BULLETIN

SINCLAIR GEBRUIKERSGROEP GRONINGEN/ASSEN



5e jaargang nummer 1 september '87

COLOFON

VOORZITTER:
Jan Dirk Burggraaf
Kluivingskampenweg 30
9761 BP Eelde
tel.05907-1697

SEKRETARIS:
Martin den Hollander
Numero Dertien 8
9644 TV Veendam
Tel. 05978-45474

PENNINGMEESTER:
S.E. Kroon
Oosterhoutstraat 96
9401 NK Assen
Tel. 05920-15912
Giro 5212298 t.n.v.
rekening SGG

VICE VOORZITTER/
PENNINGMEESTER:
J. van Alteren
De Grouw 6
9351 LP Leek
Tel. 05945-15678

VERHUUR:
C. van Krimpen
Koldakker 34
9407 BM Assen
Tel. 05920-70093

REDAKTIE BULLETIN:
Jan R. Kloosterman
Geert Valckeshof 28
9351 RX Leek
Tel. 05945-16328

Orm Heerkens
Burg. van Waninglaan 33
9351 LS Leek
Tel. 05945-12669

Het SGG-bulletin is een uitgave van de Sinclair Gebruikersgroep Groningen. Het bulletin verschijnt 10 keer per jaar.

Artikelen, listings of andere publikaties zijn voor verantwoording van de inzender.

De sluitingsdatum voor kopij wordt in elk bulletin vermeld.

Opgave voor lidmaatschap + abonnement op het bulletin á f. 17,50 per jaar bij de penningmeester (zie boven).

Overname van artikelen, illustraties en andere publikaties uitsluitend toegestaan met toestemming van de redaktie.

VAN DE REDAKTIE



Dag allemaal! Daar zijn we weer na een lange periode van diepe rust. Waarschijnlijk is iedereen zo langzamerhand terug uit Spanje, de Noordkaap, de Friese meren of de Drentse hei. Terug in de realiteit én naar de computer. We vermoeden tenminste dat niemand gedurende de lange natte dagen achter z'n Spectrum, QL of ZX-81 heeft zitten zweten. Gezien de binnengekomen kopij zal dat wel niet het geval geweest zijn. Sjongejonge, dat was schrikken toen we na de vakantie de deur bijna niet open konden krijgen van de enorme stapel kranten en reklamefolders. Maar kopij..... ho maar! Eén van de leden heeft zich tenminste veel moeite getroost om iets voor het bulletin in te leveren. Hij kwam helemaal vanuit Drente op de fiets(!) naar leek om een prachtig programma in te leveren. Bedankt Bert! In dit nummer zitten weer oude bekenden zoals Han van Abbe met Rekenen met de ZX-81, en Bert Westenberg met iets over de Opus. Op pagina vijf zit een nieuw gezicht, de column van de voorzitter wordt nu geschreven door onze nieuwe voorzitter, Jan Dirk Burggraaf. Verder hebben we in dit nummer:

- Een puzzeloplosser van Bert van der Zaag.
- De Spectrum + 3
- Hoeveel K over?
- Handige tip voor Microdrive?

Jan en Orm.

GRONINGEN

De volgende avond in Groningen is op donderdag 24 september 1987. Dat zou tijd worden ook want een heleboel mensen hebben elkaar al een hele poos niet meer gezien. Ook de avonden in het volgende seizoen zijn weer op het bekende adres:

School "de Wijert"
van Schendelstraat 1
Groningen

24 september zal er in Groningen een demonstratie gegeven worden door Micha Vesper van het ontvangen van weerstations en fax met behulp van de Spectrum.

Noteer ook maar vast de volgende avond in je agenda, deze is op 20 oktober. Voor november is dat op de 26e.

ASSEN

In Assen zijn er in september geen gebruikersavonden. Er zijn echter wel open dagen in het gebouw van de buurtvereniging waar de gebruikersavonden altijd worden gehouden, en de SGG is daar natuurlijk ook aanwezig. De open dagen zijn op 11 en 12 september op het volgende adres:

Het Markehuus
Walakker 11
Assen (Peelo - Wijk 7)

De gebruikersavonden zijn vastgesteld op elke 2e donderdag van de maand, dus de eerstvolgende avond is op 8 oktober.
In Assen zal Martin den Hollander uitleg geven over Masterfile en Tasdwars.

KOPIJ

Kopij voor het volgende nummer kan ingeleverd worden op de volgende gebruikersavond of vóór 24 september. Alle artikelen, illustraties, poke's, spelbesprekingen, e.d. graag op cassette, cartridge of 3½" (Opus) disk. Insturen die kopij, want het bulletin is er voor en door jullie!

REDAKTIE ZOEKT REDAKTIE !

De laatste tijd hebben wij (Jan en Orm) met veel plezier het bulletin verzorgd. We vinden het echter zo langzamerhand tijd worden voor een andere redaktie. Orm heeft inmiddels een andere computer gekocht, en Jan is dat binnen korte tijd ook van plan.

Een redaktie die niet volledig achter het gebruik van een Sinclair computer staat, of er zelfs geen meer heeft kan volgens ons niet goed functioneren.

Daarom hebben we besloten om op vrij korte termijn te stoppen met het verzorgen van het bulletin. Als niemand de redaktie van ons overneemt, betekent dat dat het bulletin niet meer uitgegeven wordt. Wij zouden dat erg jammer vinden, maar onze deadline is bereikt.

Met alle plezier willen we de nieuwe redaktie inwerken en op weg helpen. Een ieder die bereid is om de redaktie over te nemen wordt verzocht kontakt op te nemen met Jan Kloosterman of Orm Heerkens. De adressen en telefoonnummers vindt je in het colofon.

We hopen dat er een reaktie komt, want het zou zonde zijn als een bulletin als het onze zou ophouden te bestaan.

De redaktie, Jan en Orm.

VAN DE VOORZITTER



Een woord van de voorzitter. Nou daar zit je dan achter je computer om voor de eerste keer een stukje neer te zetten dat door de leden gelezen wordt (of niet!).
Allereerst even mezelf voorstellen.
Jan Dirk Burggraaf 43 jaar geleden op deze aardbol neergezet met momenteel als hobby's compjoeteren en als ik te weinig frisse lucht krijg volgens anderen dan ga ik het water op met de zeilplank.
Deze zomer vanwege het prachtige computerweer heb ik meer achter het toetsenbord gezeten dan dat er van het prachtige zomerweer genoten kon worden. Helaas zal bij velen iets in diezelfde richting ondernomen zijn.

Plannen voor het komende winterseizoen voor de club. Ja ze zijn er en met z'n allen kunnen we er zeker weer wat goeds van maken.

In ieder geval vinden we als bestuur dat jullie als leden wel geluiden mogen laten horen of de koers die tot dusver uitgezet was gevolgd kan worden.
In oktober komt er een ledenvergadering. Dit zal voor het eigenlijke computeren op de gebruikersavond gehouden worden.
Het is niet de bedoeling om dit als een officiële gebeurtenis de revue te laten passeren. Echter een kascommissie zal er aangesteld moeten worden (twee leden). Verder willen we de mogelijkheid onderzoeken of onze vereniging een officiële status moet krijgen of dat we zo door moeten gaan. En zo zijn er nog wel een aantal zaken waar we dan van gedachten over willen wisselen.

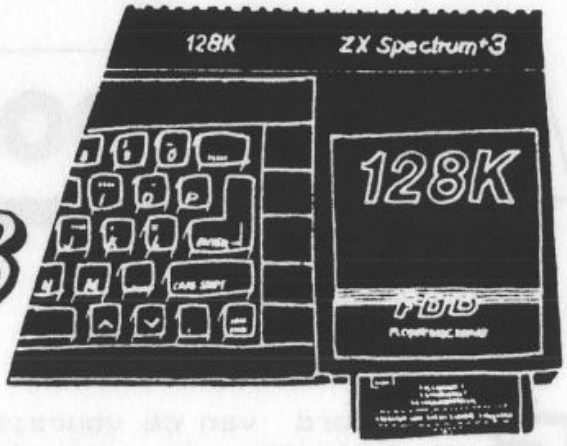
Tot slot heb ik nog een vraag aan de technische leden onder ons. Van mijn computer doen enkele groepen toetsen het soms niet meer en het spontaan opnieuw formatteren van diskettes is ook een verschijnsel.
Na onderzoek bleek dat het toetsenbord goed was, de klachten zouden in slechte verbindingen zitten. Wie helpt mij?

