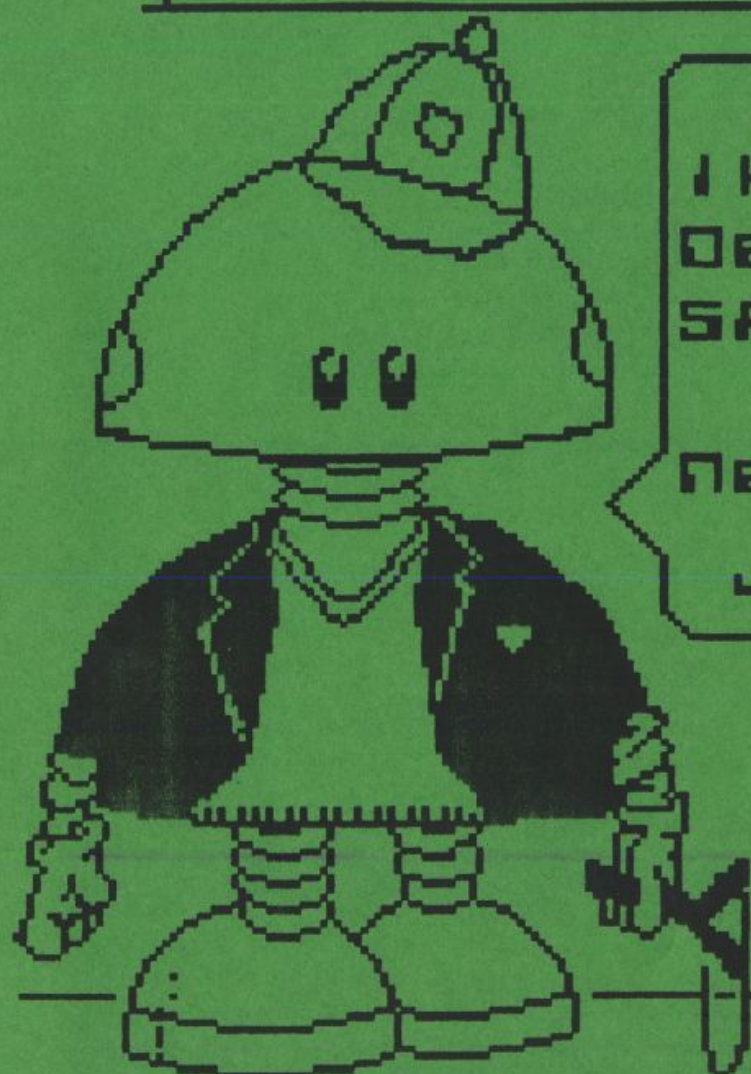


BULLETIN

SINGLAIR
GEBRUIKERSGROEP
GRONINGEN/ASSEN



Hallo,
ik ben SAM.
De eerste
SAM COUPE
is in
Nederland
!!aiaaor

SAM coupe

7e jaargang nr 7
maart '90

COLOFON



VOORZITTER:

Jan Dirk Burggraaf
Kluivingskampenweg 30
9761 BP Eelde
☎ 05907-1697

SEKRETARIS:

Martin den Hollander
Numero Dertien 8
9644 TV Veendam
☎ 05978-45474

PENNINGMEESTER/ VERHUUR:

Jan Arends
Heiligelaan 66
9636 CP Zuidbroek
☎ 05985-2247
Giro 5965342 t.n.v.
rekening SGG

VICE VOORZITTER/ MATERIAALCOMM

PR:
J. van Alteren
De Grouw 6
9351 LP Leek
☎ 05945-15678

ALGEMEEN:

Roelof Koning
Selwerderstraat 26
9717 GK Groningen
☎ 050-124298

REDAKTIE:

Mevr. F. Elstrodt
Kam. Onnesstraat 172
9727 HS Groningen
☎ 050-263930

Rudy Biesma
Betuwe 18
9405 JJ Assen
☎ 05920-50643

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

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

De sluitingsdatum voor kopij wordt in elk bulletin vermeld.

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

Het lidmaatschap van onze gebruikersgroep bedraagt f 17,50 per kalenderjaar voor personen tot en met 17 jaar voor oudere personen is dit f 25,00 per kalenderjaar. Bij deze prijs is het abonnement op het bulletin inbegrepen.

U kunt lid worden van de SGG door U op te geven bij de penningmeester.



HALLO ALLEMAAL

Voor U ligt alweer ons maart nummer, U mag best weten dat wij een beetje trots zijn op ons bulletin, dat helemaal gemaakt wordt op de Spectrum. Onze laatste aanwinst is weer een handig hulpmiddel voor voornamelijk redactionele leden. Maar daarover volgende keer meer.

Onze voorplaat toont U SAM, waarmee U veel te maken krijgt met de SAM Coupe. In dit nummer leest U Edwin zijn eerste belevenissen met deze nieuwe computer. Het brein van Edwin heeft duizend toeren gedraaid, om dit figuurtje (SAM) met de Spectrum uit te printen.

Zoals U in dit nummer kunt zien is de "wie maakt de 'kleinste' screendump (routine) -kompetitie" volop aan de gang.

Vervelend voor Tonnie is, dat hij zich ver-Stap-t heeft, maar voor de redactie is dit erg gezond, de DISCiPLE kopij vliegt binnen.

Hans Werter biedt een EPROM service voor onze leden, geweldig! Heeft U een Opus 2.1 ROM dan kunt U deze bij Hans omruilen voor een 2.22 versie.

Flora en Rudy

```
*****
*                                     *
*               Wat kunt U verwachten:               *
*   Een soort DTP programma, heel mooi, heel handig   *
*   AUTOSysteem loader voor de DISCiPLE                 *
*   Handige routinetjes voor in de Multiface            *
*   Hoe werkt de floppydisk-controller (DISCiPLE & Opus) *
*   De SAM in de praktijk                               *
*                                     *
*****
```

In dit nummer:

-Gebruikersbijeenkomsten	: redactie
-Van onze voorzitter	: Jan Dirk
-Opus ROM 2.1 => 2.22	: Hans Werter
-Vlinder	: Flora
-Small Dump versie 3	: Edwin Blink
-Operatie geslaagd patient overleden	: Roelof Koning
-Muziek 128K (Pride)	: Frans Postma
-Machinetaal grapje	: Edwin Blink
-DISCiPLE copier bug	: Tonnie Stap
-Me and my SAM	: Edwin Blink
-Machine Code stap voor stap deel 8	: J.v.Alteren
-Snel en precies rekenen deel 28	: H.v.Abbe

SLUITINGSDATA KOPIJ 24 MAART

GEBRUIKERSBIJENKOMSTEN



In het: **DENKSPORTCENTRUM**
OLIE MULDERSWEG 43
GRONINGEN

Telefoon: 050-126937

DATA GRONINGEN:

10 mrt. zater van 14.00 -17.30
 10 apr. dins van 19.30 -22.30
 12 mei. zater van 14.00 -17.30
 12 jun. dins van 19.30 -22.30

HOUTEN:

17 MRT. 12 MEI.
 23 JUNI. 8 SEPT. 27 OKT.
 In het HCC-kantoor Standerdmoen 8
 te Houten. tel:03403-78788.

HOOGVEEEN:

Elke 2e maandag vanaf 19.30
 In het Wijkcentrum "DE MAGNEET"
 ORION 2 HOOGVEEEN.

