) welke niet op de standaard Spectrum werken.

----- JAN BREDENBEEK -----

SP

PROGRAMMAVERSNELLINGSTECHNIEKEN BIJ BASIC

SP

Het zou ideaal zijn als we ons bij het programmeren in een taal, in BASIC bijvoorbeeld, niet zouden behoeven te bekommeren om de computer die het programma moet verwerken. Dan konden we mooie, duidelijke, logisch gestructureerde programma's schrijven, met zinvolle namen voor de variabelen en voorzien van toelichtingen, zodat de werking ervan gemakkelijk te volgen en te lezen was. Bij een computer als onze Spectrum is dat evenwel dikwijls niet mogelijk of wenselijk. Om te beginnen moeten we al een speciaal BASICDIALECT gebruiken en voorts vormen de beschikbare geheugen-RUIMTE en de gewenste executieTIJD twee belangrijke beperkingen. De aspecten "mooi", "kort" en "snel" stellen verschillende, vaak tegenstrijdige eisen aan programma's. Geen dier aspecten mogen we verwaarlozen, maar we moeten goed weten welke eisen we laten prevaleren en op welke plaatsen. Zoals de titel aangeeft zullen we ditmaal vooral naar de TIJD kijken, de RUIMTE is al eens aan bod geweest bij de PROGRAMMAVERKORTINGSTECHNIEKEN (IMPULS 08-27) en wellicht wil iemand nog eens wat over de STRUCTUUR schrijven.

Dat BASIC traag is ligt voornamelijk daaraan, dat zo'n programma niet wordt gecompileerd (voor de executie geheel in machinecode vertaald), maar geïnterpreteerd (tijdens de executie, opdracht na opdracht vertaald). Als we echter wat meer van die verwerking te weten komen, kunnen we de executie nog flink wat versnellen. Daartoe beginnen we met twee op elkaar lijkende opzoekprocessen.

Bij een sprong met GO TO / SUB wordt het regelnummer opgezocht vanaf het begin van het programmeergeheugen, waarvan het adres opgeslagen is in de systeemvariabele PROG (23635/6). Hierbij wordt gebruik gemaakt van de twee bytes die in elke regel diens lengte aangeven (zie het schema bij mijn BASICTOOLKIT in IMPULS 09-31). Hoe meer regelnummers er voorkomen lager dan het gezochte, desto langer het duurt. Beperk dus het aantal regels door er meer opdrachten in op te nemen, en zet de vaakst doorlopen regels vooraan. Een programma zal dan niet altijd met RUN starten en op de meest logische wijze van voren naar achteren doorlopen worden. De initialisatie, het toekennen van beginwaarden aan variabelen en het maken van strings, komt dus juist bij voorkeur achteraan. Toelichtende REM's kosten dus niet alleen ruimte, maar ook tijd, tenzij U ze in toch al nodige regels zet, achteraan natuurlijk.

Bij NEXT, en soms bij FOR - zie mijn artikel in IMPULS 08-09 - , moet er ook nog een opdracht binnen een regel opgezocht worden. Dat gaat nog langzamer omdat er geen opdrachtlengthen opgeslagen worden. Neem als voorbeeld de lus FOR t=1 TO 1e6: NEXT t eens. Staat die achteraan in een regel 1 met in totaal 127 opdrachten (meer kunnen er niet geëxecuteerd worden) en geeft U na een minuut BREAK en PRINT t, dan ziet U een groter getal dan wanneer U hetzelfde doet nadat U al die opdrachten in aparte regels zette. Plaats een FOR-opdracht dus altijd vooraan in een regel. Het zal U niet helpen als U na de FOR-opdracht met een nieuwe regel begint, want het opgeslagen opdrachtnummer heeft altijd betrekking op de regel met FOR, die dus juist geheel doorzocht moet worden wanneer die opdracht niet daar, maar in de volgende regel staat.

Dan het opzoekproces bij een variabele. Dat geschiedt vanaf het beginadres van het variabelengeheugen, opgeslagen in de systeemvariabele VARS (23627/8). Hoe meer er voor de gezochte staan hoe langer dat duurt. Variabelen worden aangemaakt in volgorde van

executie. Numerieke variabelen blijven op hun plaats bij waardeverandering, maar bij LET a\$=... bijvoorbeeld wordt een oude a\$ gewist door al wat volgt terug te schuiven, en wordt er voor de nieuwe a\$ achteraan plaatst gemaakt. Deze tijdkostende procedure vindt niet plaats als er een substringvormer (zie IMPULS 09-38) achter staat, dus met LET a\$()=... kunt U dat zondig voorkomen. Denk erom dat FOR een lopende variabele (19 bytes) creëert, een reeds voorkomende gewone wist, waarna die altijd zo blijft. Ook gedimensioneerde variabelen zijn niet meer weg te krijgen, hoogstens kleiner achteraan te zetten door een nieuwe DIM-opdracht. Beperk daarom het aantal variabelen door elke variabele zo vaak mogelijk opnieuw te gebruiken, want op elke plaats een zinvolle naam kost veel variabelen, dus ruimte en tijd. Bovendien moeten lange namen wel iedere keer geheel gelezen en vergeleken worden!

Er zijn ook nog opzoekprocessen bij RETURN, RESTORE, READ en FN. Wie zou die eens precies willen beschrijven in onze IMPULS?

Tenslotte zijn er aanzienlijk verschillen tussen de executietijden voor de diverse opdrachten en functies. Probeer dus tragere te vervangen, of te verhuizen naar minder vaak doorlopen regels. Daarvoor moeten we dan natuurlijk wel weten welke traag zijn en welke sneller. Het is ondoenlijk om hier behalve wat voorbeelden een volledig overzicht te geven. Hebt U voor een programmeerprobleem verschillende oplossingen, zet die dan achtereenvolgens in een lus van het type 1 LET t=t+1: ... : GO TO 1, executeer ze met LET t=0: GO TO 1 en geef, telkens na dezelfde tijd, BREAK en PRINT t zodat U kunt zien welke het snelst is. Ik voorspel U onverwachte uitkomsten. Dan nu een aantal voorbeelden. FOR-NEXT-lussen zijn nogal traag, een lus met GO TO is sneller. Zo iets als IF a=b THEN GO TO 100 met GO TO 10 in de volgende regel kan, als 10 de beginregel is, beter vervangen worden door GO TO 100*(a=b) of nog beter door GO TO 100 AND a=b. Logische bewerkingen zijn sneller dan rekenkundige, buit de mogelijkheden uit van SINCLAIRLOGICA, waarvan een tabel in IMPULS 08-51 staat. Pas liever IF NOT a en IF a dan IF a=0. en IF a<>0 toe. Optellen / aftrekken gaat vlugger dan vermenigvuldigen, wat weer sneller is dan delen. Langzaam zijn de functies met iteratie: die waarvoor we SQR, f, EXP en LN gebruiken en de goniometrische (enige malen LN EXP voor een variabele levert zondig een vertraging op die niet door een toetsindruk beïnvloed kan worden). Concatenatie (is optelling van strings) is ook erg traag en kost bovendien werkgeheugen. Combineer trage PRINT-opdrachten zoveel mogelijk tot eentje met meer "items" (gescheiden door , en ').

De PROGRAMMAVERKORTINGSTECHNIEKEN uit IMPULS 08-27 werken bijna allemaal verdragend. FPR's worden juist in de regels opgenomen omdat daarmee zonder conversie gerekend kan worden. Variabelen, maar vooral trucs als met VAL kosten veel tijd. Vermijdt dat dus op plaatsen waar snelheid belangrijk is. Het kan wel bij een menu bijvoorbeeld, daar wordt toch al op een keuze gewacht. Menu's zetten we daarom ook bij voorkeur achteraan in het programma.

