

BULLETIN

SINCLAIR GEBRUIKERSGROEP GRONINGEN/ASSEN



Beste redactie,
Voordat ik met vakantie
ga, stuur ik U mijn copy
voor het september
bulletin.
Vriendelijke groeten.

J. J. J.

COLOFON

VOORZITTER:

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

SEKRETARIS:

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

PENNINGMEESTER:

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

**VICE VOORZITTER/
PENNINGMEESTER:**

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

VERHUUR:

C. van Krimpen
Koldakker 34
9407 BM Assen
☎ 05920-70093

REDAKTIE:

Mevr. F. Elstrodt
Kam. Onnesstraat 172
9727 H5 Groningen
☎ 050-263930

Rudy Blesma
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.

SLUITINGSDATUM KOPIJ 1 SEPTEMBER

VAN DE REDAKTIE



HALLO ALLEMAAL

Een extra dik bulletin voor de vakantie, de redactie is heel erg blij met zoveel leuke copy. We concluderen hieruit dat ons bulletin leeft.

Wij zijn erg tevreden met de aangeleverde kopij, we hopen dat dit na de vakantie zo door blijft gaan. Een van de vaste schrijvers is de heer van Abbe en van hem kregen wij het verzoek een oproep te plaatsen over enkele onderdelen. Hij zoekt een ZX81 geheugenuitbreiding van 16 of 32K. Telefonisch te bereiken : 01751 - 14216.

Tijdens het doorbladeren van ons bulletin 5e jaargang, kwamen we onderstaande namen tegen bij de ingezonden artikelen, listings, cursussen, etc. Wij danken hen voor hun bijdrage.

Jan Kloosterman * Martin den Hollander * Orm Heerkens * Jan Smidhoff * J.D. Burggraaf * Arie Lautenbach * Bert v.d. Zaag * Herman Vesper * Han v. Abbe * Bert Koning * Bert Westenberg * Ramon v. Alteren * Kees v. krimpen * Harm Scharft * drs.P.C.Sextrum * Frans Postma * Roland van Leusden * F.Grunefeld * Carlo Delhez * Stef Kroon * Roelof Koning * J.v.Alteren *

We willen U nog even wijzen op de aankomende ledenvergadering, in het meinumner staat hier meer over. Na de vergadering demonstreert Stef Kroon zijn digitale voltmeter.

LET OP op 25 juni houdt de Sinclair GG van de HCC een gebruikersdag van 10.00 tot 16.00 in "de Bron" te Utrecht.

De redactie weet nog geen data van gebruikersavonden voor september, vandaar dat de sluitingsdatum van 1 september maar een gok is. Heeft U de kopy klaar wilt U het dan inleveren zodat wij niet tegen de klok hoeven te racen .

-In september wil Ramon beginnen met een adventure rubriek.

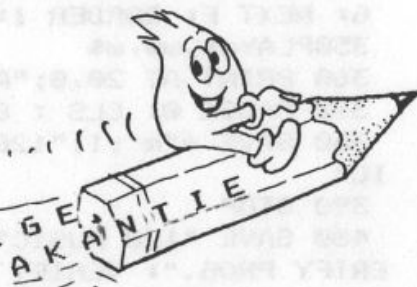
-Verder krijgt U een overzicht van alle ruilabbonnementen.

In dit nummer kunt U o.a lezen:

- Music voor de 128.
- 120 kleuren voor de 128.
- 64 kleuren voor de spectrum.
- Iets over Talen.
- Een Databankje voor de Opus.
- Vergroot.
- 3D letters.
- Jaaroverzicht 5e jaargang.
- Snel en precies rek. met correctie deel 15.
- Uitleg en tips voor gebruik v.d. Seikosha SP-1000.

Wij wensen U veel leesplezier en tot

DONDERDAG 23 JUNI in de Weijert



BULLETIN SGG

Weer een leuk stukje music voor de 128 ingestuurd door
Frans Postma.