MAART		1990			
ZO	-	4	11	18	25
MA	-	5	12	19	26
DI	-	6	13*	20	27
WO	-	7	14	21	28
DO	1	8	15	22	29
VR	2	9	16	23	30
ZA	3	10*	17	24	31

APRIL		1990			
ZO	1	8	15	22	29
MA	2	9	16	23	30
DI	3	10*	17	24	-
WO	4	11	18	25	-
DO	5	12	19	26	-
VR	6	13	20	27	-
ZA	7	14	21	28	-

DATA EEMSMOND:

17 mrt. 31 mrt. 14 en 18 april.

In de L.O.M. School "De Wenakker" Pastorielaan 2 Appingedam.

VERHUUR VERHUUR VERHUUR

U kunt bij de club het volgende huren:

A.M.X. muis met interface
 CURRAH microspeech
 VTX 5000 modem

Inlichtingen bij de heer Jan Arends.

CASSETTE SERVICE

Van bijna alle programma's die in ons bulletin stonden hebben wij een cassette bandje gemaakt.
 U kunt dit bandje op een gebruikersbijeenkomst lenen, meestal ligt deze bij de deurwacht, zoniet vraag het dan aan Flora.
 De programma's waar Copyright op zitten, kunnen natuurlijk niet op deze cassette voorkomen.

VAN ONZE VOORZITTER



Een bijzonder interessante avond vond ik de laatste bijeenkomst.

Wat was er dan aan de hand zult U misschien zeggen. Uitsluitend Spectrums in allerlei uitvoeringen.

Inbouw, uitbouw, ombouw en opbouw en dan de randapparatuur zoals cassetterecorders, microdrives, interface 1, Opus, Disciple, losse disc-drives met een behoorlijke opslagcapaciteit en dan de printers dat is toch gewoon op een computeravond.

Dat is waar, het lijkt gewoon, maar toch als men eens een andere clubavond bezoekt valt het je toch wel op dat wij met onze Spectrum computer hobby een unieke plaats innemen.

Altijd maar weer je spullen meenemen en aan anderen laten zien wat je nu weer ontwikkeld hebt of wat voor probleem het apparaat nu weer aan je toevertrouwd, dat is toch echt voor mijn gevoel de juiste manier waarop we met elkaar op de goede weg blijven.

Genoeg hobbybladen die op de materie ingaan zou je denken. Niets is minder waar.

Onder andere het blad Disciple heeft moeten afhaken om wat voor reden dan ook.

En daar werden nu juist de specifieke problemen aangekaart die dit Interface met zich meebracht. Jammer toch.

Maar een van onze leden die "nogal" bekend is met dit Interface hebben wij bereid gevonden om zijn kennis uit te dragen in ons maandblad.

We kunnen dan ook de komende tijd artikelen van hem verwachten.

Wij als bestuur hebben gemeend om aan de mensen die geabonneerd waren op de Disciple Nieuwsbrief dit keer gratis ons maandblad toe te sturen ter kennismaking.

Totzover de bijdrage van jullie voorzitter

J.D. Burggraaf



ROM 2.1 TO 2.22

2.1

2.22

PROGRAMMEER SERVICE VOOR EPROMS

Een aantal clubleden heeft al kennisgemaakt met de eeproms versie 2.22 voor de OPUS. Deze kennismaking was niet in alle gevallen een succes omdat bepaalde spelletjes tevergeefs zoeken naar bepaalde routines die zich in de versie 2.22 op een andere plaats bevinden. Niettemin ga ik onverdroten door met adverteren want deze versie heeft toch wel een aantal voordelen nl.:

- Samenwerking van de OPUS met een Spectrum 128K is mogelijk.

Je kunt snel hele blokken uit een programma verwijderen met het commando CLEAR LINE x,y waarbij x de eerste, te verwijderen, regel is en y de laatste.

Ditzelfde geldt ook voor variabelen; deze kunnen selectief worden gewist met het commando CLEAR DATA a\$, of Clear DATA a (of welke \$, of numerieke variabele dan ook) Wie er, na dit gelezen te hebben, trek in heeft gekregen moet op de volgende clubbijeenkomst zijn eeprom maar even komen omruilen bij mij Overigens ben ik ook een poging aan het wagen om de ROM van de ZX81 te copiereen; ik heb eens gelezen dat daar behoefte aan is. Wie een EPROM geprogrammeerd wil hebben moet het mij maar even zeggen, alles is mogelijk tot 64 kbytes toe zolang ik het tenminste niet hoeft in te toetsen !

Hans Werter.

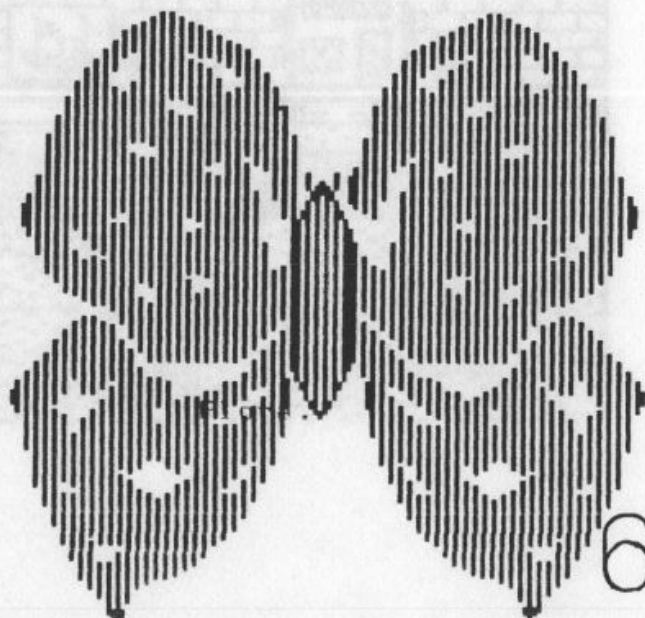
VLINDERS

Een leuke demonstratie van gekleurde pixels.

```

35 BORDER 0: PAPER 0: INK 9
37 CLS
40 LET a= INT ( RND *128)
60 LET b= INT ( RND *88)
70 LET c= INT ( RND *7)+1
90 INK c
100 PLOT 128-a,88-b
110 PLOT 128+a,88-b
120 PLOT 128-a,88+b
130 PLOT 128+a,88+b
140 IF RND >.99 THEN GO TO 35
150 GO TO 40

```



SMALL DUMP V.3



Vorige maand heb ik een greyscale routine geschreven. Hiermee kun je ook gewone dumps maken door de kleuren weg te laten.

Echter, ik vond het ook wel een uitdaging om eens te proberen een screendump routine te schrijven die een plaatje heeeel..... klein afdruckt.

Hierbij stuitte ik op dezelfde problemen die ik eerst ook met mijn greyscale routine had.

Namelijk cirkels werden eieren, en geen cirkels.

Maar na wat berekeningen is het mij toch gelukt.

En hier is hij dan :
Small Dump V.3
maakt een screen dump
zo groot als
een luciferdoosje
(5.4 cm bij 4.1 cm)
en dat in 140 bijtes



Voordat je de listing intikt is het wel belangrijk dat je printer over een double density bit image mode en een n/216 INCH line feed beschikt.

Anders werkt de routine niet.

Als je printer andere controle codes heeft, kun je ze in de regels 290 t/m 310 veranderen.

Net als bij mijn vorige routine moet de printer eerst geïnitialiseerd worden.

Voor de Opus Discovery : OPEN # 3;"B"
VOOR +D en Disciple : POKE @6,1

Eventueel kun je bij het initialiseren ook een linker kantlijn meegeven. (ESC "l"; CHR\$ x; x = de kantlijn)

Tot slot wil ik nog even zeggen dat als je de routine lager dan 30000 wilt hebben, dan moet je van een object adres gebruik maken. (= een tussen adres).

VEEL PLEZIER,

Edwin Blink.