In de volgende IMPULS zullen we bekijken welke invloed de nu nog wat theoretisch behandelde programmaversnellingsstechnieken op de executietijd hebben aan de hand van een praktijkvoorbeeld. Deze artikelen kwamen tot stand na een lezing over dit onderwerp gehouden op onze Landelijke Gebruikersdag van 22 mrt in Utrecht.

computercollectief

Amstel 312 (t.o. Garre) / 1017 AP Amsterdam / Giro 4 475 158 / Bank NMB 69.79.15.646

HIEONDER EEN OVERZICHT VAN ACTUELE EN NIEUW
BINNENGEKOMEN BOEKEN EN SOFTWARE VOOR SINCLAIR COMPUTERS

in BELGIE: IST COMPUTERWINKELTJE pvba, M Sabbestraat 39
B-2800 MECHELEN, tel 015-206645
(E10 afrit 9 Mechelen Nrd, 2e Lichten Links)

- nederlandstalige ZX Spectrum boeken :

*WERKBOEK MACHINETAAL VOOR DE ZX SPECTRUM	f 37,50
inclusief cassette met assembler.	
*VAN BASIC NAAR MACHINETAAL OP DE ZX SPECTRUM ..	f 37,50
ZX SPECTRUM PLUS, tips, programma's, BASIC	f 33,75
ZX SPECTRUM HARDWARE BOEK	f 25,75
LEREN PROGRAMMEREN ZX SPECTRUM 3 (met foto's) ..	f 27,50
LEREN PROGRAMMEREN ZX SPECTRUM 4 (met foto's) ..	f 27,50
MACHINECODE MET DE ZX SPECTRUM	f 30,-
*40 GRAFISCHE PROGRAMMA'S VOOR ZX SPECTRUM	f 29,50
FINANCIELE PROGRAMMA'S VOOR DE ZX SPECTRUM	f 25,-
TOOLKITS en ENKELE SPELEN VOOR DE ZX SPECTRUM ..	f 25,-
*WERKEN MET DE ZX MICRODRIVE	f 32,50
ZX SPECTRUM MACHINETAAL ROUTINES	f 32,75
PROGRAMMATUUR 1 VOOR DE ZX SPECTRUM	f 22,50

- engelse ZX Spectrum boeken :

*COMPLETE SPECTRUM	f 49
SPECTRUM INTERFACING AND PROJECTS	f 35
ASSEMBLY LANGUAGE FOR ARCADE GAMES	f 42
PRACTICAL ROBOTICS AND INTERFACING FOR ZX SPECTRUM	f 32
MAKING THE MOST OF YOUR ZX MICRODRIVE	f 27
ADVANCED GRAPHICS FOR THE ZX SPECTRUM	f 52
ADVANCED SPECTRUM MACHINE LANGUAGE (Webb)	f 36
ARTIFICIAL INTELLIGENCE ON THE SPECTRUM	f 36
MACHINECODE APPLICATIONS FOR THE ZX SPECTRUM ..	f 36
THE COMPLETE SPECTRUM ROM DISASSEMBLY (Logan)	f 49
*THE ASTRONOMERS SOFTWARE HANDBOOK	f 45
ZX SPECTRUM ASTRONOMY	f 36
INSIDE YOUR SPECTRUM	f 36
*SINCLAIR REFERENCE DIARY - 1986 zakagenda	f 16
*HANDBOOK COMPUTERKRACKEN EN BEVEILIGEN	f 32,50

* Onze nieuwe ZOMER 1986 PRIJSLIJST is nu uit! *
* Stuur ons een kaartje met je naam en adres en de *
* vermelding 'Sinclair Impuls' en we sturen hem *
* gratis aan je toe. *
* COMPUTERCOLLECTIEF - Amstel 312 - 1017 AP AMSTERDAM *

- engelse Spectrum boeken :

THE SPECTRUM OPERATING SYSTEM	f 32
SPECTRUM MICRODRIVE BOOK (Logan)	f 32
*SPECTRUM SHADOW ROM DISASSEMBLY	f 45
*THE MICRO CLOAK AND DAGGER BOOK codes, cryptogr. ...	f 36
*THE PURPLE PLANET - MicroPROLOG for the Spectrum ..	f 35
*SPECTRUM + LOGO - complete LOGO Listing	f 16
*SPECTRUM ASSEMBLY LANGUAGE COURSE	f 69

compleet met tape met assembler en monitor programma.

- boeken voor de Sinclair QL :

*SINCLAIR QL LEREN PROGRAMMEREN	f 24,50
THE SINCLAIR QDOS COMPANION (Pennell)	f 36
QL GAMESMASTER (Ewbank)	f 39
QL ADVANCED USER GUIDE (Dickens)	f 79
*QL SUPERBASIC - The Definitive Handbook (Jones) ...	f 59
INSIDE THE SINCLAIR QL (Naylor)	f 36
QL SUPERBASIC - a programmer's guide	f 36
THE QL BOOK OF GAMES (Hurley)	f 36
*ADVANCED QL MACHINE CODE (Denning)	f 45
QL ASSEMBLY LANGUAGE PROGRAMMING (Opie)	f 59
QUICK QL MACHINE LANGUAGE (Giles)	f 39
ASSEMBLY LANGUAGE PROGRAMMING ON THE SINCLAIR QL ..	f 39

ACTUELE EN NIEUW BINNENGEKOMEN SOFTWARE VOOR SPECTRUM/QL

***** SPECTRUM 128 software : *****

MATCHDAY 128	f 45
SAMANTHA FOX STRIP POKER 128 ...	f 39
GLADIATOR 128	f 45
SNEEVO'S WHIRLED	f 45

***** -- arcade games en adventures : *****

*GHOSTS & GOBLINS	f 36
*QUAZATRON	f 39
*BIGGLES	f 45
*THE PLANETS	f 45
CYBERUM	f 45
*BATMAN	f 36
*THEY SOLD A MILLION II	f 45
*SABOTEUR!	f 45
GREEN BERET	f 36
*HEAVY ON THE MAGIC	f 45
*COMMANDO	f 39
TAU CETI	f 49
V	f 36
ZOIDS	f 39
SUPERBOWL	f 45
RAMBO	f 39
*BOMB JACK	f 36
*SPITFIRE 40 spectrum	f 49
4 CRASH SMASHES	f 45
*WEST BANK	f 36
*BACK TO SKOOL	f 35
*LORD OF THE RINGS	f 79

***** -- nuttige programma's : *****

TASWORD III cartridge	f 69
*BETA BASIC 3.0	f 69
TASCOPY screen dump	f 45
*OCF ART STUDIO cassette versie ..	f 69
OMNICALC II spreadsheet	f 69
*ASTRONOMER II planetarium	f 49
MINI OFFICE SPECTRUM	f 29
*WHAM - THE MUSIC BOX	f 49

***** -- denkspelen en simulaties : *****

MONOPOLY	f 49
CLUEDO	f 49
SUPERCHESS 3.5	f 49
*WATERLOO - war game	f 49
FRANK BRUNO'S BOXING	f 35
HYPERSPORTS (Konami)	f 39
*TOMAHAWK	f 49
*YIE AR KUNG FU	f 39
WAY OF THE TIGER	f 45
*SKYFOX	f 39
WORLD SERIES BASEBALL	f 34
*WORLD SERIES BASKETBALL	f 35
MATCHDAY - 3d voetbal	f 39
*WINTER GAMES	f 39
*ELITE 3d ruimte simulatie	f 65

***** -- utilities : *****

SUPERCODE III 150 routines	f 65
PAINT PLUS (opvolger Paintbox) ..	f 49
TRANSFORMER (microdrive util.) ..	f 45
THE ARTIST	f 69
UDG ontwerper	f 36

***** -- programmeertalen : *****

LASER BASIC COMPILER	f 65
LASER BASIC	f 65
ABERSOFT FORTH	f 69
HISOFT PASCAL compiler	f125
HISOFT C - compiler	f125
HISOFT DEVPAK assembler	f 69
WHITHE LIGHTNING microdrive v....	f 95
THE COLT - Hisoft Basic compiler ..	f 65