128 MUSIC

```
10 BORDER 1: PAPER 4: INK 0
20 LET i$="M7T145UX1000W005N3Eabe"
30 LET d$="06eeeeeee"
40 LET T$="(V1105N6D#C5b6ag5#f6e#f5g9a6D#C5b6ag5#f6e#f5e9d
)"
50 LET M$="(V1305N5eg3#fed5e3g#fe5d)(V11N5$eg3f$ed5$e3gf$e
5d)"
60 LET k$="(UX1000W6N3$ED$Eg)"
70 LET J$="M7UX1000W005N9&&(3b#CD#Fb#CD#Fb#CD#Fb#CD#F)"
80 LET e$="#C#C#CDDDbb"
90 LET U$="(V905D#fa#Cgabg04AD#FGbD#FaEgb#FaDGbA#CEA#CEA#C
05D#fa#Cgabg04AD#FGbD#FaEgb#FaDEg#faDgaDeD)"
100 LET n$="(V1304N5CE3DCb5C3EDC5b)(V11N5C$E3DC$b5C3$EDC5$b
"
110 LET I$="(03N3C&8&3&5$a7$b3&)"
120 LET o$="M7UX1000W005N9&&&&3gabDgabDgabDgabD"
130 LET s$="aAaAaAaA"
140 LET x$="04N9D"
150 LET V$="(UX1000W003dDDd1DDDD3dD)"
160 LET y$="03N9D"
170 LET P$="UX200W3N7ab6C5D3E)V13N7ccccccc"
180 LET z$="V1303N9d"
190 LET Q$="(02N3C&8&3&5$a7$b3&)"
200 LET f$="": FOR f=1 TO 12: LET f$=f$+i$: NEXT f
210 LET f$=f$+d$+d$+d$+d$
220 LET g$=j$+e$+e$+e$+e$
230 LET h$=o$+s$+s$+s$+s$
240 LET a$=f$+t$+m$+k$+k$+k$+k$+f$+t$+x$
250 LET b$=g$+u$+n$+i$+g$+u$+y$
260 LET c$=h$: FOR f=1 TO 8: LET c$=c$+v$: NEXT f
270 LET c$=c$+p$+q$+h$
280 FOR f=1 TO 9: LET c$=c$+v$: NEXT f
290 LET A$="V15"+A$
300 LET B$="V13"+B$
310 LET C$="V9"+C$
320 CLS
330 FLASH 0: PRINT AT 4,4;"MADE BY: Frans Postma ";AT 7,10
;1988"
340PLAYa$,b$,c$: FOR F=1 TO 9: FOR G=1 TO 7: BORDER G: NEXT
G: NEXT F: BORDER 1: INPUT ""
350PLAYw$,w$,w$
360 PRINT AT 20,0;"ANY KEY..TO PLAY AGAIN...."
370 PAUSE 0: CLS : GO TO 10
380 SAVE *"m";1;"128 MUSIC" LINE 10: VERIFY *"M";1;"128 MUS
IC"
390 STOP
400 SAVE "128 MUSIC" LINE 1: CLS : PRINT "REWIND TAPE ,TO V
ERIFY PROG.": VERIFY "128 MUSIC"
```

VAN DE VOORZITTER

Onze tweede vergadering van het seizoen wordt op de komende gebruikersavond gehouden.

Een paar zaken vragen natuurlijk de aandacht. Ten eerste zal voor het eerst de kascommissie verslag uitbrengen van hun bevindingen over de toestand van de kas. Als ik dit intyp zijn de heren al bij onze penningmeester geweest en hebben volledig inzage gekregen in het geheel. Het resultaat horen we op de vergadering. Verder hebben we getracht statuten op te stellen en zo als u weet konden en kunnen deze tot voor de vergadering ingezien worden door U. Als de vergadering er mee akkoord gaat, zullen we nog voor het nieuwe seizoen begint er dan voor zorgen dat we naar de notaris gaan. We starten in september dan als officiële vereniging.

Ook is er een bestuursverkiezing. Als bestuur vinden we het beter dat er elk jaar en volgens een bepaald rooster in ieder geval twee bestuursleden aftreden. Men kan uiteraard zich herkiesbaar stellen.

Een bestuur bestaande uit zeven mannen en/of vrouwen lijkt ons het meest geschikt.

Hieruit blijkt dat er op dit moment nog een bestuurslid gezocht wordt door ons. U kan natuurlijk ook op eigen initiatief zich bij ons melden.