BULLETIN SGG

Listing Smalldump V.3 van Edwin Blink.

```
10 CLEAR 29999: PRINT "Small Dump V.3"
20 PRINT AT 2,0;"bij het plaatsen van de routine";
  AT 4,0;"op adressen tussen 23552 - 30000"; AT 6,0;"moet
  gebruik gemaakt"; AT 8,0;"worden van een Object adres."; AT
  4,12; OVER 1;" ____"; AT 6,0;" ____"
30 LET 1290=4
40 LET 1300=4
50 LET 1310=4
60 REM regel 30 t/m 50 lengte printer controle codes (regel 29
  0 t/m 310)
70 INPUT "run adres:";rad
80 INPUT "met object adres ?" LINE a$
90 IF a$="j" OR a$="J" THEN INPUT "object adres:";oad: GO TO
  110
100 LET oad=rad
110 PRINT AT 12,0;"** De code wordt ge POKE'd **"
120 RESTORE 150: FOR n=0 TO 127+1290+1300+1310: READ a: IF a=33
  3 THEN LET a=rad+ PEEK (oad+n-1): POKE oad+n, INT (a/256):
  POKE oad+n-1,a-256* INT (a/256): NEXT n
130 POKE oad+n,a: NEXT n
140: CLS : PRINT AT 10,0;"Save de code nu met:"; AT 12,0;"SAVE
  ""naam"" CODE ";oad;",";128+1290+1300+1310: STOP

150 DATA 62,3,205,1,22,175,79,71,30,32
160 DATA 33,132,333,205,33,333,4,30,96,33
170 DATA 136,333,205,33,333,120,198,4,254,192
180 DATA 56,231,201,197,229,205,115,333,205,47
190 DATA 333,225,205,118,333,193,201,197,213,205
200 DATA 76,333,203,19,4,4,48,247,203,123
210 DATA 40,3,5,24,240,123,15,15,15,215
220 DATA 209,193,12,32,228,201,120,254,192,208
230 DATA 205,177,34,87,62,128,40,4,15,21
240 DATA 24,250,166,8,124,230,24,15,15,15
250 DATA 246,88,103,8,126,32,3,15,15,15
260 DATA 230,7,254,4,201,33,128,333,126,230
270 DATA 127,215,126,35,23,216,24,246

280 REM regel 290 t/m 310 controle codes voor de printer.
  De contole codes mogen niet hoger dan 127 zijn.
  Bij de laatste code moet 128 opgeteld worden (End marker).
290 DATA 27,76,0,1+128: REM Double density graphics 256 bytes
300 DATA 13,27,74,2+128: REM cariage return, n/216 INCH line -
  feed met n=2
310 DATA 13,27,74,7+128: REM cariage return, n/216 INCH line
  feed met n=7
```


BULLETIN SGG

OPERATIE GESLAAGD ??



patient overleden.

De M.C.N.-screendump voor de OPUS, stammend uit 1985 en 148 bytes lang, die ik jarenlang gebruikt heb omdat het zo'n lekker korte routine was, heeft bij mij het loodje gelegd.

Het begon toen ik een kleine tekortkoming (deze routine herstelt de normale linefeed van de printer niet) wilde opheffen. Een eerste blik op de disassembly-listing vertelde mij al dat er een aantal 'doublures' in voorkwam die dan ook maar meteen verholpen moesten worden. Het resultaat was een verbeterde (en werkende) uitvoering van nog slechts 105 bytes lang.

De magische grens van 100 bytes lag nu verlokkelijk dichtbij, en toen gebeurde er iets merkwaardigs. Was het onvoorzichtigheid, of speelden duistere krachten een noodlottig spel? Ik herinner me dat eens, heel lang geleden Maar laat ik niet afdwalen en verdergaan met m'n verhaal. (Het was overigens een nuchtere opmerking van Edwin Blink die een nieuw licht op de zaak wierp)

Met hernieuwde kracht begon ik aan het omwerken, opwerken en uitwerken. Dit resulteerde hierin dat er van de oude M.C.N.-routine niets meer over bleef, maar dat er een geheel nieuwe OPUS-screendump routine is ontstaan met een lengte van, let op, 73 bytes! Kenners (ikzelf dus) noemen dit kort, om niet te spreken van superkort, zelfs voor een 'recht voor z'n raap' routine als deze. Dit moet op een avond toch in te typen zijn?

Ongetwijfeld zullen echt slimme rakkers er nog een paar bytes uit kunnen knippen, maar ik besluit met een tevreden gevoel en onderstaande listing.

Roelof Koning.

P.S. Zorg dat RAMTOP goed staat,
Een printer-channel hoeft niet geopend worden,
Let eens op het scherm tijdens de dump,
Mogelijk moet voor jouw printer een ander getal gegeven worden ipv. de 24 in de laatste DATA-regel.

P.P.S. Een echte OPUS-bug gezien:
Probeer: 10 CLS #: LIST daarna RUN

```
10 LET op=2659: INPUT "versie 2.22? (j/n) ";i$
20 IF i$<>"j" AND i$<>"J" THEN LET op=2479
30 INPUT "op welk adres? ";a
40 LET tab=a+65: LET sub=a+54
50 RANDOMIZE op: LET olo= PEEK 23670: LET ohi= PEEK 23671
60 RANDOMIZE tab: LET tlo= PEEK 23670: LET thi= PEEK 23671
70 RANDOMIZE sub: LET slo= PEEK 23670: LET shi= PEEK 23671
100 RESTORE: FOR f=0 TO 72: READ q: POKE a+f,q: NEXT f
120 PRINT "codeblok = ";a;" ";73;"start = USR ";a
130 STOP
1000 DATA 175,79,71,17,tlo,thi,26,205,slo,shi,26,19
1010 DATA 61,32,247,30,8,120,205,177,34,62,1,203,6
1020 DATA 23,36,48,250,205,slo,shi,12,29,32,237,121
1030 DATA 167,32,231,120,198,8,254,192,32,211,62,10
1040 DATA 205,slo,shi,62,13,205,8,23,197,205,olo,ohi
1050 DATA 193,195,72,23,13,27,74,24,27,75,0,1
```

MUZIEK 128K



PRIDE

```

350 LET a$="UX1500W2T12005(3&1gCEgCE)(3&1aDFaDF(3&1gDFgDF)(3&1g
CEgCE)(3&1aEAaEA)(3&1#faD#faD)(3&1gDGgDG)((3&1egCegC)))(3&1d#
fCd#fC)(3&1dgbdbg)(3&1eg#Ceg#C)(3&1daDdaD)(3&1dfbdfb)(3&1cgC
cgC)04((3&1aCFaCF)))(3&1gbFgbF)(3&1gcEgcE)(3&1$bCE1$bCE)(3&1a
CEaCE)(3&1aC$EaC$E)(3&1bCDbCD)(3&1gbDgbD(3&1gCEgCE)(3&1gCFgC
F)(3&1gbFgbF)(3&1aC#FaC#F)(3&1gCGgCG)(3&1gCFgCF)(3&1gbFgbF)(
3&1g$bEg$bE)3&1faCFCaCafafdf05N3&1gbDFDbDbgdbdfedV15N9C"

351 LET b$="UX400W704(1&4_5E)(1&4_5D)(1&4_5D)(1&4_5E)(1&4_5E)((
1&4_5D))UX1500W2((1&4_5C))(1&4_5a)(1&4_5b)(1&4_5$b)(1&4_5a)(
1&4_5$a)(1&4_5g)((1&4_5f))(1&4_5d)(1&4_5e)(1&4_5g)(1&4_5f)(1
&4_5c)((1&4_5f))(1&4_5e)((1&4_5d))(1&4_5$e)(1&4_5e)((1&4_5d)
)(1&4_5c)1&4_5_7c1&4_5_7$c9E"

352 LET c$="UX400W704N7CCCCbbCCCCCbbbaaddgggggffffeeeedd03ggCC
CCUX2000W2Nff#f#f$a$a$agggggggggggggggggggcc9ccc"

353 PLAY a$,b$,c$
    
```

Frans Postma.