***** -- Adventure selectie : *****

WORM IN PARADISE	f 49
HACKER	f 39
SHADOW OF THE UNICORN +interface ..	f 79
BORED OF THE RINGS	f 39
SORDERONS SHADOW (Beyond)	f 49
MARSPORE space trilogy	f 45
ARNHEM - wargame	f 45
*THE PRICE OF MAGIC	f 45

***** QL SOFTWARE *****

*HISOFT DEVPAK QL	f 185
*TECHNI QL (2d CAD package)	f 195
*METACONCO PASCAL	f 399
*QL ROM SUPERBASIC EXTENSIONS ..	f 149
MATCHPOINT f 69	BRIDGE II QL .. f 79
HYPERDRIVE f 79	LATTICE C f460
MONQL ROM f149	GRAPH QL ... f110
Q.DOCTOR .. f 95	PSION CHESS .. f 95

winkel open van woensdag t/m zaterdag tussen 11.00 t/m 17.00 (maandag/dinsdag gesloten) - alle prijzen inclusief BTW
verzendkosten f 6 per bestelling - onze NIEUWE ZOMER 1986 CATALOGUS is nu uit! vraag hem aan! (gratis).

microcomputer tijdschriften boeken en software

dealer aanvragen welkom

QL

QUILL VERSUS TASWORD?

SP

In Impuls nummer 8 beschrijft Pieter Schim van der Loeff zijn perikelen met de QL en in het bijzonder met tekstverwerken op deze machine. Hij komt daarbij tot de conclusie dat de combinatie Spectrum/Tasword 2 toch zijn voorkeur geniet boven de QL. Wel schrijft hij erbij dat er aan de QL inmiddels weer het een en ander verbeterd kan zijn.

Welnu, sinds enige maanden bezit ik ook een QL met de tot nu toe beste ROM en software. Ik heb vroeger veel gewerkt met Tasword en nu werk ik met de bij de QL geleverde tekstverwerker QUILL en ik moet nu zeggen dat ik voor geen goud terug wil naar Tasword.

Allereerst is de Spectrum een machine die zich eigenlijk niet goed leent voor tekstverwerking in verband met de lage schermresolutie. 64 Karakters per regel is bij de Spectrum de grens van wat al eigenlijk niet meer mogelijk is (je krijgt letters in een 4 x 8 matrix die zelfs met een monochrome video-ingang nauwelijks te lezen zijn) en vooral met zwarte letters tegen een witte scherm-achtergrond krijg je al gauw last van "beeldscherm-moeheid". Aan de andere kant zijn 64 karakters per regel weer net te weinig voor het uitprinten op A4-papier: de meeste op A4 geprinte documenten tellen doorgaans zo'n 70 a 72 aanslagen per regel.

De QL is wat dat betreft stukken beter omdat de resolutie een karakter van 6 x 10 pixels toelaat bij 85 kolommen per regel. De leesbaarheid is dan ook veel beter dan op de Spectrum, iets wat Schim van der Loeff overigens ook constateert.

Iedereen die met Tasword werkt zal wel op de hoogte zijn van de perikelen met de zogenaamde "vreemde" tekens. Dat houdt in dat je als je bijv. een umlaut of accenten op een letter wilt hebben of over wilt schakelen naar onderstrepen, je moet gaan zitten rommelen met grafische tekens die niet alleen nietszeggend zijn maar ook nog de justificatie in het honderd laten lopen.

Niets van dat alles bij QUILL: zowat alle vreemde tekens zijn in de QL karakterset ingebouwd en kan je dus zondermeer op het scherm krijgen. Ook onderstrepen, vet printen, superscript en subscript kan je zo op het scherm krijgen.

Bovendien is het installeren van die codes voor de printer veel eenvoudiger dan bij Tasword: Je runt gewoon het INSTALL-programma op de QUILL cartridge en kiest dan je printer uit een lijst van pakweg 10 printers. En mocht jouw printer er toevallig niet bijzitten dan is het zelf definiëren van de codes nog altijd stukken eenvoudiger dan bij Tasword.

Wat de bediening betreft: Bij het werken met QUILL krijg je continu zogenaamde "on-screen prompts" op het scherm die je steeds aanwijzingen geven en mocht je er dan toch niet uitkomen dan is een druk op de F1-toets onder alle omstandigheden voldoende om de juiste hulp te geven.

De bediening van Tasword is eenvoudig maar heeft een aantal inconsequenties, bijv. bij invoegen met de AND-toets moet de cursor na de betreffende regel staan maar bij "Invoeg aan" juist weer ervoor. Ook kan je de kantlijn van een document wel groter maar niet kleiner maken, en bij het deleten of invoegen van een woord "tegen de kantlijn aan" heb ik een aantal situaties gehad waarbij Tasword net niet deed wat je verwachtte.

Ook kent Tasword geen zgn. "headers en footers" (koppen resp. voeten boven en onderaan een pagina), iets wat Quill volop heeft met de mogelijkheid pagina's automatisch te nummeren.

Een ding moet ik eerlijkheidshalve wel vermelden en dat is dat de snelheid van de microdrives nog steeds niet geweldig is. Met het SAVEN van een document van zo'n 3 kantjes is de QL al gauw zo'n minuut bezig. Dit ligt voornamelijk aan de manier van opslag van de tekstfiles, het feit dat de laadtijd van QUILL zelf is teruggebracht van 70 seconden bij de eerste versie naar ca. 15 seconden nu bewijst dat het beter moet kunnen. Nu er echter flink wat diskinterfaces voor de QL beschikbaar zijn hoeft dit echter ook niet meer zo'n probleem te zijn. Overigens kun je bij de QL rustig doortypen terwijl de microdrive draait dankzij de toetsenbord-buffer. De Spectrum reageert echter nergens op zolang de microdrive draait.

Er is een heleboel geschreven over de QL, maar als iemand beweert dat de Spectrum met Tasword beter is dan de QL met Quill dan heeft die toch niet echt achter de QL gezeten. Niet alleen op het gebied van tekstverwerking biedt de QL meer, ook op het gebied van programmeren (BASIC en machinecode) laat hij de Spectrum ver achter zich.

Helaas echter schijnt Sinclair ook mee te moeten doen aan de "dumprage": de QL's die nu voor f. 999,- verkocht worden zijn niet van de laatste versie (nl. versie 1.03 ofwel de "JM" ROM, dit kunt u testen met "PRINT VER\$"). Voordat de prijs omlaag ging was versie 1.10 ("JS" ROM) nog te koop welke op een aantal punten verbeterd is (o.m. extra keywords i.v.m. foutbehandeling welke niet gedocumenteerd zijn).

Hetgeen niet wegneemt dat de QL op een prijs van onder de duizend gulden de aangekondigde 128K Spectrum volledig voorbijstreeft in termen van ontwerpfilosofie. De grootte van het RAM-geheugen mag dan gelijk zijn; het maakt echter heel wat uit of dit geheugen wordt ondersteund door twee processoren waarvan een een 32-bit ontwerpfilosofie heeft, of slechts een driftig bankswitchende 8-bit processor. Dat de beschikbare software voor de laatste goedkoper en groter in hoeveelheid is laten we dan wel buiten beschouwing.

JAN BREDENBEEK

DIEPENDAAELSELAAN 255

1214 KD HILVERSUM

NASCHRIFT:

Het artikel "QUILL VERSUS TASWORD" deed mij goed. Ik ben blij dat de QL eindelijk doet wat'ie van meet af aan had moeten doen. Dat ik nooit achter de QL zou hebben gezeten is niet waar!

Ik heb echter niet achter de verbeterde versie gezeten.

De bezwaren van Breedenbeek tegen Tasword heb ik overwonnen:

zwart scherm met groene letters, monitor, diskdrive (Thurnall, zie elders in dit nummer). Voor de tekens heb ik op verschillende wijzen een goed werkende oplossing gevonden, de meeste tekens zie ik ook op het scherm: ÿ, f, ü, en é inclusief.