Dit was het voor dit ogenblik. Volgende keer hopenlijk wat gerichter.

J.D.Burggraaf

EINDELIJK. . . .

DE PLUS 3



Eindelijk horen we dan iets van de Spectrum + 3. Na de lancering van de + 2, waarvan Amstrad beweert er een half miljoen van te hebben verkocht, is ie er dan. Kompleet met een echte diskdrive(?). Laten we eens zien wat er allemaal met het Speccie is gebeurd.

Het apparaat is Spectrumzwart met een regenboogstreepje op de drive. Op de plaats waar bij de +2 de datarecorder zat prijkt nu de 3" diskdrive. Deze rare maat disks zijn de standaard van Amstrad. Op deze disks kun je 178K per kant kwijt en dat is niet veel als je de nieuwe generatie diskdrives bekijkt (bijvoorbeeld Betadisk). Aan de achterkant van de + 3 zit een connector voor een tweede drive. Daar zit ook en parallel printerpoort, MIDI-interface, monitor- en tv aansluiting, cassette-aansluiting en de edge-connector. Binnenin zit 64K voor de Disk Operating System (+3 Dos), 48K originele Spectrum ROM en 128K ROM.

De +3DOS is geschreven door Locomotive Software, een firma die al heel lang samenwerkt met Amstrad. Het zal dus voor velen geen verrassing zijn dat de +3DOS veel overeenkomsten vertoond met de DOS voor de CPC464 en de 6128 van Amstrad. De +3DOS heeft trouwens meer mogelijkheden in machinetaal dan in BASIC. Vanuit BASIC kun je bijvoorbeeld geen file OPENEN naar disc, terwijl dat in machinetaal wèl kan. De drive zelf is ietwat aan de langzame kant, zeker als je hem vergelijkt met andere disk-interfaces zoals de Disciple of Betadisk.

LOAD, SAVE en INTERFACE 1

Het laden en saven van de +3 is een verhaal op zich. Alle Tape-commando's werken op de diskdrive. Maar kun je dan ook programma's op tape laden? hoor ik vragen. Jawel, dat is heel simpel opgelost. Het zit 'm allemaal in de zogenaamde device namen. Bij de +3 heb je voor het laden of saven vier mogelijkheden: tape, disk 1 (de ingebouwde), disk 2 (die kun je erbij kopen) of Ramdisk. Voor de +3 heten ze respektievelijk T:, A:, B: en M:, en om het goede stuk hardware te kiezen zet je de goede letter bij het commando. Om dus een programma van tape te kunnen laden type je dan Load "T:PROGRAMMA", en om naar disk 1 te kunnen saven type je Save "A:DINGETJE".

Verder zijn alle Spectrum Interface 1 keywords zoals; MOVE, FORMAT, ERASE en CAT te gebruiken, maar..... de +3 is NIET Interface 1 en Microdrive compatible (wie verzint daar nou eens een mooi Nederlands woord voor). Interface 1 past er nog wel achterop maar werkt niet meer. Amstrad, die sowieso al niet zo weg was van de Microdrives, heeft de rechten van dit handige

apparaatje bij Sinclair gelaten bij de overname. Inklusief de code voor de Interface 1 ROM. Dit betekend dat (voor Amstrad) de Microdrive nu definitief verleden tijd is.

PRINTERS

Er is veel aandacht besteed aan het printergebeuren. Naast de +2 seriëele poort is er nu ook een parallel interface. Alle print-keywords werken zoals ze op alle Spectrums doen, alleen de COPY is enigzins veranderd. De gewone COPY geeft nog steeds een gewone puntje voor puntje screendump, maar COPY EXP geeft een prachtige greyscale dump (is ook al geen fatsoenlijk Nederlands woord voor). Ook is er een COPY EXP INVERSE commando, die een negatief scherm print. Veel screens zien er dan veel beter uit en het kan ook veel lint schelen.



DE +3 BINNENIN

Het binnenste van de +3 ziet er anders uit dan we van een Spectrum gewend zijn. Zelfs met het disk-interface zitter er minder chips in dan in de 128 of de +2. In plaats van 16 chips met de 128K RAM zitten er maar 4 (vier) in. Er zijn twee 32K ROMs met de software, de RS232 chips, de geluidschip, de TV beeld-, en de TV-geluidschip. Het videocircuit en de chips die alles met elkaar verbinden zitten samen in een kleïn chipje. Deze behandeld ook nog onder andere de joystick- en de printerpoort. De chip is een geheel nieuw ontwerp en het is nu nog niet te zeggen of ie compatible is met alle "oude" hardware, maar hij schijnt te werken met alle spellen voor de +2.
De BASIC voor de +3 is trouwens klakkeloos overgenomen van de +2, zelfs met alle foutjes die daar nog in zaten.

TOEKOMST?

Amstrad heeft veel kritiek gehad op de +2, en ze hebben er een beetje van geleerd. Het geluid is iets beter geworden en het beeld ook. Er kan weer een cassetterecorder worden aangesloten (zij het met van die kleine walkman plugjes). Maar er zitten nog steeds van die rare eigen standaard joystick aansluitingen op.
En dan de prijs! In Engeland 250 pond. Ruw omgerekend ongeveer duizend gulden! Het is "maar" een Spectrum met een diskdrive, en in deze tijd met alle goedkope Amiga's en Atari's is duizend gulden veel te duur!

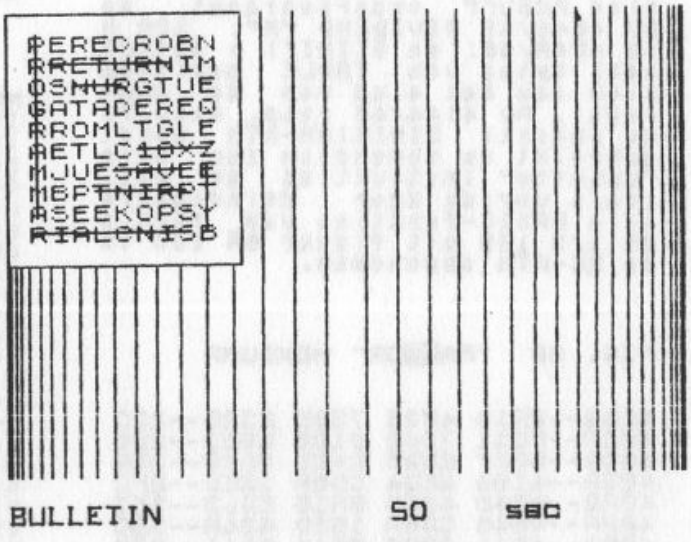
Jan R. Kloosterman.