Getracht zal worden ook deze avond voor wat betreft het vergadergedeelte dit zo kort mogelijk te houden opdat ook de kleine demonstratie verzorgd door Stef Kroon de volledige aandacht zal kunnen krijgen.

Verder wens ik U en Uw huisgenoten een plezierige zomervakantie toe.

J.D. Burggraaf



120 kleuren op een SPECTRUM 128

Dit programma werkt alleen op de 128 omdat het gebruik maakt van het tweede scherm dat deze heeft.

Het principe is vrij eenvoudig, je neemt twee plaatjes, de ene zet je in het normale schermgeheugen (adres 16384), de andere zet je in het tweede schermgeheugen (adres 49152, bank 7). Nu laat je SPECTRUM deze schermgeheugens zeer snel, afwisselend inschakelen hierdoor vloeien de plaatjes samen en ontstaan nieuwe (meng)kleuren. Ook ben je niet meer gebonden aan twee kleuren per karakterpositie maar nu zijn in zo'n blokje vier kleuren mogelijk.

Voer het programma in, RUN het en wacht af.

```
10 CLEAR 32767
20 FOR f=32768 TO 32849
30 READ a
40 POKE f,a
50 NEXT f
60 RANDOMIZE USR 32768
100 DATA 1,253,0,62,23,237,121,
33,0,64,17,0,192,1,0,24,54,240,6
2,204,18,35,19,11,120,177,32,244
,1,0,3,17,0,216,33,0,88,121,230
110 DATA 127,119,121,15,15,15,2
30,127,18,11,120,177,35,19,32,23
9,1,253,127,30,23,237,89,118,123
,238,8,95,175,219,254,238,31,230
,31,40,240,58,92,91,237,121,201
```

De ZX Spectrum kent slechts 8 kleuren. Door middel van een aardige software-truc, kunnen echter 64 kleurtinten worden verkregen, waarbij dat helaas alleen geldt voor user defined graphics. Toch zijn daar heel aardige dingen mee mogelijk, zoals bijvoorbeeld een bijzonder gekleurde omkadering. Het bijgaande voorbeeld programma, geschreven door onze medewerker Pandit Reuvers, laat zien hoe die 64 tinten te verkrijgen zijn. In regel 40 ziet u een blokje, dat bestaat uit 8 x 8 (= 64) puntjes. Het aardige is nu, dat door het geven van een verschillende kleur aan die puntjes, merkwaardige kleurcombinaties ontstaan, die normaal op de Spectrum niet voorkomen. In het voorbeeld programma worden al die kleurcombinaties doorlopen om te laten zien wat er allemaal mogelijk is.

```
10 REM *****
15 REM ** 64 kleuren op **
20 REM ** SPECTRUM **
25 REM ** P.M REUVERS **
30 REM ** Voor R.A.M **
35 REM *****
40 REM ** GRAPHIC A **
45 RESTORE 50
50 DATA 170,65,170,65,170,65,1
70,65
60 FOR a=0 TO 7: READ b: POKE
USR "a":a,b: NEXT a
70 FOR a=0 TO 7
80 FOR b=0 TO 7
90 PRINT INK a; PAPER b;" "
100 NEXT b: NEXT a
```

64 KLEUREN OP DE SPECTRUM.



IETS OVER TALEN

Ons alfabet is in twee delen gerangschikt in de computer: de hoofdletters van ASCII waarde 65 t/m 90 en de kleine letters van 97 t/m 122.

Behalve deze volgorde is er eigenlijk geen logische rangschikking. Anders ligt dat bij het lettergrepen-schrift zoals *Hindi* en *Sanskrit*.

Hier zijn vijftientig konsonanten (medeklinkers) in vijf groepen van vijf verdeeld.

k kh g gh ñ	harde gehemelte klanken
c ch j jh ñ	zachte gehemelte klanken
ṭ ṭh ḍ ḍh ṇ	tongklanken
t th d dh n	tandklanken
p ph b bh m	lipklanken
y r l v	halfklinkers
ś ṣ s	sisklanken
h	toonloze letter

Het is dan mogelijk om d.m.v. nummers van de letters (K t/m Jh = 1 t/m 9, T t/m Dh = 1 t/m 9, P t/m M = 1 t/m 5 en Y t/m H = 1 /8 en N=0) de nummerreeks van een woord te bepalen.