MACHINETAAL GRAPJE

```

5 CLEAR 29999: PRINT INK 7:"JoJoHereWeGo"
10 LET l=3e4: LET a$="12463512"
15 READ a: IF a <> 333 THEN POKE l,a: LET l=l+1: GO TO 15
20 FOR p=0 TO 2: FOR x=0 TO 31: FOR y=0 TO 7: IF POINT(x+32*
p,175-y) THEN POKE 22528+32*(y+8*p)+x,8* VAL a$(y+1)
25 NEXT y: NEXT x: NEXT p: PRINT AT 0,0: TAB 11;" "
30 RANDOMIZE USR 30000: GO TO 30
110 DATA 118,243,62,24,33,1,88,17,0,88,8,1,31,0,26,237,176,18,
19,35,8,61,32,-14,251,201,333
    
```

Edwin Blink.

DISCIPLE-COPIER BUG



Waarschuwing voor gebruikers van
de diskcopiers van Tonnie Stap.

Al bijna 2 jaar geleden schreef ik in de DISCIPLE Nieuwsbrief over een bug in de DISCIPLE betreffende de bitmap flag. Deze bug blijkt nu ook op te treden bij gebruik van mijn copiers. Omdat de bug toch nog iets anders is dan ik toen dacht volgt hier eerst een beschrijving:

Disk WRITE protected Not enough SPACE on disk

Als je probeert te SAVEn en je krijgt deze foutmeldingen dan wordt er een flag verkeerd gezet in de DISCIPLE. Deze flag geeft aan of de DISCIPLE de bitmap moet laden bij de volgende SAVE. Normaal staat de flag op 0 en dan wordt de bitmap geladen.

Tijdens het SAVEn wordt er echter eerst 2 en dan 1 in gezet. Na het SAVEn wordt het (bewaarde) oude getal er weer in gezet. Echter na de genoemde foutmelding blijft er het getal 1 in staan! Gevolg is dat de bitmap niet meer van disk gehaald wordt en de DISCIPLE de hele tijd met de bitmap in z'n geheugen blijft werken.

Hierdoor zal de DISCIPLE denken dat de lege sectoren van de disk waarop geprobeerd werd te SAVEn ook leeg zijn op de disk die je eventueel later in je drive doet! Is die schijf voller en zijn er dus meer sectoren bezet dan zal een nieuwe file over de al bestaande geSAVED worden. Zolang je echter met dezelfde disk blijft werken is er niets aan de hand.

De oplossing is om zelf de flag goed te zetten na beide bovenstaande foutmeldingen en wel door in te typen:

POKE @ 6999,0

Het verschil met mijn originele artikel is dat ik toen dacht dat de bitmap helemaal niet meer wordt aangepast maar dat had ik mis. In de interne bitmap worden de sectoren van de files die je later SAVED keurig ingevuld, bij ERASE worden ze echter niet meer weggehaald!

Toen ik destijds de bug ontdekte was ik enorm opgelucht, ik had er namelijk al heel wat files aan verloren. Een echte oplossing voor de bug heb ik nog steeds niet, hij zit waarschijnlijk ergens in de DISCIPLE ROM dus is er een nieuwe EPROM nodig om hem te verhelpen.

BULLETIN SGG

Nu wat betreft mijn disk copiers, hierin heb ik gebruik gemaakt van de flag en wel als volgt:

Nadat een file is geSAVED op de DESTINATION disk zet de copier de bitmap load flag op 2 zodat bij de volgende SAVE de bitmap niet weer opgehaald wordt. Dit doe ik omdat het ophalen van de bitmap een paar seconden kost en als je veel files te kopiëren hebt kan dat al snel in de minuten lopen. Om nu te voorkomen dat de flag zo blijft staan heb ik bij de BEEP routine in de copier opdracht gegeven de flag weer op 0 te zetten. Bij elke foutmelding of diskwisseling wordt er een BEEP gegeven en zal de flag dus weer goed staan. Dat was op zich niet fout, maar als de laatste file geSAVED is, wat gebeurt er dan?

Alleen als je de optie MULTIPLE COPIES aan hebt staan volgt er een BEEP, anders volgt meteen een reset terwijl de flag nog op 2 staat! Ook hier zul je dus zelf de flag op 0 moeten zetten!

De bug treed op in al mijn diskcopiers, dus ook in die voor de PLUS-D, de door Willem Lageman aangepaste versie van de copier is helaas ook niet vrij van de bug. Wat betreft de PLUS-D weet ik trouwens niet of de bug ook optreed na de foutmeldingen 'Disk WRITE protected' en 'Not enough SPACE on disk', ik vermoed echter van wel.

Ik betreur het ten sterkste dat ik nu pas achter deze bug ben gekomen, 2 jaar na het voltooien van de copiers. Echter ook niemand anders heeft de bug opgemerkt (of doorgespeeld), ik zelf ben er pas recent achter gekomen. Ik gebruikte de copiër namelijk enkele tientallen keren achter elkaar en toen bleek dat er toch wel erg vaak files overschreven werden, na een paar dagen begon ik eindelijk de copier te verdenken en na een beetje spitwerk vond ik de bug.

In een toekomstige versie van de copier zal ik deze bug zeker oplossen, voorlopig moet hij nog achteraf gecorrigeerd worden.

Tonnie Stap.



Een bug minder!



ME AND MY SAM



In december heb ik de SAM Coupé bij MGT besteld en hebben ze me verteld dat ik hem in januari zou krijgen.

Op 2 febr was het dan zover, s'morgens vroeg werd ik mijn bed uitgebeld door de postbode.

Half wakker heb ik Snel het een en ander aangetrokken en met de girocheque wapperend in mijn hand holde ik naar beneden.

Nadat ik met de postbode alles afgehandeld had, rende ik naar boven. En daar zat ik dan, met een verzegeld pakket voor mij.

Na een schaar te hebben gevonden verbrak ik de verzegeling en verwijderde de verpakking.

Tussen al die schuimrubberen stukjes vond ik een grote doos met daar naast een klein doosje.

Op het eerste gezicht zag het er goed verzorgd uit, en dat bleek ook zo te zijn.

De SAM Coupé, zat in een mooie doos. Toen ik die openmaakte kwam ik eerst bij de boeken en demo's en daaronder was de SAM verborgen.

Het is een goede deugdelijke verpakking, het lijkt wel een koffertje. Makkelijk mee naar de club avond te nemen.

Zo, nu kon ik hem eens goed bekijken, ó wat is hij mooi.

Een witte kleur en licht grijze toetsen. Maar er ontbreekt een disk-drive.

Nadat het kleine doosje weer in mijn oog viel, zag ik op de doos 'SAM Drive' staan.

Gauw het doosje geopend en ja hoor daar zat hij in een bescherm zakje.

Tot mijn verwondering was hij lekker plat. Na het drive-je in de Coupé te hebben gezet en alles aangesloten te hebben was ik even teleur gesteld.

Ik vond het beeld te breed. Dit klopte volgens mij niet en toen ik een cirkel tekende wist ik het zeker.

Maar gelukkig heb ik op mijn monitor wat knoppen zitten om het beeld bij te regelen.

Zo nu is een cirkel een cirkel en geen ei meer.