Wat de BASIC en de andere mogelijkheden betreft kan ik alleen maar eerbiedig het hoofd buigen voor Breedenbeeks gezag!

SvdL

Voor hen die met de Spectrum aan telecommunicatie willen gaan doen heb ik een eenvoudig RS232-modeminterface ontworpen, dat ik samen met Jack Lelieveld heb uitgewerkt en getest. Het interface is voor ongeveer f 15,- aan onderdelen na te bouwen en ieder die wel eens een soldeerbout heeft vastgehouden zou het aan de praat moeten kunnen krijgen. Er hoeft bovendien niet perse een printje ontworpen te worden, een dubbelzijdige banenprint voldoet al. De schakeling bevat het minimum aan onderdelen dat nodig is om een modem aan te sluiten op de Spectrum. De software is bij Rik Koevoets verkrijgbaar door f 30,- over te maken op postgiro 3749935 onder vermelding van "terminalprogramma zelfbouwinterface". Desgewenst is er ook een versie voor het werken via Interface 1.

Onder de kreet "telecommunicatie" kan de computerhobbyist werken met databanken als VIDITEL verstaan, de zogeheten "VIEUWDATA SYSTEMS", of de "BULLETIN BOARD SYSTEMS", zoals bijvoorbeeld het FIDONET van de HCC. Er bestaan tussen deze twee een paar technische en een paar gebruikersverschillen.

De viewdatasystemen sturen steeds volle pagina's van 24 regels, met 40 tekens per regel. U kunt dan de informatie lezen en de volgende pagina laten komen, of een keuze maken uit de geboden alternatieven. Technisch worden deze systemen gekenmerkt doordat ze gebruik maken van ontvangst-/zendsnelheden van 1200/75 baud. Dat wil zeggen, ontvangen met 1200 baud, ofwel 1200 bits per seconde, dus 120 tekens per seconde bij 10 bits per teken. Die 10 bits zijn als volgt samengesteld: 1 startbit, 7 databits, 1 pariteitsbit en 1 stopbit. Het verzenden gebeurt met 75 baud (7,5 tekens per seconde). Dat de ontvangstsnelheid 16 maal zo groot is als de zendsnelheid, is bepalend voor het gebruik van het gehele viewdatasysteem. De gebruiker hoeft steeds met slechts enkele toetsaanslagen zijn keuze duidelijk te maken, waarna het systeem antwoordt met hele pagina's. U ontvangt veel meer informatie dan U zelf verstuurt. Als U als gebruiker een volle pagina wilt versturen, dan duurt dat 2 minuten en 8 seconden (een scherm bevat 24*40 dus 960 tekens en $960/7,5=128$). Het systeem stuurt U diezelfde pagina in 8 seconden, dat is 16 maal zo snel terug. Toch is communicatie met 1200/75 baud best werkbaar gebleken omdat er, als gezegd, relatief weinig informatie verzonden wordt.

Bulletinboards werken heel anders. Zij sturen U telkens slechts enkele regels tekst. Bijvoorbeeld:

BBS: Welkom bij dit Bulletin Board System.

U bent beller nummer 1517.

U hebt nog 19 minuten, wat is Uw naam?

U: Piet.

BBS: Goedenavond Piet, wat is Uw woonplaats?

U: Bergen op Zoom.

BBS: Dank U Piet. Er zijn momenteel geen berichten speciaal aan U gericht. Wel is er een algemeen bericht van de sysop aan alle gebruikers. Desgewenst kunt U

Als het scherm vol is schuiven alle regels naar boven waarbij de bovenste verdwijnt, oftewel scrollen, net als bij basiclistings. De regellengte varieert van BBS tot BBS. Er zijn er met 40, maar ook met 64, 72 of 80 tekens per regel, terwijl de wat betere BBS U de mogelijkheid biedt dat zelf in te stellen.

Uit het voorbeeld bleek al dat de ontvangen en verzonden informatie elkaar qua hoeveelheid niet veel ontlopen. Vandaar dat de bij viewdata gebruikte 1200/75 baud hierbij niet zo in trek is. Doorgaans zijn zend- en ontvangstsnelheid gelijk: 300/300 baud.

Het hier bijgevoegde ontwerp van een RS232-modeminterface voor de Spectrum is bruikbaar in combinatie met elk RS232-modem. En er is keus te over. Bijvoorbeeld: WS2000, Teletron 1200, Protek, VICCOM, PHILIPS of het PTT-modem. Het laatste is helaas evenals het bekende VTX5000 niet omschakelbaar naar andere snelheden dan 1200/75 baud.

De software, het terminalprogramma waar ik het al over had, kan goed gecombineerd worden met elk van deze modems en biedt de mogelijkheid tal van dingen, zoals de baudrate, in te stellen via een menu. Iedere baudrate is mogelijk, maar bepalend hiervoor is de computer waarmee U in contact wenst te komen. Werkt die op 300 baud, dan zult U eerst Uw modem daarop in moeten stellen en vervolgens Uw programma. Dan pas kunt U bellen en contact maken.

Naast de baudrate kunt U nog meer via het menu instellen, zoals de parity, een techniek voor controle op een juiste ontvangst.

Ook heeft het programma een keuze half-duplex/full-duplex. In de stand full-duplex zal de andere computer alle tekst die U intikt terugsturen, waarna die op Uw scherm verschijnt. Bij half-duplex verbindingen gebeurt dat niet en verschijnt er niets. In dat geval kunt U het programma in de stand half-duplex zetten, waarna het de verzonden tekst zelf op Uw scherm plaatst.

Om op zowel viewdata- als bulletinboardsystemen probleemloos te kunnen werken kan met de ENTER-toets de functie gewisseld worden. Of die toets geeft een hekje, #, voor viewdata, of CR voor BBS. Daarnaast kan de interpretatie van CR gewisseld worden tussen "wagen terug" (printpositie naar het begin van dezelfde regel), of CR+LF (printpositie naar het begin van de volgende regel).

Al deze instelmogelijkheden maken dat U met dit programma en een van de genoemde modems, via Uw Spectrum en zelfbouwinterface met zogoed als elk ander computersysteem in contact kunt komen. Bv:

VIEUWDATA:

VIDITEL	070-151515 of 020-318318
Computer Service Lijn (C.S.L.)	070-251589
HOBBYTEL	035-284822
VIATEL muurkrant	03402-26263 inloggen: 003395,3395
INFO SOURCE	015-611314
InfoCall	01899-15545

BULLETIN BOARDS:

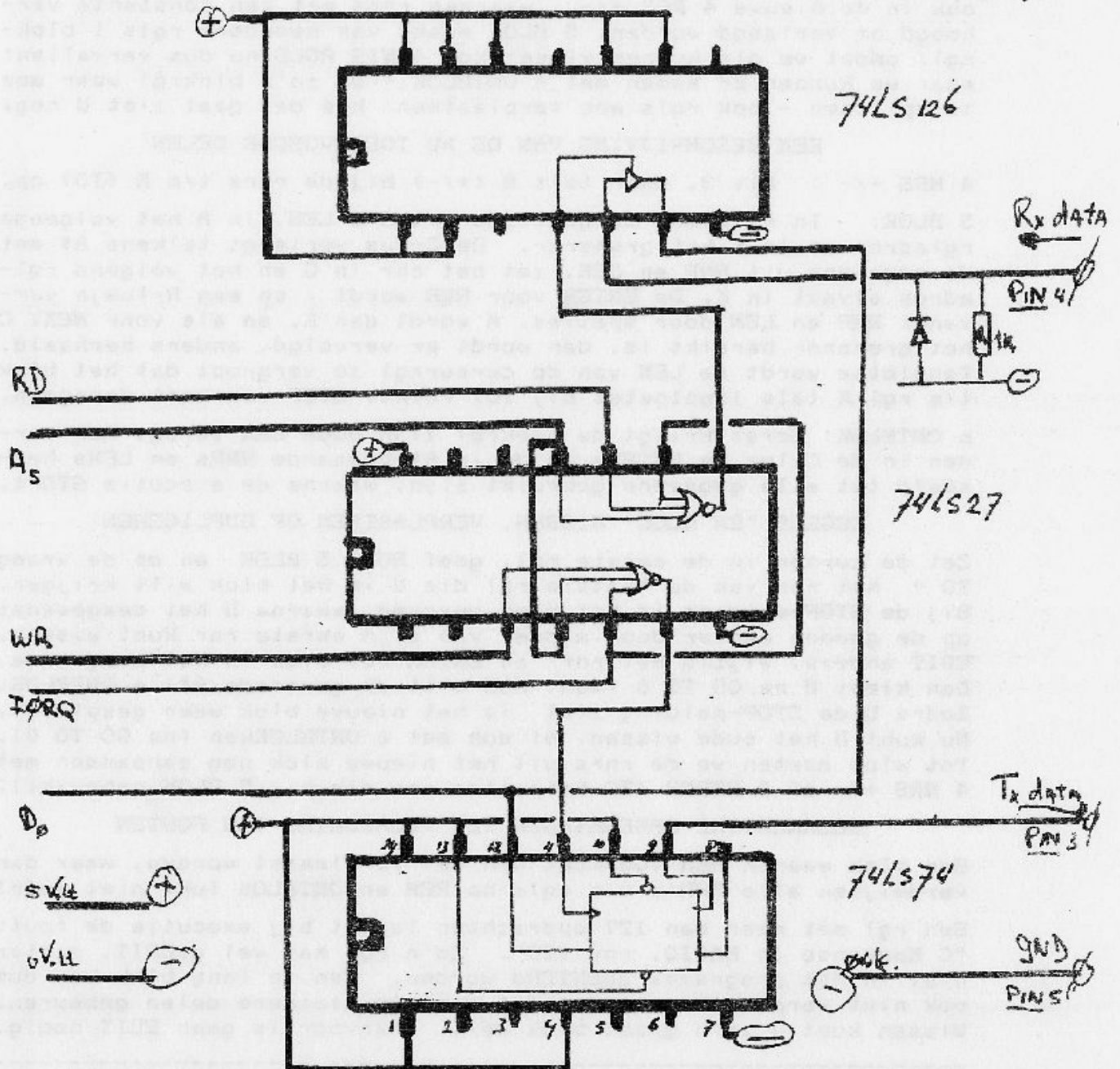
FIDONET (hoofd node)	08380-37156
BAMESTRA	02998-3603
EXPOLCON	01830-30406
Mailbox Vianen	03473-72260
Waterland	02990-26009

H J KOEVOETS - HEER DANIELSTRAAT 116a - 3073 DG ROTTERDAM

ZIE OOK 12-17

Spectrum
I/O Adres. 223

(+) 0V.H. PIN 7



SP

"TOOLKIT 7" IN BASIC VOOR REGELMANIPULATIE

SP

Hier dan de beloofde uitbreiding met VERPLAATSING VAN REGELS van mijn "TOOLKIT" uit IMPULS 09-31, waarnaar ik U moet verwijzen om dit artikel te kunnen begrijpen. De LISTing krijgt U hiernaast. In 3 HERNR is de C-lus zo gewijzigd, dat elk bestaand rnr gevonden wordt, ook als er een hoger rnr aan voorafgaat. Dit gebeurde ook in de nieuwe 4 NRS +/-, waarmee rnrs met een constante verhoogd of verlaagd worden. 5 BLOK maakt van meerdere rgls 1 blokrgl: omdat we die kunnen wissen kon 4 WIS RGLS nu dus vervallen: maar we kunnen er samen met 6 ONTBLOK - om zo'n blokrgl weer mee te splitsen - ook rgls mee verplaatsen: hoe dat gaat ziet U nog.

EEN BESCHRIJVING VAN DE NU TOEGEVOEGDE DELEN

4 NRS +/- : Als 3, maar telt B (+/-) bij de rnrs t/m K (TO) op.

5 BLOK: In A\$ komen de gegevens uit R's LEN, in A het volgende rgladres en in K het grensrnr. De C-lus verlengt telkens A\$ met de gegevens uit RNR en LEN, zet het rnr in C en het volgend rgladres alvast in K. De ENTER voor RNR wordt : en een R-lusje vangt RNR en LEN door spaties. A wordt dan K, en als voor NEXT C het grensrnr bereikt is, dan wordt er vervolgd, anders herhaald. Tenslotte wordt de LEN van de cursorrgl zo vergroot dat het blok t/m rgl K (als ingetoptst bij TO) reikt. STOP voorkomt doorgaan.

6 ONTBLOK: Eerst krijgt de blokrgl zijn oude LEN terug. Dan worden in de C-lus de ENTERs en de in A\$ bewaarde RNRs en LENS hersteld tot alle gegevens gebruikt zijn, waarna de executie STOPt.

REGELS "EN BLOC" WISSEN, VERPLAATSEN OF DUPLICEREN

Zet de cursor in de eerste rgl, geef RUN, 5 BLOK en op de vraag TO ? het rnr van de laatste rgl die U in het blok wilt krijgen. Bij de STOP-melding is het blok gevormd, waarna U het desgewenst op de gewone manier door middel van zijn eerste rnr kunt wissen. EDIT anders, wijzig het rnr en ENTER het blok in het programma. Dan kiest U na GO TO 0 (want RUN wist de gevormde A\$) 6 ONTBLOK. Zodra U de STOP-melding ziet is het nieuwe blok weer gesplitst. Nu kunt U het oude wissen, of ook met 6 ONTBLOKken (na GO TO 0). Tot slot moeten we de rnrs uit het nieuwe blok nog aanpassen met 4 NRS +/- of 3 HERNR (TO hetzelfde rnr als bij 5 BLOK gebruikt).

BELANGRIJKE OPMERKINGEN TER VOORKOMING VAN FOUTEN

Een blok waarin REM voorkomt kan wel verplaatst worden, maar dan verdwijnen alle FPR's uit rgls na REM en ONTBLOK lukt niet meer!

Een rgl met meer dan 127 opdrachten levert bij executie de fout: "C Nonsense in BASIC, rnr:128". Zo'n rgl kan wel geEDIT, echter niet in het programma geENTERd worden. Een te lang blok kan dus ook niet verplaatst worden: dat moet in kleinere delen gebeuren. Wissen kunt U zo'n groot blok wel, daarvoor is geen EDIT nodig.

:DE REGELMANIPULATIEMOGELIJKHEDEN UIT HET MENU VAN DEZE TOOLKIT:

- | | |
|-----------|--|
| 1 VERLENG | verlengt de cursorrgl met de volgende rgl en wist die |
| 2 SPLITS | voeg vooraf REM ccc (bv 009) toe vooraan in een rgl na opdracht nr ccc ontstaat een nieuwe rgl met rnr+1 geef achteraf EDIT ENTER + EDIT ENTER (FPR-herstel) |
| 3 HERNR | STEP ? TO ? STEPnummering na de cursor- t/m de TO-rgl |
| 4 NRS +/- | +/- ? TO ? verhoogt/verlaagt dezelfde rnrs als bij 3 |
| 5 BLOK | TO ? verlengt de cursor- t/m de TO-rgl, vormt A\$ tbv: |
| 6 ONTBLOK | splitst een met 5 gevormd blok na GO TO 0 (wegens A\$) |
| 7 STOP | wis de toolkit, alleen doen als rgl 0 de cursorrgl is |


```

10 DEF FN V(a)=a+4+PEEK (a+I+I)+X*PEEK (a+PI):
  DEF FN H(g)=INT (g/X): DEF FN L(g)=g-X*FN H(g):
  LET I=SGN PI: LET X=PEEK PI+I:
  LET A=VAL "PEEK 23635+X*PEEK 23636":
  LET R=VAL "PEEK 23625+X*PEEK 23626":
  FOR C=I TO R: LET A=FN V(A):
    LET C=X*PEEK A+PEEK (A+I): NEXT C: LET B=A+INT PI:
  INPUT "regel ": (R), "adres ": (A) "'1 VERLENG", "5 BLOK",
    "2 SPLITS", "6 ONTBLOK", "3 HERNR", "7 STOP ", "4 NRS +/-", k
11 FOR k=k TO I:
  LET A=FN V(A): POKE A-I, CODE ":":
  FOR C=A+VAL "4" TO FN V(A)-I: POKE C-4, PEEK C: NEXT C:
  POKE B, FN H(C-B-I): POKE B-I, FN L(C-B-I):
  PRINT "Tik! EDIT ENTER": STOP : NEXT k
12 FOR k=k TO I+I:
  LET K=A+VAL "B":
  FOR C=I TO VAL (CHR$ PEEK (K-PI)+CHR$ PEEK (B+PI))+
    CHR$ PEEK (K-I)): POKE K-4, PEEK K:
    LET K=K+I: LET C=C-(PEEK K<>58): NEXT C:
  POKE K-PI-I, VAL "13":
  POKE K-PI, FN H(R+I): POKE K-I-I, FN L(R+I):
  LET A=FN V(A)-K-I: POKE K, FN H(A): POKE K-I, FN L(A):
  LET A=K-B-PI-I: POKE B, FN H(A): POKE B-I, FN L(A):
  STOP : NEXT k
13 FOR k=k TO PI:
  INPUT " STEP ": B, " TO ": K:
  FOR C=I TO I: LET A=FN V(A): LET C=X*PEEK A+PEEK (A+I)=K:
    LET R=R+B: POKE A, FN H(R): POKE A+I, FN L(R): NEXT C:
  STOP : NEXT k
14 FOR k=k TO PI+I:
  INPUT "+/- ": B, " TO ": K:
  FOR C=I TO I: LET A=FN V(A): LET R=X*PEEK A+PEEK (A+I)+B:
    POKE A, FN H(R): POKE A+I, FN L(R): LET C=R=K+B: NEXT C:
  STOP : NEXT k
15 FOR k=k TO VAL "5":
  LET A$=CHR$ PEEK (B-I)+CHR$ PEEK B:
  LET A=FN V(A): INPUT " TO ": K:
  FOR C=I TO K: LET A$=A$+CHR$ PEEK A+CHR$ PEEK (A+I)+
    CHR$ PEEK (A+I+I)+CHR$ PEEK A$(A+PI):
    LET C=X*PEEK A+PEEK (A+I): LET K=FN V(A): POKE A-I, 58:
    FOR R=A TO A+PI: POKE R, 32: NEXT R: LET A=K: NEXT C:
  POKE B, FN H(A-B-I): POKE B-I, FN L(A-B-I):
  STOP : NEXT k
16 FOR k=k TO PI+PI:
  POKE B-I, CODE A$: POKE B, CODE A$(I+I):
  FOR C=PI TO LEN A$ STEP VAL "4": LET A=FN V(A): POKE A-I, 13:
    FOR R=NOT I TO PI: POKE A+R, CODE A$(C+R): NEXT R: NEXT C:
  STOP : NEXT k
17 POKE A+I, I: PRINT "Tik! I ENTER": STOP

```

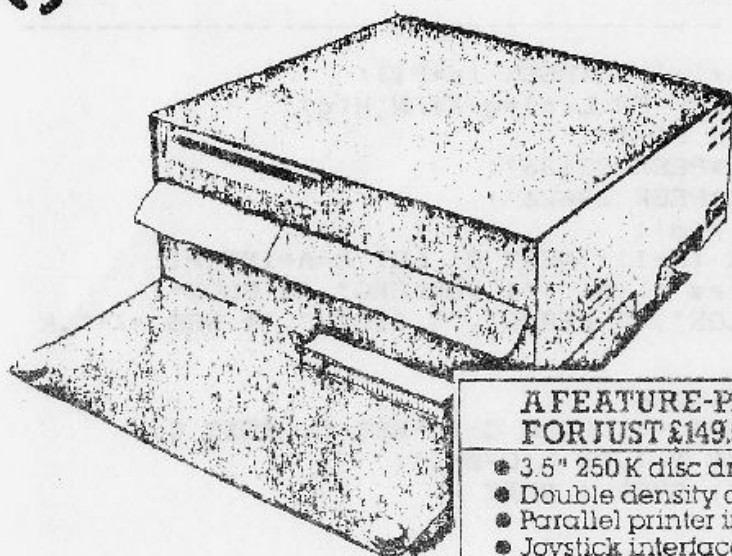
Toets deze regels in en voeg ze evenals de vorige keer samen met 1 VERLENG, nu echter in zeven keer, en POKE tenslotte het rnr 0. Deze toolkit 7 telt 1101 bytes, maar kan zonodig verkort worden.

juni 1986

Opus Discovery

nu voor

fl. 495,-



A FEATURE-PACKED SYSTEM FOR JUST £149.95 INCLUDING:

- 3.5" 250 K disc drive
- Double density disc interface
- Parallel printer interface
- Joystick interface
- Video monitor interface
- Peripheral through connector
- Built-in power supply
- Utilities on ROM including format and verify
- RAM disc facility
- Random access files fully supported
- Connections for second drive upgrade
- Comprehensive user manual

= 115 zwielen Uw
microdrive + interface
in. Tot max. fl. 175,-
Zo wordt Uw nieuwe
Opus nog goedkoper!

AMX MOUSE

FL. 275,-

= vernieuwd!

De AMX-Muis is nu nog beter! Met nog fraaiere styling en betere grip. Ook de bijgeleverde programma's zijn aangepast en verbeterd.

Voor al de "extended Basic" is zeer mooi: het stelt U in staat Mouse-Control in te bouwen in Uw eigen of andere programma's. Incl. Demo's, Icon-Designer en uitgebreide handleiding. Ook te gebruiken met Art Studio!

Data-Skip levert ook programma's voor AMX-Muis, o.a. Dbase AMX.

Topograaf AMX

f 19,-

f 29,-

- Multiface One, versie B6.2 f 195,-
- VTX-5000, modem + interface f 195,-
- Brother HR5, ser. 80-kol. printer. f 295,- !!!
- Fax 100, par. zeer solide printer. f 600,-
- Saga 3 Elite, een pracht keyboard. f 295,-
- Sinclair Q.L., compleet. 495,- !!!



BEL VOOR INLICHTINGEN, PRIJZEN
EN ADVIES



01820-20581

DATA-SKIP, GOUDA

L. Willemsteeg 10-12
2801 WC Gouda

Bestellingen d.m.v. ingesloten betaalkaart of vooruitbetaling op giro 47.27.958 t.n.v. Data-Skip, Gouda

SP

TEKENS VERHOGEN EN VERBREDEN

SP

Bestudering van het volgende programmaatje kan U op ideeën brengen hoe U, met BASIC, schermregels in de hoogte uit kunt rekken.

```
1 FOR x=0 TO 263-8*PEEK 23688: POKE 23677,x: POKE 23678,60:
  FOR y=167 TO 175: DRAW INVERSE POINT (x,y):0,3: NEXT y:
  NEXT x
2 INPUT "Uw tekst:" LINE t$: CLS : PRINT INK 7:t$: GO TO 1
```

SAVE deze regels met LINE 2, die om Uw tekst vraagt en die daarna met witte INK, dus onzichtbaar, bovenaan het scherm plaatst. Dan laat regel 1 de abscis x tot de laatste PRINT-positie lopen, zie IMPULS 08-51, waarna de PLOT-positie ingesteld wordt op x,60 door POKEN van de systeemvariabele COORDS. Daarvandaan wordt negen keer een lijntje van 3 pixels omhoog getrokken, zwart als de pixel in de onzichtbare tekst PAPER is en anders wit.