WOORDZOEKER

Wel eens een woordzoeker gemaakt? Zo'n ingewikkelde puzzel waarbij je in een brij van letters horizontaal, vertikaal of diagonaal woorden moet zoeken? Ze staan ook veel in (buitenlandse) computerbladen waarbij je dan fantastische prijzen kunt winnen. Dan kun je vanaf nu je Spectrum het werk laten doen. Want we kregen van Bert van der Zaag een prachtig programma die dat helemaal zelf doet. Het enige dat je nu nog zelf moet doen is het onderstaande programma intoetsen en de (demonstratie)data te veranderen. Het letterveld mag zo groot zijn als het beeldscherm toelaat. Het programma geeft ook nog de tijd aan die de Spectrum er over doet om de puzzel op te lossen. Veel puzzelplezier ermee!

```
  10 BORDER 4: CLS
  20 FOR N=0 TO 19: PLOT N*(N/3),0: DRAW 0,175: PLOT 255-N*(N/3)
,0: DRAW 0,175: NEXT N
  30 LET R=10: LET K=9: LET W=15
  35 REM R en K is grootte puzzel. W is aantal woorden.
  40 DIM A$(R,K)
  50 PRINT AT 0,0;" "+A$(1)+" ";AT R+1,0;" "+A$(1)+" "
  60 FOR A=1 TO R
  70 READ A$(A, TO )
  80 PRINT AT A,0;" "+A$(A)+" "
  90 NEXT A
 100 PLOT 0,175: DRAW K*8+16,0: DRAW 0,-R*8-16: DRAW -K*8-16,0:
DRAW 0,R*8+16
 110 PRINT #1;AT 0,29;"sec"
 120 POKE 23674,0: POKE 23673,0: POKE 23672,0
 130 DEF FN T()=(65536*PEEK 23674+256*PEEK 23673+PEEK 23672)/50
 140 LET B$=" "
 150 FOR A=1 TO W: PRINT #1;AT 0,0; OVER 1;B$
 160 READ B$: PRINT #1;AT 0,0;B$
 170 FOR Y=1 TO R
 180 FOR X=1 TO K
 190 IF B$(1 TO 1)=A$(Y,X) THEN GO TO 240
 200 NEXT X
 210 PRINT #1;AT 0,25;INT (FN T())
 220 NEXT Y
 230 GO TO 170
 240 IF X+LEN B$>K+1 THEN GO TO 280
 250 FOR N=X+1 TO K+1
 260 IF LEN B$<N-X+1 THEN PLOT 8+(X-1)*8,164-(Y-1)*8: DRAW LEN B
$*8-3,0: NEXT A: GO TO 610
 270 IF B$(N-X+1 TO N-X+1)=A$(Y,N) THEN NEXT N
 280 IF X+LEN B$>K+1 THEN GO TO 330
 290 IF Y+LEN B$>R+1 THEN GO TO 330
 300 FOR N=X+1 TO K+1
 310 IF LEN B$<N-X+1 THEN PLOT 8+(X-1)*8,167-(Y-1): DRAW LEN B$*
8-3,-LEN B$*8+3: NEXT A: GO TO 610
 320 IF B$(N-X+1 TO N-X+1)=A$(Y+N-X+1-1,N) THEN NEXT N
 330 IF Y+LEN B$>R+1 THEN GO TO 370
 340 FOR N=Y+1 TO R+1
 350 IF LEN B$<N-Y+1 THEN PLOT 11+(X-1)*8,167-(Y-1)*8: DRAW 0,-L
EN B$*8+3: NEXT A: GO TO 610
 360 IF B$(N-Y+1 TO N-Y+1)=A$(N,X) THEN NEXT N
 370 IF X-LEN B$<0 THEN GO TO 420
 380 IF Y+LEN B$>R+1 THEN GO TO 420
```

```
 390 FOR N=X-1 TO 0 STEP -1
 400 IF LEN B$<X-N+1 THEN PLOT 14+(X-1)*8,167-(Y-1)*8: DRAW -LEN
 B$*8+2,-LEN B$*8+2: NEXT N: NEXT A: GO TO 610
 410 IF B$(X-N+1 TO X-N+1)=A$(Y+X-N+1-1,N) THEN NEXT N
 420 IF X-LEN B$<0 THEN GO TO 460
 430 FOR N=X-1 TO 0 STEP -1
 440 IF LEN B$<X-N+1 THEN PLOT 11+(X-1)*8,164-(Y-1)*8: DRAW -LEN
 B$*8+6,0: NEXT A: GO TO 610
 450 IF B$(X-N+1 TO X-N+1)=A$(Y,N) THEN NEXT N
 460 IF X-LEN B$<0 THEN GO TO 510
 470 IF Y-LEN B$<0 THEN GO TO 510
 480 FOR N=X-1 TO 0 STEP -1
 490 IF LEN B$<X-N+1 THEN PLOT 14+(X-1)*8,161-(Y-1)*8: DRAW -LEN
 B$*8+2,LEN B$*8-2: NEXT A: GO TO 610
 500 IF B$(X-N+1 TO X-N+1)=A$(Y+N-X+1-1,N) THEN NEXT N
 510 IF Y-LEN B$<0 THEN GO TO 550
 520 FOR N=Y-1 TO 0 STEP -1
 530 IF LEN B$<Y-N+1 THEN PLOT 11+(X-1)*8,164-(Y-1)*8: DRAW 0,LE
N B$*8-6: NEXT A: GO TO 610
 540 IF B$(Y-N+1 TO Y-N+1)=A$(N,X) THEN NEXT N
 550 IF X+LEN B$>K+1 THEN NEXT X: NEXT Y
 560 IF Y-LEN B$<0 THEN NEXT X: NEXT Y
 570 FOR N=X+1 TO K+1
 580 IF LEN B$<N-X+1 THEN PLOT 11+(X-1)*8,164-(Y-1)*8: DRAW LEN
B$*8-6,LEN B$*8-6: NEXT A: GO TO 610
 590 IF B$(N-X+1 TO N-X+1)=A$(Y+X-N+1-1,N) THEN NEXT N
 600 GO TO 200
 610 LET T=FN T()
 620 LET M=INT (T/60)
 630 LET S=INT (T-M*60)
 640 PRINT #1;AT 0,0;M;" Min. ";S;" Sec.",,
 650 PAUSE 0
 660 STOP
 999 REM puzzel
1000 DATA "PEREDROBN"
1010 DATA "RRETURNIM"
1020 DATA "OSNURGTUE"
1030 DATA "GATADEREQ"
1040 DATA "RROMLTGLE"
1050 DATA "AETLC18XZ"
1060 DATA "MJUESAVEE"
1070 DATA "MBPTNIRPT"
1080 DATA "ASEEKOPSY"
1090 DATA "RIALCNISB"
1099 REM woorden
1100 DATA "RUN"
1110 DATA "RETURN"
1120 DATA "SAVE"
2000 DATA "BYTE"
2010 DATA "ZX81"
2020 DATA "SINCLAIR"
2030 DATA "PRINT"
2040 DATA "PROGRAMMA"
2050 DATA "BULLETIN"
2060 DATA "SPECTRUM"
2070 DATA "POKE"
2080 DATA "BORDER"
2090 DATA "QL"
2100 DATA "DATA"
2110 DATA "ROM"
```



SNEL & PRECIES REKENEN MET DE ZX 81 (8)

Nu dus de laatste en snelste versie van
ONTBINDING IN PRIEMFACTOREN
genaamd FA■■OR.
Het sterk gereduceerde BASIC-programma staat in figuur 8A. De REM-regel bevat 247 bytes, waarvan 72 voor de eigenlijke rekenroutines op de adressen 40B3 t/m 40FA. De regels 10, 20 en 30 zijn ook "vertaalbaar" in MC maar dat brengt geen tijdwinst en wel komplikaties.

FIG. 8A "FA■■OR" BASIC-PROGRAM

```
 1 REM E(RNDY?■74 UNPLOT )

10 INPUT N$
20 LET N=VAL N$
30 PRINT N$;"=";USR 16514
```

De "omweg" van N$ via de funktie VAL naar N is gehandhaafd omdat daardoor enerzijds het eenvoudige werken met de FP-representatie mogelijk is. Anderzijds -als eerder besproken- is het PRINTen van N$ veel simpeler dan dat van N, omdat we rekening moeten houden met getallen van meer dan 8 digits. Daardoor is het direkt werken met INPUT N ook trager en gekompliceerder, zoals eveneens al behandeld is.

In figuur 8B worden de twee hexdumps gegeven van de MC-RTN van figuur 8C. Deze laatste RTN verschilt met die van FA■■OR op de volgende punten:
-de RTN wordt niet meer naar het RAM-gebied met beginadres 6490 hex verplaatst maar werkt direkt in de REM-regel; STORE is in het systeemvariabelen gebied PRBUFF ondergebracht; op AD 4046/49 DIVIDEND "N", $OR N op 404A/4B; de 5(INIT) + 43(REPeat) bytes van TABLE bevinden zich aan het eind van de REM-regel, AD 4144/48 resp. 4149/78
-de snelste DIVISION-RTN wordt gebruikt en bovendien zijn alle "onnutte" instruktles er uit, zoals vorige keer geindiceerd
-alle BASIC-funktles van regels 40 t/m 199 uit figuur 8A zijn in de MC-RTN opgenomen.