Nadat ik de demo had geladen van tape (ja, want ik kan nog niet met de drive werken omdat de DOS nog wat verbeteringen ondergaat en daarom later wordt toegezonden), en wat leuke figuurtjes had gezien, werd er verteld dat deze in Basic gemaakt zijn en dat je in machine code nog beter kan doen.

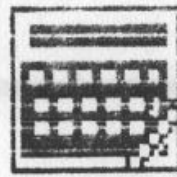
Nou dat wilde ik wel zien en tot mijn verbazing kwam er op eens een plaatje met een maan mannetje, net een echte foto !!!.

Na een paar dagen ben ik begonnen met een printer interface te maken omdat de Coupé die niet heeft.

En omdat er maar weinig voor nodig was heb ik gelijk maar een ADC/DAC chip bij het doosje van 10 bij 5 cm in gedaan, waarmee je bv. muziek in de computer kan lezen.

Nu ben ik bezig met het uitzoeken hoe de Coupé in elkaar zit.

Edwin blink, een trotse SAM Coupé bezitter.

**MACHINECODE
STAP VOOR STAP DEEL 8**

(c) J. v. Alteren, SGG; 220290.

Uit ZX-Computing afl.: apr/mei '85, blz. 88 e.v..
door David Nowotnik.

8e aflevering:

DEEL 3: Verplaatsen van blokken: (Vervolg)

Het laatste deel theorie van dit gedeelte zijn de krachtige blok verplaats-opdrachten van de Z80. Deze maken het mogelijk, om een blok bytes (max 65536) van het ene gedeelte naar het andere gedeelte van het geheugen te verplaatsen.

Het werkt als volgt:

Het startadres van het te verplaatsen blok wordt geladen in het HL-registerpaar, het bestemmingsadres wordt gegeven door DE, en het aantal bytes dat moet worden verplaatst gaat naar BC. Daarna maakt de opdracht LDIR de transactie compleet.

LDIR betekent: Load Increment Repeat

De waarde in BC werkt als een teller. De byte aangewezen door het HL-register wordt verplaatst naar het adres dat in DE staat; HL en DE worden verhoogd terwijl BC verlaagd wordt. Daarna herhaalt deze opdracht zichzelf automatisch totdat BC nul wordt.

Er is een niet repeterend alternatief LDI, deze instructie doet verder hetzelfde als LDIR (BC wordt wel verlaagd).

Beginners met mc gebruiken meestal LDIR en dit wordt ook in de voorbeelden gedaan.

De Z80 heeft nog een blok verplaats opdracht nl LDDR. Het verschil met de LDIR is dat de LDDR instructie HL en DE verlaagd ipv verhoogd.

LDDR betekent: Load Decrement Repeat

HL moet nu geladen worden met het eindadres van het te verplaatsen blok terwijl DE het eindadres van de bestemming moet bevatten. In BC staat weer het aantal te verplaatsen bytes.

LDD is de niet repeterende versie van LDDR.

Het doet er echt niet toe of U LDIR of LDDR gebruikt, behalve als er enige overlap is tussen de oorspronkelijke en toekomstige geheugenblokken is.

De hexadecimale opcodes van de rekenkundige en blokverplaats instructies staan in tabel 3.

Voorbeeld 1

```

50 REM INITIALISATIE
60 CLEAR 29999
70 LIST 900
100 DIM F$(8,3)
110 LET F$(1)=" S "
120 LET F$(2)=" Z "
130 LET F$(6)="P/V"
140 LET F$(8)=" C "
200 REM VOER MACHINE CODE IN
210 LET X=3E4
220 LET T=0
230 INPUT Y
240 IF Y=-1 THEN GO TO 3E2
250 POKE X,Y
260 LET X=X+1
270 LET T=T+Y
280 GO TO 230
300 IF T=2100 THEN GO TO 500
310 PRINT "INVOER FOUT (";T;")"
320 STOP
500 CLS
510 PRINT AT 21,0;"GEEF WAARDE VAN 'A' REGISTER"
520 INPUT A
530 PRINT AT 21,0;"GEEF DE TE VERGELIJKEN WAARDE"
540 INPUT C
550 POKE 32500,A
560 POKE 32501,C
570 IF USR 30000 THEN
580 CLS
590 PRINT
600 PRINT "WAARDE IN 'A' REGISTER IS ";A
610 PRINT
620 PRINT "VERGELIJKINGS WAARDE = ";C
630 PRINT
640 PRINT "INHOUD VAN DE FLAGGEN:"
650 LET X=32502
660 FOR I=1 TO 8
670 PRINT
680 IF I=3 OR I=4 OR I=5 OR I=7 THEN GO TO 700
690 PRINT F$(I);" ";PEEK X
700 LET X=X+1
710 NEXT I
720 PRINT AT 20,2;"DRUK EEN TOETS"
730 IF INKEY$="" THEN GO TO 730
740 GO TO 500

```

BULLETIN 566

machine code voorbeeld 1:

33,244,126	LD	HL,32500	Hier staat de waarde voor A.
126	LD	A,(HL)	
35	INC	HL	HL wijst naar de te
190	CP	(HL)	vergelijken waarde. Vergelijk.
245	PUSH	AF	
209	POP	DE	'LD E,F'
6,8	LD	B,8	Er zijn 8 flag bits.
35	LOOP	INC HL	Wijs naar de volgende byte.
54,0	LD	(HL),0	Maak de waarde 0.
203,19	RL	E	Verplaats 1 bit van E in carry.
48,2	JR	NC,NEXT	Volgende bit als deze 0 is.
54,1	LD	(HL),1	Anders geef aan dat deze 1 is.
16,245	NEXT	DJNZ LOOP	Ga door totdat alle flags
201	RET		bekeken zijn en RETurn dan.

Voorbeeld 2:

```

10 CLEAR 24999
20 FOR i=25000 TO 25023
30 READ x: POKE i,x
40 NEXT i
50 DATA 33,0,64,17,112,98
60 DATA 1,0,27,237,176,201
70 DATA 33,112,98,17,0,64
80 DATA 1,0,27,237,176,201
90 CLS
100 FOR i=1 TO 20
110 PRINT INK i/3; PAPER 8-i/3;"111122223333444455556666777788"
120 NEXT i
130 RANDOMIZE USR 25000
140 CLS
150 PRINT AT 12,0;"Druk een toets"
160 IF INKEY$="" THEN GOTO 160
170 RANDOMIZE USR 25012

```

machine code voorbeeld 2:

25000	LD	HL,16384	Startadres van het schermgeheugen
	LD	DE,25200	Startadres van de bewaarplek
	LD	BC,6912	Lengte van het scherm
	LDIR		Verplaats de bytes
	RET		
25012	LD	HL,25200	Verplaats de bytes terug naar het scherm
	LD	DE,16384	
	LD	BC,6912	
	LDIR		
	RET		

De voorbeelden:

Het eerste voorbeeld demonstreert de stand van de flaggen na een vergelijking van twee getallen dmv de ComPare instructie.

Geef RUN en voer de machinecode getal voor getal in en eindig met -1.

Als U een fout heeft gemaakt zal het programma dit melden. Na invoer van de machinecode komt het programma terecht in de hoofdroutine, vanaf hier staan er instructies op het scherm zodat een verdere omschrijving niet nodig is. Er staan 3 nieuwe instructies in de mc-listing, deze zullen in volgende afleveringen worden behandeld.

Het tweede voorbeeld demonstreert het gebruik van de LDIR instructie.

De machinecode staat bij dit voorbeeld in de DATA regels.

Voor beide programma's geldt:

Na het intikken eerst SAVEn, dan pas RUNnen.

De volgende keer krijgt U te maken met "uitgangen" en BIT instructies.