U kunt de uitrekking vergroten of verkleinen door de factor 3 te wijzigen. De hoogte op het scherm wordt bepaald door de ordinaat 60 die U ook kunt veranderen (pixelcoörd. handboek b.114/p.102). De tekst verschijnt INVERS. Om tekens als j ed. binnen het zwart te houden wordt vanaf 167, dus 1 pixel onder regel 1, "gelezen". INVERS ingetikte tekens worden onderstreept. Met NOT voor POINT krijgt U de tekst gewoon, waarbij U 167 door 168 kunt vervangen.

Tenslotte wil ik U nog laten zien hoe U ook de tekenbreedte kunt vergroten. Een verdubbeling krijgt U na RUN 3 bij het volgende:

```
1 POKE 23678,60: FOR Y=167 TO 175:
  DRAW INVERSE POINT (X,Y):0,3: NEXT Y: RETURN
2 FOR X=0 TO 263-8*PEEK 23688:
  POKE 23677,2*X: GO SUB 1: POKE 23677,2*X+1: GO SUB 1: NEXT X
3 INPUT "Uw tekst:" LINE T$: CLS : PRINT INK 7:T$: GO TO 2
```

Natuurlijk gaat het hier mis met te lange regels, maar het gaat slechts om het principe. Veel succes bij Uw eigen toepassingen.

SP

SIMPEL TEKENPROGRAMMA

SP

```
1 LET x=128: LET y=88: LET t=1
2 PLOT OVER 1:x,y: LET i$=INKEY$:
  PRINT #0:AT 1,0:"plaats: ";x:TAB 11:";";y:
  TAB 20:"tekenen: "; "aan" AND NOT t:"uit" AND t:
  PLOT INVERSE t:x,y: IF i$="t" THEN LET t=NOT t
3 LET x=x+((i$="e" OR i$="d" OR i$="c") AND x<255)
  -((i$="q" OR i$="a" OR i$="z") AND x):
  LET y=y+((i$="q" OR i$="w" OR i$="e") AND y<175)
  -((i$="z" OR i$="x" OR i$="c") AND y): GO TO 2
```

Gebruiksaanwijzing: Na RUN kunt U sturen met de toetsen rondom toets "S" en wisselen tussen "tekenen: aan/- uit" met toets "T".

De "status line" onderaan toont de coördinaten en de tekenstand. Bovendien markeert een knipperende pixel de schermpositie.

U kunt dit programma zelf verfraaien en uitbreiden met kleuren, een stand "sturen", SCREEN\$ SAVEN, CLS, STOP en wat U maar wilt.

SP

SPECTRUM-"BUGS"

SP

Onder de "insiders" van de lezers zal wel bekend zijn dat de Spectrum-ROM een aantal "bugs" bevat. Hoewel het effect van deze "bugs" meestal niet merkbaar is kunnen in sommige gevallen bij het programmeren in BASIC hierdoor onverwachte effecten optreden. Nu is het niet mijn bedoeling de Spectrum-ROM af te schilderen als "een programma vol met fouten", want fouten zitten er tenslotte in ieder door mensen gemaakt programma. Maar toch is het wel belangrijk om de "bugs" van de Spectrum te kennen om soms urenlang zoek en gekieker te vermijden. Daarom wil ik nu een opsomming van de tot nu toe bekende Spectrum bugs.

1. De "deling" fout.

Zoals u in hoofdstuk 24 van het handboek kunt lezen slaat de Spectrum niet-gehele getallen op in "floating-point" vorm. Dat betekent dat de getallen in wetenschappelijke notatie wordt opgeslagen met een mantisse en een exponent, waarbij in dit geval de mantisse numeriek tussen 0.5 en 1 ligt. De lengte van de mantisse is 32 bits waardoor de nauwkeurigheid 1 op 2^{32} bedraagt. Bij het rekenen met zulke getallen kijkt de Spectrum echter ook naar het 33e bit en soms naar het 34e bit indien de mantisse naar links moet worden geschoven om tussen 0.5 en 1 te komen. Dit is nodig voor de afronding. Indien er geen shift nodig was bepaalt het 33e bit van een rekenkundige uitkomst of er naar boven afgerond wordt, indien er wel een shift nodig was wordt dit door het 34e bit bepaald.

Het probleem is nu dat er in de ROM-routine die twee getallen op elkaar deelt een fout is gemaakt waardoor het 34e bit van de uitkomst altijd 0 is. Hierdoor wordt indien de mantisse van de deler groter is dan die van het deeltal, de uitkomst altijd naar beneden afgerond (in delen van 2^{-32} wel te verstaan)! Dit kunt u zien door in te tikken: `PRINT 1/2-0.5`. U krijgt dan als antwoord 2^{-32} , dus ongeveer $2.3E-10$.

2. De "-65536" fout.

Eveneens in hoofdstuk 24 van het handboek staat beschreven dat gehele getallen tussen -65535 en +65535 (+ of - $2^{16}-1$) als integers opgeslagen worden in 3 bytes. Een byte geeft hierbij het teken aan (0 voor positieve en 255 voor negatieve getallen), de andere twee bevatten het minst en het meest significante byte van de waarde van het getal (of de waarde van het getal plus 65536 indien het negatief is). Overigens beweert het handboek dat de laatstgenoemde twee bytes de waarde van het getal plus 131072 (2^{17}) bevatten indien het negatief is. Dit is onzin omdat de twee bytes dan een getal tussen 65537 en 131071 zouden moeten bevatten, hetgeen onmogelijk is (de waarde van het getal zelf ligt dan immers tussen -65535 en -1).

Omdat dit systeem in feite een 17-bits twee-complementsysteem is kunnen we getallen opslaan tussen -2^{16} en $2^{16}-1$, dus tussen -65536 en +65535. Het probleem vormt nu het getal -65536. Dit getal past wel in het "integer" formaat, maar kan als zodanig niet door de Spectrum verwerkt worden. Toch is het mogelijk dat dit getal in deze vorm kan ontstaan, namelijk door optellen/afrekken van twee kleinere gehele getallen en door de INT-functie. Bijvoorbeeld `PRINT -65000-536` geeft nu -1E38 omdat de printroutine het getal -65536 in "integer" formaat niet kan verwerken. `PRINT INT -65536` geeft -1. Bij `LET A=-65536:PRINT A` treden echter geen problemen op omdat het getal -65536 nu in normale "floating-point" notatie is opgeslagen.

3. De "CHR\$ 8" fout.

Bij het PRINTen van dit controlekarakter hoort de printpositie (de "cursor") een plaats terug te gaan. Dit gebeurt ook, zolang de cursor tenminste niet op het begin van de tweede schermregel staat! Want dan blijkt "backspacing", zoals dat heet, niet meer mogelijk te zijn. Als de cursor echter op de bovenste regel staat is het echter mogelijk om van het scherm af te "backspacen", met mogelijk een crash tot gevolg! Dit komt omdat de printroutine bij het testen of het begin van het scherm is bereikt tegen het verkeerde regelnummer test (regel 1 i.p.v. 0 op het scherm).

PRINT OVER 1:AT 14,11:"B":CHR\$ 8:"/" werkt goed, maar:

PRINT OVER 1:AT 0,31:"B":CHR\$ 8:"/" geeft geen B met een streepje erdoor.

4. De "CHR\$ 9" fout.

Het printen van dit karakter hoort de cursor een plaats naar rechts op te schuiven. Omdat de nieuwe positie van de cursor echter niet vastgelegd wordt in de systeemvariabelen (S-POSN en DF-CC) heeft PRINT CHR\$ 9 geen enkel effect.

5. De "CHR\$ 10 en 11" fout.

Gezien het voorgaande en de informatie die in het handboek staat bij de karakterset zou men verwachten dat deze twee karakters de cursor omhoog resp. omlaag zouden bewegen. In werkelijkheid zijn hiervoor echter geen voorzieningen getroffen in de ROM en deze twee karakters (en trouwens ook CHR\$ 12, DELETE) printen alleen een vraagteken uit.