FIG. 8B "FA■■OR" HEXDUMP

```
4082--2A10 403E 73BE 2320--22C
408A--FC11 7640 0105 00ED--286
4092--B0EF E525 24E5 3000--3E2
409A--4100 2E34 CD8A 15ED--2FC
40A2--4348 40CD 8A15 ED43--367
40AA--4640 CD8A 15ED 434A--36C
40B2--4011 0000 0E44 0641--0EA
40BA--0A83 5F30 0314 2839--194
40C2--79EE 7820 020E 480C--263
40CA--D92A 4640 ED4B 4840--349
40D2--78D9 2100 0006 20D9--271
40DA--ED6A CB11 17D9 ED6A--47A
40E2--3807 ED52 3006 1918--1E5
40EA--03A7 ED52 10E9 7CB5--413
40F2--2831 2A4A 40ED 5230--27C
40FA--BDEF E530 3748 2E2D--398
4102--3040 4474 2403 3200--181
410A--1501 C5A4 0332 2C00--1E0
4112--08A0 0134 CDDB 15EF--389
411A--34CD DB15 EFE5 34CD--4C6
4122--DB15 CF42 4BCD 2015--34E
412A--EF2D E501 05C5 0234--302
4132--CDDB 153E 17D7 EFE5--4BD
413A--3032 000E 00DF 34C3--246
4142--9340 0201 0202 0402--0E0
414A--0402 0406 0206 0402--01E
4152--0406 0602 0604 0206--024
415A--0406 0804 0204 0204--022
4162--0806 0406 0204 0602--026
416A--0606 0402 0406 0206--024
4172--0402 0402 0A02 0A  --022
```

```
16514-2A10 403E 73BE 2320-  556
16522-FC11 7640 0105 00ED-  694
16530-B0EF E525 24E5 3000-  994
16538-4100 2E34 CD8A 15ED-  764
16546-4348 40CD 8A15 ED43-  871
16554-4640 CD8A 15ED 434A-  876
16562-4011 0000 0E44 0641-  234
16570-0A83 5F30 0314 2839-  404
16578-79EE 7820 020E 480C-  611
16586-D92A 4640 ED4B 4840-  841
16594-78D9 2100 0006 20D9-  625
16602-ED6A CB11 17D9 ED6A-1146
16610-3807 ED52 3006 1918-  485
16618-03A7 ED52 10E9 7CB5-1043
16626-2831 2A4A 40ED 5230-  636
16634-BDEF E530 3748 2E2D-  923
16642-3040 4474 2403 3200-  385
16650-1501 C5A4 0332 2C00-  480
16658-08A0 0134 CDDB 15EF-  905
16666-34CD DB15 EFE5 34CD-1222
16674-DB15 CF42 4BCD 2015-  845
16682-EF2D E501 05C5 0234-  770
16690-CDDB 153E 17D7 EFE5-1213
16698-3032 000E 00DF 34C3-  582
16706-9340 0201 0202 0402-  224
16714-0402 0406 0206 0402-  030
16722-0406 0602 0604 0206-  036
16730-0406 0804 0204 0204-  034
16738-0806 0406 0204 0602-  038
16746-0606 0402 0406 0206-  036
16754-0402 0402 0A02 0A  -  034
```

NEXT DIVISOR & DIVISION routines
In vergelijking met figuur 8C is in de ZEEFroutine het volgende gewijzigd: de 4 waarden die met de lokatie van TABLE samenhangen zijn aan de nieuwe situering aangepast. Verder wordt INC D benut om te testen of de "gevaarlijke" deler 65537(=1) is bereikt. Zie de bespreking van regel 110 in het voorgaande artikel.
In DIVISION worden voor MI-N nu de registers A en C' gebruikt in plaats van D'E'. Daardoor wordt het begin van de LOOP-RTN met 11 T-cycles bekort en deze vaakst doorlopen lus wezenlijk sneller. De niet-nodige instruktles AND A na ADD HL,DE en SCF voor DJNZ zijn verwijderd, waardoor 2 bytes en 4 T-cycles zijn bespaard.
Vo r de goede orde: de instruktie JR C, FORCE-SUBTRACT (was SAVE-CARRY) speelt pas een rol als a' een delers >2**15 in het geding zijn, maar dan is dit stukje routine onmisbaar. Probeer maar eens 17009085049=41243*41243 of 4294967071=65521*65551 met op AD 40E0 A7/00 in plaats van 3807.

FIG. 8C "FACTOR" MC-ROUTINE

```
                SEEK N AND TRFR TO MEM-5

16514           4082--2A1040      LD    HL,(4010) VARS
                4085--3E73        LD    A,73      "N" in VARS
SEEK            4087--BE          CP    (HL)
                4088--23          INC   HL
                4089--20FC        JR    NZ,4087   SEEK
                408B--117640      LD    DE,4076   MEM-5
                408E--010500      LD    BC,0005
                4091--EDB0        LDIR            TRFR

                FP-TO-32 BITS HEX; SQR N

FP-TO-HEX       4093--EF          RST   28;FPA:
                4094--E5                GET-MEM-5:N
                4095--25/24             SQR/INT
                4097--E5/30 00 41 00    N/65536:M
                409C--2E/34             N-MOD-M/END
                409E--CD8A15      CALL  158A      FP-TO-BC
                40A1--ED434840    LD    (4048),BC HI-N
                40A5--CD8A15      CALL  158A      FP-TO-BC
                40A8--ED434640    LD    (4046),BC LO-N
                40AC--CD8A15      CALL  158A      FP-TO-BC
                40AF--ED434A40    LD    (404A),BC INT SQR N

                NEXT-DIVISOR SIEVED FOR 2,3,5,7

                40B3--110000      LD    DE,0000   RES DIVISOR
                40B6--0E44        LD    C,44      LO-TABLE-INIT
NXT-DIV         40B8--0641        LD    B,41      HI-TABLE
                40BA--0A          LD    A,(BC)
                40BB--83          ADD   A,E
                40BC--5F          LD    E,A
                40BD--3003        JR    NC,40C2   NO-INC
TEST-END        40BF--14          INC   D         DE=DIV "D"
                40C0--2839        JR    Z,40FB    LAST-FACTOR
NO-INC          40C2--79          LD    A,C       LO-TABLE-END
                40C3--EE78        XOR   78        INC-POS
                40C5--2002        JR    NZ,40C9   LO-TABLE-REP -1
                40C7--0E48        LD    C,48      C=NXT POS in TABLE
INC-POS         40C9--0C          INC   C

                DIVISION                          XOR → NC

                40CA--D9          EXX
                40CB--2A4640      LD    HL,(4046) LO-N in H'L'
                40CE--ED4B4840    LD    BC,(4048)
                40D2--78          LD    A,B       HI-N in A,C'
                40D3--D9          EXX
                40D4--210000      LD    HL,0000   RES REMAINDER
                40D7--0620        LD    B,20

LOOP            40D9--D9          EXX
                40DA--ED6A        ADC   HL,HL
                40DC--CB11        RL    C
                40DE--17          RLA
                40DF--D9          EXX
                40E0--ED6A        ADC   HL,HL
                40E2--3807        JR    C,40EB    FORCE-SUBTR.
                40E4--ED52        SBC   HL,DE     TRIAL-SUBTRACT
                40E6--3006        JR    NC,40EE   CONT
                40E8--19          ADD   HL,DE     RESTORE HL
                40E9--1803        JR    40EE      CONT
FORCE-SUBTR.    40EB--A7          AND   A
                40EC--ED52        SBC   HL,DE
CONT            40EE--10E9        DJNZ  40D9      LOOP

                40F0--7C          LD    A,H       REMAINDER ZERO?
                40F1--B5          OR    L
                40F2--2831        JR    Z,4125    FACTOR in DE if yes

                TEST D>SQR N

                40F4--2A4A40      LD    HL,(404A) SQR N
                40F7--ED52        SBC   HL,DE     D <=SQR N?
                40F9--30BD        JR    NC,40B8   NXT-DIV if yes

                PRINT FINAL FACTOR  H=INT(N/100)
                                    R=N-100*H

LAST-FACTOR     40FB--EF          RST   28;FPA:
                40FC--E5/30 37 48       N/100:M
```

```
                4100--2E                N-MOD-M:R,H
                4101--2D                DUPL:H,H
                4102--30 40 44 74 24    STK 1E6
                4107--03/32             SUBTRCT/<0?
                4109--00 15             JP-TRUE ..      AD 411F
                410B--01/C6             XCH/MEM:R
                410D--A4/03             STK-10/SUBT
                410F--32/2C             <0?/NOT
                4111--00 08             JP-TRUE .       AD 411A
                4113--A0/01             STK-0/XCH
                4115--34                END-CALC

H>=1E6 & R<10   4116--CDDB15    CALL    15DB            PRINT H
                4119--EF        RST     28;FPA:
              •                 4111A--34               END-CALC
H>=1E6 & R>9.R<10 411B--CDDB15  CALL    15DB            PRINT H / Ø
                411E--EF        RST     28;FPA:
            ••  411F--E5/34             GET-MEM/END
                4121--CDDB15    CALL    15DB            PRINT N / R
H<1E6 / H>=1E6  4124--CF(42)    RST     08;

                N/D=N   +   PRINT D*   +   N=2? : JP

FACTOR          4125--42        LD      B,D             BC-DE="D"
                4126--4B        LD      C,E
                4127--CD2015    CALL    1520            STK-BC
                412A--EF        RST     28;FPA:
                412B--2D/E5             DUPL:D,D/N
                412D--01/05/C5          XCH/DIV/MEM     N/D-N
                4130--02/34             DELETE/END      FP-D

PRINT D*        4132--CDDB15    CALL    15DB            PRINT-FP
                4135--3E17      LD      A,17            "*"
                4137--D7        RST     10

N=2?            4138--EF        RST     28;FPA:
                4139--E5/30 32 00/0E    N/STK 2/=?
                413E--00 DF             JP-TRUE ..      AD 411F
                4140--34                END-CALC
                4141--C39340    JP      4093            FP-TO-HEX

                TABLE

TABLE-INIT      4144--          02 0102 0204- 011
TABLE-REP       4149--0204 0204 0602 0604- 030
                4151--0204 0606 0206 0402- 032
                4159--0604 0608 0402 0402- 036
                4161--0408 0604 0602 0406- 040
                4169--0206 0604 0204 0602- 032
                4171--0604 0204 020A 020A- 040  4178-TABLE-END
```