Een voorbeeld is de 72 toonsoorten van het Zuid-Indische muzieksysteem. Om het nummer te bepalen worden de eerste twee lettergrepen van het woord genomen, het nummer erbij gezocht en dan omgedraaid.

Ramapriya: R=2 en M=5 dus 25. Omgedraaid 52. Onder welke Ramapriya in dat systeem voorkomt.

Astanayika
(The eight heroines)

Tweede Nayika

Virahotkantita: viraha-separation
utkantita-longing for beloved persons or things
Woman distressed by absence of lover/ husband

Bereft of lover's dalliance
She languishes in pain;
And all her youthful radiance
She deems but cold and vain,
Unsought, her tender beauteous form
And dishevelled her hair;
No lover clasps her rounded arm
And sorrow is her share



A.P. Lautenbach

EEN DATA BANKJE VOOR DE OPUS

Er zijn voor het opslaan van je programma namen al een heleboel programma's geschreven. Daar heb ik mij dan ook niet mee beziggehouden. In het DUC-magazine No. 10 staat er ook een met de verzuchting, "Ik heb wel het geduld om zo'n programma te maken maar om alle gegevens in te voeren niet."

Nu, daar is dit programma voor. Binnen een half uur had ik 806 namen ingevoerd die op 38 verschillende schijfjes stonden. In elk bestandje staat dan: 1. de naam, 2. soort progr. (basic, mach. code.....), 3. naam van de disk. Verder is er een optie om elk bestand in te delen bij een groep v.b. spel, tekenprogr. screens..enz. Op elk van deze onderdelen kan worden geselecteerd.

Het progr. is een samenvoeging van twee bestaande n.l.: Nuttincat, deze stond op DUCDISK 2 en telefoonklapper als listing afgedrukt in Sinclair Gebruiker jaargang 2 no. 1. Ik heb het ingericht voor 1000 bestandjes en 50 disk's maar er zijn nog 18000 bytes over in mijn 48K.

Het is misschien heel goed mogelijk om de selectie-mogelijkheden uit te breiden. Zodat je b.v. ook de basic-programma's van een schijf kunt krijgen of misschien een alfbetiserings-routine. Het zou ook best mogelijk moeten zijn om de gegevens in te lezen in een RAF-bestand. Mogelijkheden om daar in de komende vakantie over na te denken.

Van het hoofdmenu zal ik alleen punt -h- toelichten. De andere punten spreken dacht ik voor zichzelf. Heb je een schijfje met een gewijzigde inhoud dan gaat het als volgt: Het schijfje gaat in de drive en je slaat toets 2 aan. Het oude bestand wordt dan gewist en het nieuwe komt helemaal achter in de file te staan. Er zit dan een gat in de file, deze kun je weghalen met optie -h-. Verder is het misschien wel handig om programma met data samen weg te schrijven en te laden.

HOOFD MENU

- 1 inlezen bestanden
- 2 wijzigen
- h herschikken(na wijziging(en))
- 3 zoeken op naam
- 4 zoeken op soort
- 5 zoeken op disk
- 6 zoeken op groep
- i indelen bij een groep
- 7 saven bestand
- 8 saven programma
- 9 laden bestand
- uitvoer sch. printer nu:scherm

Er is ruimte voor 1000 namen
Daarvan is 806 gebruikt.

Als iemand belangstelling heeft voor dit programma, mijn naam en adres staan onder dit stukje.