6. De "scroll?" fout.

Indien u op de "scroll?" vraag antwoordt met CAPS SHIFT-2, CAPS SHIFT-7 of CAPS SHIFT-SYMBOL SHIFT verschijnt de laatst ingetikte BASIC-regel op het onderste schermgedeelte. Indien u daarna op een andere toets drukt verschijnen er in bepaalde gevallen vreemde karakters op het scherm gevolgd door een foutmelding. Bij het programmeren van de ROM is er kennelijk met deze situatie geen rekening gehouden.

Probeer het eens met: FOR I=1 TO 1000:PRINT "a":NEXT I.

7. De "programmacursor" fout.

In bepaalde gevallen is het mogelijk de programmacursor (tussen het regelnummer en de regel zelf) in een EDIT-regel te krijgen.

10 PRINT "Sinclair IMPULS"

11 + ENTER

EDIT geeft nu "10>PRINT "Sinclair IMPULS"" op het onderste scherm. Indien u weer ENTER tikt krijgt u het "syntax-error" vraagteken bij de ">".

8. De "CLOSE #" fout.

Indien u probeert een gesloten stroom tussen 4 en 15 te CLOSEn volgt er een crash (Vb. NEW gevolgd door CLOSE #4). Dit geldt echter alleen voor Spectrums zonder Interface-1. Als de Interface-1 wel is aangesloten heeft het CLOSEn van een gesloten stroom geen effect.

9. De "SAVE..DATA" fout.

Het is mogelijk een enkelvoudige string te SAVEn op cassette met SAVE..DATA. Bij het teruglezen echter wordt deze string wel ingeladen, maar kunnen we hem niet uitlezen.

10 CLEAR:LET a\$="Sinclair IMPULS"

20 SAVE "SI" DATA a\$()

30 CLEAR:PRINT "Spoel tape terug.."

40 LOAD "" DATA a\$()

50 PRINT a\$

Op regel 50 krijgt u "variable not found".

Overigens is deze fout bij de Microdrive verholpen. Daar krijgt u bij een poging om een enkelvoudige string te SAVEN de nogal duistere foutmelding "Nonsense in BASIC".

10. De "SCREEN\$" fout.

Als deze functie wordt gebruikt in een expressie na een operator (bijv. in `a$+SCREEN$ (0,0)`), levert deze expressie een verkeerde waarde op. Dit zal ik illustreren met het volgende voorbeeld.

```
10 LIST:LIST:LIST
20 LET a$=""
30 FOR L=0 TO 21
40 FOR C=0 TO 31
50 LET a$=a$+SCREEN$(L,C)
60 NEXT C
70 NEXT L
```

Indien u dit programma RUNt en PRINT a\$ geeft zou u de listing van het programma terug moeten krijgen, maar u krijgt slechts een string van twee karakters lang. Dit komt doordat het resultaat van de functie SCREEN\$ door een fout in de ROM tweemaal op de calculatorstack wordt gezet waardoor een eventueel tussenresultaat in een expressie wordt verduisterd.

Om deze fout te vermijden moet u de functie SCREEN\$ nooit gebruiken in combinatie met andere functies of operators, dus altijd alleen. Als u in het bovenstaande programma regel 50 verandert in:

```
50 LET b$=SCREEN$ (L,C):LET a$=a$+b$
```

werkt het wel goed.

Overigens is deze functie, en ook de functie ATTR, behept met nog een fout: ze accepteren argumenten die buiten het scherm vallen (bijv. `SCREEN$ (100,100)`)!

11. De "STR\$" fout.

Indien deze functie een argument toegevoerd krijgt met een absolute waarde tussen 0 en 1 (met uitzondering van 0 en 1 zelf) wordt een tussenresultaat verduisterd, op dezelfde manier als bij SCREEN\$. Bijv. `PRINT "2"+STR$ 0.5` zou 20.5 moeten uitprinten maar print 0.5 uit omdat er door STR\$ onbedoeld een lege string op de calculatorstack wordt gezet. Gebruik deze functie dan ook nooit in combinatie met andere functies en operators indien het argument een niet-geheel getal tussen -1 en +1 kan zijn.

Verder ontstaat bij een foutmelding veroorzaakt door de VAL-functie een vreemd effect indien de Interface-1 is aangesloten: bij een "Nonsense in BASIC"-melding veroorzaakt door VAL geeft de Spectrum altijd aan dat deze op regel 0 ontstond, ook al ontstond deze in een andere regel.

`10 PRINT VAL ""` geeft zonder Interface 1 de melding: Nonsense in BASIC, 10:1 maar met Interface 1: Nonsense in BASIC, 0:1. Dit komt doordat het argument van VAL gedurende de uitvoering van het programma op syntax wordt getest. In de Interface-1 ROM wordt dan echter bij een foutmelding het regelnummer op 0 gezet hetgeen tot bovengenoemd effect leidt.

Tot zover dan de "Spectrum-bugs" voor zover deze in de 16K ROM zitten. Misschien ga ik een volgende keer nog in op de bugs die in de 8K Interface-1 ROM zitten. Deze bugs waren voor Sinclair zelfs de reden om een nieuwe versie van de Interface-1 ROM uit te brengen. Dit in tegenstelling tot de "oude" 16K ROM, welke het ondanks bovengenoemde bugs al 4 jaar heeft volgehouden zonder een enkele wijziging...

----- JAN BREDENBEEK -----

 OVERZICHT VAN ARTIKELEN DIE TE KOOP ZIJN VIA DE STICHTING IMPULS

ART. -NR.	OMSCHRIJVING	PER STUK OF EVT. VERMELDE EENHEID	PRIJS FL.
1002	AVT-120, printer		1095.00
1003	AVT-180, printer		1295.00
1012	AVT-120 met par.centri.interface (MS Tronics)		1295.00
1013	AVT-180 met par.centri.interface (MS Tronics)		1495.00
1022	AVT-120 met buffer van 2 kbyte en RS232-ingang		1370.00
1023	AVT-180 met buffer van 2 kbyte en RS232-ingang		1570.00
1030	inktlint voor de AVT-80		28.00 +
1031	inktlint voor de AVT-100		30.00 +
1032	inktlint voor de AVT-120		30.00 +
1033	inktlint voor de AVT-180		30.00 +
1041	SCREENCOPYprogramma voor de AVT-100		27.50 +
1051	BUFFER van 2 kb met RS232-ing. vr 80 kol.printer		285.00 #
1052	8 kb-uitbreiding voor de bovenstaande buffer		95.00 #
2001	wit pinfeedpapier, 241 mm x 11''	2000 vel	72.50 #
2002	factuurofdrukpapier, 241 mm x 12''	250 vel	40.00 #
3001	MONITOR, monochroom amber of groen, 9 of 11 inch		375.00
3002	KLEURENMONITOR, 14 inch, RGB en AV		925.00
4001	printer-kabel, van interface 1 naar buffer		67.50 &
4002	printer-kabel, van interface 1 naar AVT-100		67.50 &
4003	aansluitkabel voor een kleurenmonitor		50.00 &
5002	c-15 computercassette		3.00 +
5003	c-60 cassette		3.50 +
5012	c-15 computercassettes	per 5 stuks	12.50 +
5012	c-60 cassettes	per 5 stuks	15.00 +
----	IMPULSOFTCASSETTE		10.00 +
6002	BASICODE 2, lees- en schrijfprogramma vd Spectrum		5.00 +
6011	cassettelabels	per 6 x 10 stuks	5.00 +
6021	opbergmap voor 6 cassettes		5.00 +
6031	opbergmap voor 11 of 12 inch pinfeedpapier		5.00 +
----	Sinclair Impuls	(2,3,4,6,7,8 of 9)	5.00 +

 Porti: +: 2.50 / &: 6.50 / #: 12.00 / anders: nihil



PORT
BETAALD
DEN HAAG

SINCLAIR
GEBRUIKERS
GROEP
Postbus 142
1740 AC SCHAGEN