TEST D>SQR N routine
Deze routine is aangepast aan de nieuwe situering.

SEEK N AND TRFR TO MEM-5 routine
De variabele N staat in VARS op een plaats die afhankelijk is van de lengte van N$, daarom is een opzoek-routine nodig. Aangezien de CODE voor N 33 hex is, moet in VARS naar de byte met inhoud 73 hex worden gezocht. Daarna worden de hierop volgende vijf bytes met de FP-representatie van N overgebracht (TRansFeRred) naar de lokatie MEM-5: 4076/7A.

F(loating) P(oint) A(rithmetic)
De funktionele "vertaling" van BASIC-regels gebeurt -zoals al is vermeld- met de operationele subroutines van de CALCULATOR. Deze FPA-RTNS zijn weliswaar relatief traag, maar toch sneller dan de equivalenten in BASIC, waarvan ze de omzetting vormen. Dat zal bij de bespreking van de TIMING blijken.
Nog afgezien daarvan vormt die omzetting een goede gelegenheid beter vertrouwd te raken met het gebruiken van FPA.
Alvorens de diverse funkties te behandelen een aantal algemene opmerkingen:

- Ik gebruik de standaard-labels van de diverse FPA-instruktie, die de operationele subroutines van de CALCULATOR aktiveren, dit keer sleehts in beperkte mate en vaak bekort. Voor de duidelijkheid gebruik ik daarnaast in de toelichtingen bij de akties de letters van de grootheden uit figuur 6A.
- Als in een regel meer instruktie voorkomen zijn deze gescheiden door een '/' en dat gelat ook voor de toelichtingen; dit is dus GEEN DEELTEKEN!
- De FP-waarden op de CALCULATOR STACK, de ST(AC)K-values, worden in principe LIFO(last-in-first-out) verwerkt; men moet dus goed op de volgorde letten bij het op de STK plaatsen van de getallen.
- Een aantal instruktie "verbruikt" de laatste of de laatste twee 'values' en plaatst veelal weer een 'last value' op de STK. Een uitzondering vormt N-MOD-M, waarbij 2 waarden worden teruggeplaatst: eerst N-M*INT(N/M) en dan INT(N/M); de oorspronkelijke 'forelast value' N en dito 'last value' M worden verbruikt.
- De "comparison" operations,

CODES 08 t/m 0E en 11 t/m 16 vergelijken de 'forelast value' met de 'last value'
- De instructies voor aftrekken (03), delen(05) en machtsverheffen(06) gebruiken alledrie eerst de 'f(orelast)' waarde en dan de 'l(ast)', bepalen dus respectievelijk f-l, f/l en f**l. Zoals al gezegd verdwijnen f en l van de STK, die nu het resultaat bevat.
- Wil men een waarde die "verbruikt" wordt nog eens kunnen gebruiken dan moet die waarde eerst worden geDUPLiceerd(2D) of worden opgeslagen in een van de 6 MEM-lokaties. Met dat laatste moeten we voorzichtig zijn, daar die FP-geheugenplaatsen ook door bepaalde CALCULATOR subroutines worden gebruikt; MEM-5 is het "veiligst" te gebruiken.
- Ook PRINT-FP en FP-TO-BC "verbruiken" de 'last value' van de STK.
- Na CODE 30, STK-DATA, volgen 2 tot 5 bytes, afhankelijk van het getal dat op de STK moet worden geplaatst; aantal en waarden van deze bytes worden volgens een "reglement" afgeleid van de FP-representatie van 't betreffende getal.

FP-TO-32 BITS HEX; SQR N routine
Eerst wordt INT SQR N berekend en dan wordt N "vertaald" van de 5 bytes FP, opgeslagen in MEM-5, in 4 bytes hex met 65536 als modulus M.
De berekende grootheden worden geSTOREd op de reeds genoemde lokaties.

PRINT FINAL FACTOR routine
De formules voor H en R zijn nog eens vermeld, waarden die verkregen worden met M=100 als modulus. STK 100 is 30 37 48.
Na duplicatie wordt H verminderd met 1E6 en wordt getest of het resultaat <0 is. Indien 'true', konditie H<1E6 dus vervuld, springt de RTN 15 hex = 21 dec verder naar de label .. op adres 411F en beeindigt het programma met PRINT N. LET OP: bij de CALCULATOR instrukties met JP begint het tellen -anders dan bij de vergelijkbare instrukties met JR- met "0" op de plaats van de 'literal' die de spronggrootte bepaalt.
Als de test 'false' oplevert dus H>=1E6 is, gaan we door met het omwisselen van R en H en het opslaan van R in MEM-5. Vervolgens brengen we 10 op de STK(CODE A4) verminderen R met 10 en testen of dat resultaat NOT<0 is dus of R>=10 is. Indien 'true' wordt nu het programma voortgezet op adres 411A (label .): PRINT H en daarna PRINT R.
Als de tweede test ook 'false' oplevert wordt 0 met CODE A0 op de STK gebracht, R en 0 worden verwisseld en we verlaten de FPA-routine LAST FACTOR. Daarna worden achtereenvolgens H, 0 en R gePRINT.
In de marges naast de PRINT-subroutine staan de kondities en wat dan gePRINT wordt nog eens samengevat.

N/D=N + PRINT D* + N=2? : JP
Deze laatste subroutine wordt gebruikt als op adres 40F0 de REMAINDER ZERO is. We hebben dan een D(eler)=FACTOR gevonden, die op de STK gezet en geduplíceerd wordt. Na N van MEM-5 te hebben opgehaald, verwisselen we D en N, bepalen N/D (05 = DIVISION), slaan N=N/D op in MEM-5, verwijderen dat resultaat van de STK en houden D over als 'last value'.
Vervolgens wordt D* gePRINT.
Daarna volgt weer een FPA-RTN, waarin de "nieuwe" N uit MEM-5 wordt gehaald, 2 op de STK wordt gezet en getest wordt of N=2 is. Zo ja, dan PRINTen we die waarde -sprong naar label .. - waarmee het programma eindigt. Zo nee, dan volgt JP FP-TO-HEX en wordt gezocht naar een eventuele volgende FACTOR.