Tabel 3:
Rekenkundig:

8 bits

Instructie:

	r:								
	A	B	C	D	E	H	L	(HL)	n
ADD A,r	87	80	81	82	83	84	85	86	C6
ADC A,r	8F	88	89	8A	8B	8C	8D	8E	CE
SUB A,r	97	90	91	92	93	94	95	96	D6
SBC A,r	9F	98	99	9A	9B	9C	9D	9E	DE

16 bits

Instructie:

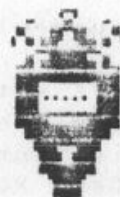
	rr:		
	HL	DE	BC
ADD HL,rr	29	19	09
ADC HL,rr	ED6A	ED5A	ED4A
SBC HL,rr	ED62	ED52	ED42

Blok verplaatsingen:

Instructie:	Opcode:
LDI	EDA0
LDIR	EDB0
LDD	EDA8
LDDR	EDB8

J. v. Alteren.

----- WORDT VERVOLGD -----



Begin 1989 had ik "9 * PRIME" met 17 LOOPS in MC ontwikkeld. Toen realiseerde ik me, dat het gebruiken van LOOPS voor het maken van "bij elkaar passende" SETS van G1 G2 G3 veel sneller zou kunnen werken, dan met de in deel 17 gevolgde methodiek. En inderdaad ben ik er in geslaagd de genoemde tijd van 488 s terug te brengen tot 20 s!. Daarmee komt de berekeningstijd van het PG op 159 s, ruim 4-maal zo snel als oorspronkelijk in (17) en 197 keer zo snel als het door HCC verkozen programma. Er is nu ook een bevredigende verdeling van de rekentijd tussen het "domme" maken van de SETS en het "moeilijker" vermenigvuldigen en testen van deze SETS. Ter onderscheiding noem ik het -nu echt laatste- PG ter oplossing van het relevante probleem:

Figuur 10 geeft de 52 regels Hexdump van "ALL 9". In figuur 2A staan de "resterende" STORE-, TABLE- en PTR-adressen. Figuur 2B bevat de "ultieme" Machine-code routines ter oplossing van ons cijferraadsstuk.

BULLETIN SIGG

In het eerste deel van de MC-RTN MAKE SET OF 3 3-DIG-NRS G1 G2 G3 worden de waarden van de tabel gerealiseerd. De 3 * 3 LOOPS gebruiken de registers D/B/C dan E/B'/C' en vervolgens D'/E'/A'. Het gebruiken van de 'exchange' registers wordt zoals gewoonlijk met r' gemarkeerd en van a' met a', het werken met de reguliere registers met r resp. a. De labels en bijschriften bij de routine maken het begrip van het functioneren van de 9 LOOPS gemakkelijk, zodat geen nadere toelichting nodig is.

Als een SET G1 G2 G3 is gevormd, die aan alle kondities voldoet, wordt de rest van het PG ingeschakeld via de CALL van CLEAR op AD 4165. Vooraf zijn DE en BC gestacked, de overige variabelen B', C', D', E' en A' worden alleen in de LOOPS gebruikt. Voor de goede orde: registerpaar H'L' wordt in de routines in het geheel niet gebruikt, daarom mag het programma worden beëindigd op AD 40B2 met de RET-instructie en is geen RST 08 einde nodig.

De resterende RTN-delen CLEAR/MULTIPLY, TEST, STORE en CONVERT zijn fundamenteel gelijk aan wat eerder is gepubliceerd in november 1988 en daarom worden deze zonder labels en bijschriften gegeven. Vanzelfsprekend zijn de CALL adressen aangepast en zijn de niet meer relevante PUSH- en POP-instructies verwijderd. Wel is de volgorde van de delen gewijzigd, namelijk logisch gemaakt. Een en ander heeft tot gevolg dat er minder bytes nodig zijn en het functioneren iets sneller is geworden.

Omdat deze subroutines worden aangeroepen met een CALL, luidt de instructie op AD 41DF RET Z (was JR Z), op AD 41F0 RET NZ (was JR NZ) en op AD 4201 RET (was JR). Deze RETURNS zijn naar label NEXT-SET op AD 4166.

RESULTATEN

De 39 oplossingen van het probleem staan in deel 18, zowel ongesorteerd op AD 5410/567F als gesorteerd en uitgePRINT.

H A N V A N A B B E

FIG. 1C BASIC-PG "ALL9"+"SORT"

```
1 REM-REGEL "ALL9" 416 BYTES
2 RAND USR 16514<>T
4 REM-REGEL "SORT" 193 BYTES
5 RAND USR 16956
6 RAND USR 17068
```

FIG. 1D HEXDUMP LINE 1 FIG. 1C

40B2--2110	5422	FB4F	3E09--238
40B4--5732	0A50	0609	B828--1D2
4092--180E	09B9	2810	78B9--251
409A--280B	A717	1717	1781--1B7
40A2--320B	5018	0C7A	0D20--158
40AA--EA10	E33D	FE06	30D8--426
40B2--C91E	087B	BA30	45B8--351
40BA--2842	B928	3F32	0C50--218
40C2--D906	0978	D9BA	2830--348
40CA--B828	2D89	282A	B828--2F8
40D2--27D9	0E09	79B8	281C--26C
40DA--D9BA	2817	B828	14B9--37F
40E2--2811	B828	0ED9	78A7--322
40EA--1717	1717	6132	0D50--16C
40F2--D918	1109	0020	D0D9--38E
40FA--D910	C8D9	1D7B	FE04--424
4102--30B1	16A1	D916	077A--30A
410A--D9B8	3068	BA28	65B8--428
4112--2862	B928	5FD9	B828--383
411A--5CB9	2859	320E	601E--244
4122--097B	B828	4B59	2848--2D8
412A--BA28	45D9	BA28	40B8--3DA
4132--283D	B928	3A8B	2837--29A
413A--A717	1717	176F	083E--1B8
4142--09BA	2827	B828	24B9--2CF
414A--2821	B828	1ED9	B828--303
4152--19B9	2816	BA28	13B8--2C0
415A--2810	D9B8	320F	6095--2BC
4162--08D5	C5CD	7D41	C1D1--4BF
416A--D908	D93D	20D3	08D9--3CB
4172--1D20	AED9	D915	208F--361
417A--C3F6	4021	FF4F	1100--379
4182--5001	0A00	EDB0	2A0A--22C
418A--5022	0350	623A	0D50--1BE
4192--CDB4	413A	0C50	CDBF--3E4
419A--412E	091E	0406	037E--121
41A2--1236	0020	1D10	F83A--1D4
41AA--0F50	CDB4	413A	0E50--2B9
41B2--180B	4FCD	BD41	791F--2D5
41BA--1F1F	1FE8	0F47	2E09--1D0
41C2--1E04	1A8E	2777	2D1D--1B2
41CA--F2C4	4110	F12E	0106--38D
41D2--04AF	ED6F	2D86	772C--365
41DA--2C10	F67E	A7C8	CD02--3EE
41E2--420E	03CD	0942	0D20--198
41EA--FA21	01FF	ED52	C021--43B
41F2--0050	ED5B	FB4F	0110--2F3
41FA--00ED	B0ED	53FB	4FC9--4F0
4202--1100	007E	CD16	422C--1E0
420A--7ECD	1342	7E1F	1F1F--278
4212--1FE6	0FC8	47AF	3717--320
421A--10FD	3001	1C82	57C9--2FC

FIG. 2A STORE-, TABLE- AND PTR-ADDRESSES "ALL9"

THE STORE AND TABLE ARE WITH DCM(DECIMAL) DIGITS HI...LO:

NUMBERS STORE	5000-500F
(DE): (00) 01/04	(HL): 05/09
G1: 0A/0B	G2: 0C/0D
G3: 0E/0F	
R(ESULTS)-TABLE	5410-567F
PTR(POINTER) R-TABLE	4FFB/C

BULLETIN 566

FIG. 2B ULTIEME MC-RTN "ALL 9"

SET POINTER R(RESULTS)-TABLE

SET-R-PTR 4082--211054 LD HL,5410 HL = 1, AD R-TABLE
4085--22FB4F LD (4FFB),HL store PTR R-TABLE

MAKE SET OF 3 3-DIG-NRS Q1 Q2 Q3

MAKE-Q1	4086--3E09	LD	A,09	
D-LOOP	408A--57	LDD	D,A	D = A = 9....6
	408B--320A50	LDD	(500A),A	store Q11
	408E--0609	LDD	B,09	
B-LOOP	4090--B8	CP	B,09	B = 9....1
	4091--2818	JR	NC,40AB	NEXT-B if B = D
	4093--0E09	LDD	C,09	
C-LOOP	4095--B9	CP	C,09	C = 9....1
	4096--2810	JR	NC,40AB	NEXT-C if C = D
	4098--78	LDD	B,B	
	4099--B9	CP	A,B	
	409A--280B	JR	NC,40A7	NXT-C if C = B
	409C--A7	AND	A,A	→ No Carry
	409D--17	RLA		
	409E--17	RLA		to left
	409F--17	RLA		nibble
	40A1--81	ADD	A,C	C in right nibble
	40A2--320B50	LDD	(500B),A	store Q12,Q13
	40A5--180C	JR	40B3	cont. with MAKE-Q2
NXT-C	40A7--7A	LD	A,D	A = D
NEXT-C	40A8--0D	DEC	C,C	
	40A9--20EA	JR	NZ,4095	C-LOOP if C > 0
NEXT-B	40AB--10E3	DEC	B,B	B-LOOP if B > 0
NEXT-D	40AD--3D	DEC	D,D	lower value of D
	40AE--FE06	CP	06,06	
	40B0--30D8	JR	NC,408A	D-LOOP if D >= 6
	40B2--C9	RET		RETURN to BASIC
MAKE-Q2	40B3--1E08	LD	E,08	
E-LOOP	40B5--78	LDD	D,E	A = E = 8....4
	40B6--BA	CP	D,BA	
	40B7--3045	JR	NC,40FE	NEXT-E if E >= D
	40B9--B8	CP	B,BA	if E = B
	40BA--2842	JR	NC,40FE	if E = C
	40BC--B9	CP	N,40FE	store Q21
	40BD--083F	LDD	(500C),A	r'
	40BF--320C50	LDD	B,09	
B'-LOOP	40C0--78	CP	D,B	A = B' = 9....1
	40C1--09	CP	D,B	
	40C2--BA	CP	D,BA	
	40C3--2830	JR	NC,40FA	NXT-B' if B' = D
	40C4--B8	CP	B,BA	if B' = B
	40C5--282D	JR	NC,40FA	if B' = C
	40C6--B9	CP	N,40FA	if B' = E
	40C7--282A	JR	N,40FA	
	40C8--B8	CP	D,BA	
	40C9--2827	JR	N,40FA	
	40D3--09	EXX		
	40D4--0E09	LDD	C,09	A = C' = 9....1
C'-LOOP	40D6--79	LDD	D,C	
	40D7--B8	CP	N,40F6	NEXT-C' if C' = B'
	40D8--281C	JR	NC,40F5	r'
	40DA--09	CP	D,40F5	NXT-C' if C' = D
	40DB--BA	CP	B,40F5	if C' = B
	40DC--2817	JR	NC,40F5	if C' = C
	40DE--B8	CP	N,40F5	if C' = E
	40DF--2814	JR	N,40F5	
	40E1--B9	CP	A,B	
	40E2--2811	JR	A,B	→ No Carry
	40E4--B8	CP	B,B	to left
	40E5--280E	JR	A,B	nibble
	40E7--09	EXX		
	40E8--78	LDD	D,B	
	40E9--A7	AND	A,B	
	40EA--17	RLA		
	40EB--17	RLA		
	40EC--17	RLA		
	40ED--17	RLA		
	40EE--81	ADD	A,C	C' in right nibble
	40EF--320D50	LDD	(500D),A	store Q22,Q23
	40F0--09	EXX		
	40F3--1811	JR	4105	cont. with MAKE-Q3

BULLETIN SGG

NXT-C'	40F5--D9	EXX		r'	
NEXT-C'	40F6--0D	DEC	C		
	40F7--20DD	JR	NZ,40D6	C'-LOOP if C' > 0	
NXT-B'	40F9--D9	EXX		r'	
	40FA--D9	EXX		r'	
	40FB--10C8	DJNZ	40C5	B'-LOOP if B' > 0	
NEXT-E	40FD--D9	EXX		r'	
	40FE--1D	DEC	E		
	40FF--7B	LD	A,E		
	4100--FE04	CP	04		
	4102--30B1	JR	NC,40B5	E-LOOP if E >= 4	
	4104--18A1	JR	40A7	cont. at NXT-C	
MAKE-03	4106--D9	EXX		r'	
D'-LOOP	4107--1607	LD	D,07		
	4109--7A	LD	A,D	A = D' = 7....1	
	410A--D9	EXX		r'	
	410B--BB	CP	E		
	410C--3068	JR	NC,4176	NXT-D' if D' >= E	
	410E--BA	CP	NON		
	410F--2865	JR	NON,4176	if D' = D	
	4111--BB	CP	NON		
	4112--2862	JR	NON,4176	if D' = B	
	4114--BB	CP	NON		
	4115--285F	JR	NON,4176	if D' = C	
	4117--D9	EXX	Z,4176		
	4118--BB	CP	B		
	4119--285C	JR	NON,4177	NEXT-D' if D' = B'	
	411B--BB	CP	NON		
	411C--2859	JR	Z,4177	if D' = C'	
	411E--320E50	LD	(500E),A	store 031	
E'-LOOP	4121--1E09	LD	E,09		
	4123--7B	LD	A,E	A = E' = 9....1	
	4124--BB	CP	B		
	4125--284B	JR	NC,4172	NEXT-E' if E' = B'	
	4127--BB	CP	NON		
	4128--2848	JR	NC,4172	if E' = C'	
	412A--BA	CP	NON		
	412B--2845	JR	Z,4172	if E' = D'	
	412D--D9	EXX		r'	
	412E--BA	CP	D		
	412F--2840	JR	NZ,4171	NXT-E' if E' = D	
	4131--BB	CP	NON		
	4132--283D	JR	NON,4171	if E' = B	
	4134--BB	CP	NON		
	4135--283A	JR	NON,4171	if E' = C	
	4137--BB	CP	NON		
	4138--2837	JR	NON,4171	if E' = E	
	413A--A7	AND	D	No Carry	
	413B--17	RLA		to left	
	413C--17	RLA		nibble	
	413D--17	RLA			
	413E--17	RLA			
	413F--6F	LD	L,A	store briefly in L	
A'-LOOP	4140--08	EX	AF,AF"	A'	
	4141--3E09	LD	A,09	A' = 9....1	
	4143--BA	CP	NON		
	4144--2827	JR	NON,416D	NEXT-A' if A' = D	
	4146--BB	CP	NON		
	4147--2824	JR	NON,416D	if A' = B	
	4149--BB	CP	NON		
	414A--2821	JR	NON,416D	if A' = C	
	414C--BB	CP	NON		
	414D--281E	JR	NON,416D	if A' = D'	
	414F--D9	EXX	Z,416D		
	4150--BB	CP	B		
	4151--2819	JR	NON,416C	NXT-A' if A' = B'	
	4153--BB	CP	NON		
	4154--2816	JR	NON,416C	if A' = C'	
	4156--BA	CP	NON		
	4157--2813	JR	NON,416C	if A' = D'	
	4159--BB	CP	NON		
	415A--2810	JR	NON,416C	if A' = E'	
	415C--D9	EXX		r'	
	415D--85	ADD	A,L	Join nibbles E' A'	
	415E--320F50	LD	(500F),A	store 032,033	
	4161--95	SUB	L	A' = right nibble	
	4162--08	EX	AF,AF"	A'	
NEXT-SET	4163--05	PUSH	DE	stack values	
	4164--C5	PUSH	BC	4 variables	
	4165--CD7D41	CALL	417D	CALL CLEAR etc.	
	4168--C1	POP	BC	retrieve values	
	4169--D1	POP	DE	4 variables	