OPMERKING
Zoals toegelicht bij de bespreking van FACTOR gedraagt SQR N zich "raar" voor N > 4294967280.
In de CALCULATOR is dat gedrag echter anders dan op grond van figuur 7 zou worden verwacht: INT SQR N is dan namelijk 8000 hex en niet 0000.
Enerzijds betekent dit dat een korrektieterm, vergelijkbaar met (K=256) in de regels 45 en 50 van figuur 6A niet nodig is. Anderzijds zou de ontbinding tussen N = 4294967280 en N = 2**32 mis kunnen gaan. Dat gebeurt echter niet, omdat alle betreffende getallen ook delers <32768 (= 8000 hex) hebben behalve 4294967291. Dit is evenwel een PRIEMgetal en het gevolg is alleen dat de RTN delers >32768 voor dit bewuste getal niet onderzoekt. Fundamenteel fout maar zonder kwalijke gevolgen.
In mijn vorige artikel heb ik gesteld dat in regel 110 IF D=1 nodig is bij het onderzoek van getallen >65521**2. Die test, nu met INC D in de NEXT-DIVISOR-RTN, heeft geen betekenis voor het zoeven genoemde PRIEMgetal, aangezien gestopt wordt < 32768. Voor PRIEMgetallen < 4294967280 is INT SQR N echter 65535 en is de "dreiging" D=1 wel aanwezig, vandaar TEST-END in de juist genoemde RTN. Het "gevaarlijkste" PRIEMgetal is dan ook 4294967279

T I M I N G
In figuur 9 staan 19 voorbeelden van ontbindingen "genummerd" A t/m S, met de tijden voor:
FACTOR het verkorte BASIC-PG
FACTOR "Publ." als besproken de vorige keer
FACTOR "Snlr." hetzelfde PG met de snelste DIVISION-LOOP
FACTOR het nu behandelde PG

Twee opmerkingen vooraf:
- het verkorte BASIC-PG is sneller dan het oorspronkelijk juni 1985 door mij gepubliceerde dat herhaald is in deel (6); daarom zijn de eerder genoemde maximale rekentijden van ruim 7 minuten gereduceerd tot 6 min 41 sec
- een verdere versnelling van ca. 3% is mogelijk door in het PG van figuur 3 van regel 50 regel 5 te maken; daardoor komt de het vaakst gebruikte variabele D vooraan in VARS te staan en dat brengt deze tijdwinst.

FIG. 9 ONTBINDING IN PRIEMFACTOREN — T I M I N G in seconden

	N$	FACTOREN	FACTOR verkort	FA■TOR Publ.Snlr.	FA■TOR
A	12345	3*5*823	.9	1.6	1.6 .9
B	12347	PRIEM	1.1	.7	.7 .4
C	65536	2**16	2.3	7.1	7.1 3.7
D	12345678	2*3*3*47*14593	1.7	2.6	2.6 1.4
E	87654321	3*3*1997*4877	13.1	2.8	2.8 1.8
F	2230922870	2*3*5*7*11*13*17*19*23	1.6	4.4	4.4 2.3
G	987654321	3*3*17*17*379721	4.7	2.8	2.8 1.6
H	999999937	PRIEM	193	10.3	9.1 8.9
I	10742333401	PRIEM > 2**30	201	10.7	9.4 9.2
J	2147483647	PRIEM 2**31-1	283	14.6	13.1 12.7
K	2147483648	2**31	4.4	14.0	14.0 7.2
L	4293001469	1.PRIEM >65521**2	401	20.2	18.0 17.8
M	4294049777	65521*65537	401	20.7	18.5 17.9
N	4294966299	3*13*331*332711	4.4	2.3	2.3 1.4
O	4294967071	65521*65551	401	20.7	18.5 17.9
P	4294967279	laatste PRIEM SQR<>raar	401	20.2	18.0 17.8
Q	4294967291	grootste PRIEM < 2**32	401	20.2	18.0 9.2
R	4294967293	9341*464773	57	4.1	3.8 3.3
S	4294967296	2**32	4.6	14.4	14.4 7.4

In figuur 9 valt in de eerste plaats op, dat de voorbeelden C, F, K en S veruit het snelst worden ontbonden in BASIC! De betreffende 4 getallen bevatten steeds een groot aantal kleine factoren. En zo'n kleine factor kost in het BASIC-PG 1/7 sec per factor, in het snelste MC-PG 0.23 sec per factor.
Dit verschil in benodigde tijd is voornamelijk te danken aan de test IF D*D>N in regel 80 van figuur 3. Ik heb deze regel ook funktioneel geintegreerd in de MC-RTN van FA■TOR met tot 50% tijdwinst voor de voorbeelden met vele factoren, echter met 20% tijdverlies voor het -veel langer durende- onderzoek van PRIEMgetallen en getallen met grote factoren. Daarom is deze test NIET ingevoerd.

De tijdwinst van de MC-RTNS blijkt 't duidelijkst bij grote factoren: daar is FA■TOR tot 22-maal zo snel als het snelste BASIC-PG. Voorbeelden: H, I, J, L, M, O, P en Q.
Zoals hiervoor verklaard, wordt voorbeeld Q extra snel berekend door de beperking van de delers tot < 2**16. Anders zou de benodigde tijd bijna tweemaal zo groot zijn geweest.

FA■TOR gebruikt met de nu ingevoerde snellere DIVISION-LOOP voor een PRIEM en bij grote factoren tot 11% minder tijd, zie de tabel van figuur 9.

Tenslotte: FA■TOR brengt ten opzichte van FA■TOR-Snlr. nog tot bijna 50% tijdwinst bij getallen met vele factoren, echter begrijpelijkerwijs slechts enkele procenten bij getallen met grote delers of PRIEMgetallen.

DELING-PROGRAMMA TOT 2**32

Vorige keer heb ik geindiceerd hoe met de "volledige" DIVISION-RTN en het "afmaken" van het quotient een exact DELING-PROGRAMMA voor getallen tot 2**32 kan worden gemaakt. Omdat ik me realiseer dat dit voor velen nog hoofdbrekens kost, als "toegift" in figuur A het programma.
Na de eerdere bespreking van het eenvoudige DELING-PG van figuur 5 en het in dit artikel behandelde, zullen de bijschriften in figuur A voldoende toelichting vormen om het PG en de RTN te begrijpen.
LET OP:
■ Na adres 40CD EXX worden alle verdere instruktie's uitgevoerd met de 'exchange-registers'.
■ Na RST 08 moet een DEFB volgen die er voor zorgt dat na uitvoering van regel 6 doorgegaan wordt met regel 7; ED is zo'n CODE.
■ Regel 6 en 7 mogen niet worden gekombineerd.

FIG. A N/D=Q R= N<2**32 D<2**16

```
  1 REM E(RNDY?■74 UNPLOT )
  2 INPUT N$
  3 LET N=VAL N$
  4 INPUT D
  5 RAND D
  6 PRINT N$;"/";D;"=";USR 1651
4
  7 PRINT "  R=";USR 16553
```

HEXDUMP

```
4082--2A10 403E 738E 2320--22C
408A--FC11 7640 0105 00ED--2B6
4092--B0EF E530 0041 002E--323
409A--34CD 8A15 C5CD 8A15--3D1
40A2--C5D9 E1C1 78D9 ED5B--5D9
40AA--3240 2100 0006 20D9--192
40B2--ED6A CB11 17D9 ED6A--47A
40BA--3808 ED52 3007 19A7--276
40C2--1804 A7ED 5237 10E7--330
40CA--2250 40D9 ED6A CB11--3BE
40D2--1747 E5CD 2015 C1CD--3D3
40DA--2015 EF01 3000 4100--196
40E2--040F C530 3748 2E2D--1E2
40EA--3040 4474 2403 3200--161
40F2--1501 C5A4 0332 2C00--1E0
```

```
40FA--08A0 0134 CDDB 15EF--389
4102--34CD DB15 EFE5 34CD--4C6
410A--DB15 CFED 4B50 40C9--450

16514-2A10 403E 73BE 2320- 556
16522-FC11 7640 0105 00ED- 694
16530-B0EF E530 0041 002E- 803
16538-34CD 8A15 C5CD 8A15- 977
16546-C5D9 E1C1 78D9 ED5B-1497
16554-3240 2100 0006 20D9- 402
16562-ED6A CB11 17D9 ED6A-1146

16570-3808 ED52 3007 19A7- 630
16578-1804 A7ED 5237 10E7- 816
16586-2250 40D9 ED6A CB11- 958
16594-1747 E5CD 2015 C1CD- 979
16602-2015 EF01 3000 4100- 406
16610-040F C530 3748 2E2D- 482
16618-3040 4474 2403 3200- 385
16626-1501 C5A4 0332 2C00- 480
16634-08A0 0134 CDDB 15EF- 905
16642-34CD DB15 EFE5 34CD-1222
16650-DB15 CFED 4B50 40C9-1104
```

MC-ROUTINE

```
16514   4082--2A1040      LD    HL,(4010)  VARS
        4085--3E73        LD    A,73       DIVIDEND "N"
SEEK    4087--BE          CP    (HL)
        4088--23          INC   HL
        4089--20FC        JR    NZ,4087    SEEK
        408B--117640      LD    DE,4076    MEM-S
        408E--010500      LD    BC,0005
        4091--EDB0        LDIR             TRFR

        4093--EF          RST   28;FPA:
        4094--E5 30 00 41 00 2E 34         N-MOD-65536
        4098--CD8A15      CALL  158A       FP-TO-BC
        409E--C5          PUSH  BC         HI-N
        409F--CD8A15      CALL  158A       FP-TO-BC
        40A2--C5          PUSH  BC         LO-N
        40A3--D9          EXX
        40A4--E1          POP   HL         LO-N in H'L'
        40A5--C1          POP   BC         HI-N in A,C'
        40A6--78          LD    A,B
        40A7--D9          EXX

        40A8--ED5B3240    LD    DE,(4032)  DIVISOR "D"
        40AC--210000      LD    HL,0000    RES REMAINDER
        40AF--0620        LD    B,20

LOOP    40B1--D9          EXX
        40B2--ED6A        ADC   HL,HL
        40B4--CB11        RL    C
        40B6--17          RLA
        40B7--D9          EXX
        40B8--ED6A        ADC   HL,HL
        40BA--3808        JR    C,40C4     SAVE-CARRY
        40BC--ED52        SBC   HL,DE      TRIAL-SUBTRACT
        40BE--3007        JR    NC,40C7    NO-RESTORE
        40C0--19          ADD   HL,DE      RESTORE HL
        40C1--A7          AND   A
        40C2--1804        JR    40C8       CONT
SAVE-CARRY 40C4--A7       AND   A
        40C5--ED52        SBC   HL,DE
NO-RESTORE 40C7--37       SCF
CONT    40C8--10E7        DJNZ  40B1       LOOP

        40CA--225040      LD    (4050),HL  store-REMAINDER
        40CD--D9          EXX
        40CE--ED6A        ADC   HL,HL      complete
        40D0--CB11        RL    C            the
        40D2--17          RLA              Q(UOTIENT)
        40D3--47          LD    B,A
                                           BC=HI-Q
        40D4--E5          PUSH  HL
        40D5--CD2015      CALL  1520       STK-BC
        40D8--C1          POP   BC         BC=LO-Q
        40D9--CD2015      CALL  1520       STK-BC
        40DC--EF          RST   28;FPA:    N=Q H=INT(N/100) R=N-100*H
        40DD--01/30 00 41 00/04/0F/C5/     XCH/65536/*/+/STK N/
        40E5--30 37 48/2E/2D/30 40 44      100:M/N-MOD-M/DUPL/1E6
        40ED--74 24/03/32/00 15/01/C5/     /-/<Ø?/JP-TRUE../EXCH/STK R
        40F5--A4/03/32/2C/00 08/A0/01/     10/-/<Ø?/NOT/JP-TRUE./Ø/EXCH/
        40FD--34                           END-CALC
        40FE--CDDB15      CALL  15DB       PRINT H
        4101--EF          RST   28;FPA:
      • 4102--34                           END-CALC
        4103--CDDB15      CALL  15DB       PRINT H/Ø
        4106--EF          RST   28;FPA:
     •• 4107--E5 34                        GET-MEM/END
        4109--CDDB15      CALL  15DB       PRINT N-Q/R
        410C--CFED        RST   08;■

16653   410D--ED4B5040    LD    BC,(4050)  REMAINDER
        4111--C9          RET
```

De volgende keer gebruiken we de verworven kennis om PRIEMgetallen te GENEREREN.

H A N V A N A B B E

W I E K A N M I J H E L P E N

Zoals blijkt uit mijn in dit periodiek gepubliceerde MC-routines maak ik nuttig gebruik van de met de CALCULATOR mogelijke 'operations'. Maar de aandachtige lezer zal hebben bemerkt dat ik de 'COMPARISON' operations:

09 <= 0A >= 0B <> 0C > 0D <

NIET heb gebruikt. Terwijl dat op een aantal plaatsen toch tot eenvoudiger routines zou hebben geleid. En wie mij kent weet dat ik zo'n mogelijkheid altijd benut.

De reden van het niet benutten is, dat deze operations bij mij niet goed werken. In plaats van een "antwoord" 'true' (= 1), of 'false' (= 0), afhankelijk van de twee 'values' die vergeleken worden, krijg ik altijd 'true' bij gelijke waarden en 'false' bij ongelijke waarden, onafhankelijk van de grootte van die waarden.

Een eenvoudige testroutine is bijvoorbeeld:

EF/Ax/Ay/zz/34/C3DB15 met

Ax en Ay 1 van de 5 STK-constant CODEs en zz 1 van de 5 bovengenoemde operations CODEs.

Wie, o wie, kan mij vertellen of ik een denkfout maak en zo ja, welke. Of zijn mijn twee ZX-81's, waarvan 1 een TIMEX is, niet in orde?
GAARNE REAKTIES, waarvan ik verslag zal doen in dit periodiek.

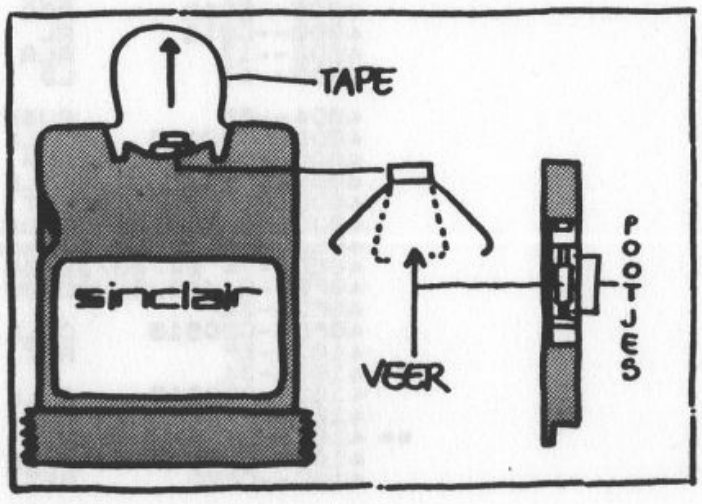
IR. H.H. VAN ABBE
VAN DER DOESLAAN 1
2242 PP WASSENAAR
TEL. 01751-14216

HANDIGE TIP VOOR MICRODRIVE?

De volgende tip voor Microdrive gebruikers vonden we in de Crash van juni. Aangezien niet iedereen de Crash leest vonden we dat we het artikeltje maar moesten vertalen en publiceren. Het betreft een simpele maar precieze operatie die er voor zorgt dat er zo'n 2-3K extra beschikbaar komt op de meeste tapes, én die vaak cartridges die het niet meer schijnen te doen nog even nieuw leven in te blazen.

Volgens de Crash kan dit allemaal bewerkstelligd worden door de kleine koperen veer die de tape tegen de Microdrive-kop drukt iets bij te buigen. De veer wordt ingedrukt als de cartridge in de Microdrive zit en als de tape niet meer goed op z'n plaats wordt gehouden ontstaan de problemen.