BULLETIN SGG

	416A--D9	EXX	AF,AF"	r'
	416B--08	EXX		r'
NXT-A'	416C--D9	DEC	A	r'
NEXT-A'	416D--3D	JR	NZ,4143	A'-LOOP if A' > 0
	416E--20D3	EXX	AF,AF"	r'
	4170--08	EXX		r'
NXT-E'	4171--D9	DEC	E	r'
NEXT-E'	4172--1D	JR	NZ,4123	E'-LOOP if E' > 0
	4173--20AE	EXX		r'
	4175--D9	EXX		r'
NXT-D'	4176--D9	DEC	D	r'
NEXT-D'	4177--15	JR	NZ,4109	D'-LOOP if D' > 0
	4178--208F	EXX		r'
	417A--C3F640	JR	40F6	cont. at NEXT-C'

MULTIPLY DECIMALLY G1*G2*G3

```

CLEAR
417D--21FF4F    LD    HL,4FFF
4180--110050    LD    DE,5000
4183--010A00    LD    BC,000A
4186--EDB0      LDIR

MULTIPLY
4188--2A0A50    LD    HL,(500A)
418B--220350    LD    (5003),HL
418E--62        LD    H,D
418F--3A0D50    LD    A,(500D)
4192--CDB441    CALL 41B4
4195--3A0C50    LD    A,(500C)
4198--CDBF41    CALL 41BF
419B--2E09      LD    L,09
419D--1E04      LD    E,04
419F--0503      LD    B,03
41A1--7E        LD    A,(HL)
41A2--12        LD    (DE),A
41A3--3600      LD    (HL),00
41A5--2D        DEC    L
41A6--1D        DEC    E
41A7--10F8      DJNZ 41A1
41A9--3A0F50    LD    A,(500F)
41AC--CDB441    CALL 41B4
41AF--3A0E50    LD    A,(500E)
41B2--180B      JR     41BF
  
```

```

MULT-BYTE
41B4--4F        LD    C,A
41B5--CDBD41    CALL 41BD
41B8--79        LD    A,C
41B9--1F        RRA
41BA--1F        RRA
41BB--1F        RRA
41BC--1F        RRA
41BD--E60F      AND    0F
41BF--47        LD    B,A
41C0--2E09      LD    L,09
41C2--1E04      LD    E,04
41C4--1A        LD    A,(DE)
41C5--8E        ADC    A,(HL)
41C6--27        DAA
41C7--77        LD    (HL),A
41C8--2D        DEC    L
41C9--1D        DEC    E
41CA--F2C441    JP     41C4
41CD--10F1      DJNZ 41C0
41CF--2E01      LD    L,01
41D1--0604      LD    B,04
41D3--AF        XOR    A
41D4--ED6F      RLD
41D6--2D        DEC    L
41D7--86        ADD    A,(HL)
41D8--77        LD    (HL),A
41D9--2C        INC    L
41DA--2C        INC    E
41DB--10F6      DJNZ 41D3
  
```

TEST RESULT: "ALL 9 ?"

```

41DD--7E        LD    A,(HL)
41DE--A7        AND    A
41DF--C8        RET    Z
41E0--CD0242    CALL 4202
41E3--0E03      LD    C,03
41E5--CD0942    CALL 4209
41E8--0D        DEC    C
41E9--20FA      JR     NZ,41E5
41EB--2101FF    LD    HL,FF01
41EE--ED52      SBC    HL,DE
41F0--C0        RET    NZ
  
```

STORE RESULT AND FACTORS

```

41F1--210050    LD    HL,5000
41F4--ED58F84F LD    DE,(4FFB)
41F8--011000    LD    BC,0010
41FB--EDB0      LDIR
41FD--ED53F84F LD    (4FFB),DE
4201--C9        RET
  
```

CONVERT DCM DIGITS INTO BITS 3 OR 2 NIBBLES

```

4202--110000    LD    DE,0000
4205--7E        LD    A,(HL)
4206--CD1642    CALL 4216
4209--2C        INC    L
420A--7E        LD    A,(HL)
420B--CD1342    CALL 4213
420E--7E        LD    A,(HL)
420F--1F        RRA
4210--1F        RRA
4211--1F        RRA
4212--1F        RRA
4213--E60F      AND    0F
4215--C8        RET    Z
4216--47        LD    B,A
4217--AF        XOR    A
4218--37        SCF
4219--17        RLA
421A--10FD      DJNZ 4219
421C--3001      JR     NC,421F
421E--1C        INC    E
421F--82        ADD    A,D
4220--57        LD    D,A
4221--C9        RET
  
```


ESGEEGEET JES



Te koop aangeboden:

Rubber Speccie in doos met toebehoren f150,00
 Voyager 7 Viditel - Info Modem f100,00 met Tape Fido.
 kabeltje + stekkertje voor de 128K f25,00
 Flora tel: 050:263930

Te koop aangeboden:

QL (MINERVA-ROM) F260,-	* t.v.-splitter f24,00
QL (JS-ROM) f 220,00	* Transformboxen f27,-
QL (JM-ROM) f 200,00	* EPROM-Toolkit II+handl f55,-
QL-APL-EPROM F100,-	* Connectoren v Canon f37,50
Philips 80 monitor groen f150,-	* JOYSTICK F25,-

Fred Vink tel: 02230-34250

Te koop aangeboden:

Een Teac 3/5 inch Dubbel zijdig Discdrive 80 tracks f100,00
 10 3/5 inch disc's DS voor f32,50 BASF
 Rudy Biesma tel: 05920-50643

Wie kan mij helpen aan het blad YOUR SINCLAIR Juli 1989.

Bel Johan Koning tel: 05946-12807.

Te koop aangeboden:

Spectrum 48K in Lo Profile toetsenbord, Interface I + 2
 microdrives + Sanyo recorder + Timex 2040 printer.
 Martin den Hollander tel: 05978-45474.

Te koop gevraagd:

Een origineel SEIKOSHA LINT met inktkussentje.
 Bel H. van Abbe tel: 01751-14216

Te koop aangeboden:

110 schijven 3,5 SD vol met Opusprogramma's inclusief drie
 doosjes, prijs nader overeen te komen.
 Graag bellen na zes uur, tel: 05490-21091 Wim Altena.

Te koop aan geboden

+D disk interface met disk drive en voeding
 ZX SPECTRUM 128K
 Diverse software oa. Operation Wolf, Last ninja 2, Robocop.
 Edwin Blink, Tel.: 050 - 418888

DRUKWERK

AFZ:

REDAKTIE SGG:

Mevr. F. Elstrodt
Kam. Onnestraat 172
9727 HS Groningen

PORT BETAALD
GRONINGEN

Elstrodt, F
K. Onnestraat 172
9727 HS Groningen