Gebruik een potlood of een (niet magnetische) schroevendraaier om een stukje van ongeveer 2 centimeter tape uit de cartridge te halen. Haal daarna voorzichtig de veer los door de pootjes uit het plastic te halen. Probeer het niet er ruw uit te trekken, want dan breekt de veer. Als de veer eruit is buig de pootjes dan voorzichtig naar elkaar toe om de veerkracht te vergroten.



Om de veer weer terug in de cartridge te krijgen moet de bovenstaande werkwijze in omgekeerde volgorde uitgevoerd worden. Als dat is gebeurt probeer dan het stukje tape weer in de cartridge te wurmen zonder er met je vingers aan te komen. Formatteer de cartridge en voila..... als alles goed is gaat-ie weer een tijdje mee. Nog een kleine tip:

probeer het liever niet met een cartridge waar waardevol spul op staat, maar het kan best de moeite waard zijn voor een cartridge die anders toch in de prullebak was beland en die op deze manier nog een hele tijd gebruikt kan worden.

Een eindje verder in dezelfde Crash stond trouwens het bericht dat Abbex, de firma die de cartridges voor Sinclair maakte, samen gaat werken met ICL om een betere, sterkere veer te maken voor de tape. ICL gebruikt namelijk cartridges in hun One-Per-Desk QL/telefoon kombinatie systeem.

HOEVEEL K OVER?

Het kan wel eens handig zijn om te weten hoeveel geheugenruimte nog beschikbaar is in BASIC als je aan het programmeren bent. Er is een ROM routine om hier achter te komen, en deze routine is vanuit BASIC op te roepen. Type het volgende in:

```
PRINT 65536-USR 7962
```

en je krijgt keurig op je scherm hoeveel geheugen er nog niet gebruikt is.

Voor de 128K kan het soms ook gemakkelijk zijn te weten hoeveel vrije geheugenruimte er nog is op de RAMdisk. In BetaBASIC 4 zit een verbeterde CAT! routine die de namen van de files en de beschikbare vrije ruimte aangeeft. Maar niet iedereen heeft BetaBASIC 4. Hetzelfde kun je ook bereiken in 128 BASIC door het onderstaande in te typen:

```
PRINT PEEK 23429 + 256 * PEEK 23430 +
65536 * PEEK 23431
```

Op je scherm verschijnt dan precies hoeveel bytes je nog kunt gebruiken. Voor elke file worden trouwens 29 bytes extra gebruikt voor filenaam, lokatie e.d. Dus een SCREEN$ van 6912 bytes gebruikt op een RAMdisk 6941 bytes.

Sinds we in Assen ook gebruikersavonden hebben is er een hoop veranderd in de Drentse hoofdstad. Zeker op computergebied. Kijk maar eens wat de kranten daar over te melden hebben:



OPUS ROM

Hallo computervrienden,

Na lange tijd niets van me te hebben laten horen, toch weer een artikel in het SGG bulletin. Deze keer een onderwerp speciaal voor de bezitters van een Opus Discovery. Zoals bijna iedereen wel weet zit er in de Opus drive een ROM die de benodigde software bevat om de diskdrive, printerpoort en de joystickpoort te besturen. Toen Sinclair met de Spectrum 128K op de markt kwam, heeft men bij Opus een nieuwe versie van de ROM gemaakt. Hiermee heeft men de diskdrive geschikt gemaakt voor het gebruik met de 128K. Door gebruik te maken van een systeem van Hook codes zoals ook bij Interface 1 is gebeurd, kan men de verschillende ROM versies door elkaar gebruiken. Via de Hook codes komt men bij de juiste routines in de ROM terecht. De 2.2 ROM, zoals de 128K versie wordt genoemd, bevat alle mogelijkheden van de 48K ROM plus nog enkele extra commando's. De 128K ROM kan ook zonder problemen met de gewone 48K Spectrum worden gebruikt.
De extra commando's zijn:

-FORMAT 6: "ram disk" drive 6 mag zowel in 48K als in 128K mode worden gebruikt. Drive 6 maakt gebruik van het extra geheugen van de Spectrum 128. Met een simpel MC CODE programma'tje kan van 48K naar 128K mode worden geschakeld, zonder het programma te verliezen. (Het programma staat beschreven in de handleiding die bij de 2.2 ROM wordt geleverd.)

-CLEAR commando: Er zijn nu twee nieuwe vormen.
CLEAR LINE <regel 1>,<regel 2> welke alle BASIC regels van <regel 1> tot en met <regel 2> wist.
CLEAR DATA <variabele> verwijdert de desbetreffende BASIC variabele.
Deze twee commando's zijn uitzonderlijk gemakkelijk als er BASIC overlay programma's van disk of ramdisk worden gebruikt.

-D channel: Wanneer het commando MOVE "d";1; "" TO "d";3 : wordt gebruikt, wordt er gevraagd of eventuele Data op disk 3 vernietigd mogen worden. Bij de oude ROM wordt deze beveiliging alleen bij het formatteren van een gebruikte disk toegepast.

-T channel: Als dit kanaal na gebruik wordt gesloten wordt er een carriage return (CHR$ 13) gegeven met het doel de printer-buffer te legen.

-Drive 4: Deze drive mag bij een dubbele Discovery worden gebruikt, voor de rechter drive, zoals we bij een Discovery met 1 drive met drive 1 en drive 3 werken.

-De parallel printerpoort kan als netwerk worden gebruikt, door twee Discoverys aan elkaar te koppelen via de printerpoort. Bij de 48K versie van de ROM werkte dit niet goed. In de 2.2 ROM (128K) heeft men de foutjes gecorrigeerd.

Bert Westenberg.
(wordt vervolgd(?))

Dat er ook nog mensen zijn die af en toe een tekening maken met de Spectrum bewijst de onderstaande illustratie van Bert van der Zaag. Is hij de enige die dit doet? of zijn er nog meer mensen die hun artistieke kwaliteiten botvieren met muis of joystick en Artstudio? Graag zouden wij meer van deze plaatjes plaatsen. Wie durft?



WAT IS . . . SCHADUW ROM

In de eerste echte originele Spectrum, die in 1982 uitkwam (is het nog maar zo kort geleden?), zat 16K ROM (Read Only Memory) en 16K RAM (Random Acces Memory). Al snel werd deze 16K machine opgevolgd door de 48K versie, die dezelfde 16K ROM had en 48K RAM. Een snel optelsommetje leert ons dat 16 + 48 = 64, dus de ROM en de RAM op de 48K gebruiken alle 64K van de Z80. In de praktijk betekend dit dat de Z80 niet meer dan 64K (ROM of RAM) tegelijk kan behandelen. Het is te vergelijken met een lelijke eend met 4 zitplaatsen, als er 4 mensen in zitten is de eend vol. Sinds de 48K Spectrum uitkwam is er een hoop hardware bijgekomen, zoals Interface 1, de 128K, de +2 en nu de +3. Al deze veranderingen vereisen maar één ding, meer geheugen. Hoe krijg je de Z80 zo gek om meer dan 64K te gaan gebruiken, hij heeft toch niet meer?

Het antwoord is simpel(?). Speciale routines om tussen verschillende geheugenbanken te switchenom een bepaalde taak te volbrengen. Om bij ons voorbeeld te blijven: je schopt een van de passagiers uit de lelijke eend om plaats te maken voor een ander. De uitdrukking "schaduw ROM" werd voor het eerst gebruikt bij interface 1, deze had zelf 16K ROM ingebouwd. Deze ROM zit op het zelfde adres als de gewone 16K ROM in de Spectrum en wordt alleen gebruikt als de Z80 besluit een Interface 1 mogelijkheid te gebruiken. Op dat moment wordt de gewone 16K ROM uitgeschakeld. Zo is dus ook de term "schaduw ROM" verklaard, de ROM is en blijft uitgeschakeld (in het donker dus) tot de Z80 besluit de originele ROM uit- en de schaduw ROM in te schakelen.

DRUKWERK

PORT BETAALD
ZUIDHORN

Afz.:

SGG
redaktieadres
Geert Valckeshof 28
9351 RX LEEK

AAN:

_____

_____