F.Grunefeld, Ommelandersdrift 27, Bedum. Tel. 05900-13505

LISTING DATABANKJE

```

10 RANDOMIZE USR 64900
20 DEF FN a(x)=PEEK (x+a)+256*PEEK (a+x+1): GO SUB 8005
30 LET aantnamen=n
40 LET a=62005: DIM d$(10): LET c=a-1: FOR d=1 TO 10: LET d$(d)
=CHR$ PEEK (a+d): NEXT d
50 GO SUB 8400
60 IF wijz THEN GO TO 2000
70 LET a=62016: LET i=63800
80 LET b=FN a(0): LET s=FN a(2): LET e=FN a(4)
90 LET mem=(e-s)*256+b-6
100 PRINT AT 20,4;"Even geduld!": IF e=65535 THEN CLS : GO TO wa
chten
110 DIM a$(10): FOR p=a+6 TO a+15: LET a$(p-5-a)=CHR$ PEEK p: NE
XT p
130 LET t=a: LET a=i: LET type=PEEK a
140 LET length=FN a(1): LET start=FN a(3): LET basic=FN a(5)
170 IF type=0 THEN LET n$(n,11)="1"
180 IF type=1 THEN LET n$(n,11)="2"
190 IF type=2 THEN LET n$(n,11)="3"
200 IF type=3 THEN LET n$(n,11)="4"
210 IF type>3 THEN LET n$(n,11)="5"
220 LET a=t: LET a=a+16: LET i=i+10
1007 IF n>aantnamen THEN RETURN
1020 REM file naam
1030 LET n$(n,1 TO 10)=a$( TO )
1100 REM disk naam
1120 IF w>aantdisk THEN RETURN
1130 LET n$(n,12)=CHR$ w: LET w$(w)=d$
1150 LET n=n+1: LET aantnamen=aantnamen+1: GO TO 80
2000 REM Wijzigen?
2010 CLS : PRINT TAB 5;"gegevens wijzigen"
2020 FOR n=1 TO aantnamen
2030 IF CODE n$(n,12)=w THEN LET n$(n,1 TO 10)="leeg": NEXT n
2040 NEXT n: LET wijz=0: GO TO 10
2100 PRINT AT 20,4;"even geduld": REM herschikken in n$
2110 LET a=1: FOR n=1 TO aantnamen
2120 IF n$(a, TO 4)="leeg" THEN LET a=a+1: GO TO 2120
2130 PRINT AT 20,16;n: LET n$(n,1 TO 13)=n$(a,1 TO 13): LET a=a+1
: NEXT n
2380 GO TO wachten
2500 REM zoeken op groep
2510 INPUT "Welke groep? yteks.tek.spel.      hulp. leer. algemeen
.osubrout. probeersels. ";g$
2520 CLS : FOR n=1 TO aantnamen: IF n$(n,13)=g$ THEN GO SUB print
naam: NEXT n
2530 NEXT n
2550 GO TO wachten
2600 REM indelen
2610 CLS : PRINT TAB 11;"Groepen""y tekst progr.""t teken prog
r.""o subroutines""h hulp progr.""l leer progr.""a algemeen"
"s spel progr.""p probeersels"
2620 PRINT ""Geef het programma""1 van deze letters.""ENTER is
niets of maakt leeg"
2630 FOR n=1 TO aantnamen: IF CODE n$(n,12)<>w THEN NEXT n
2640 IF n$(n,1)=" " THEN LET gr=0: GO TO wachten
2650 PRINT AT 16,0: GO SUB printnaam
2660 PRINT AT 18,0;n$(n,13)

```

BULLETIN SGG

```
2670 INPUT "groep? ";i$: PRINT AT 18,0;i$
2680 LET n$(n,13)=i$: PRINT AT 17,0;"
"
2690 NEXT n
2990 STOP
3000 REM zoeken op naam
3010 CLS : PRINT TAB 8;"Zoeken op naam"
3020 PRINT ,,,,"Welke naam?"
3030 INPUT i$
3033 IF i$="menu" THEN RETURN
3034 IF i$="" THEN RETURN
3035 PRINT AT 20,0;" "
3037 PRINT AT 5,0;
3040 IF LEN i$<10 THEN GO TO 3100
3050 PRINT AT 20,0;"Opgegeven naam is te lang"
3100 CLS : FOR n=1 TO aantnamen
3110 IF n$(n,1 TO (LEN i$))=i$ THEN GO SUB printnaam
3190 NEXT n: GO TO wachten
4000 REM zoeken op soort
4005 CLS : PRINT TAB 6;"Zoeken op soort": PRINT : PRINT
4010 PRINT ""1 Basic""2 Num.array""3 Cha.array""4 Mach.code"
"5 Prinfile"
4060 PRINT : PRINT "Van welk soort wil je alle namen gelist
zien?"
4080 INPUT i: IF i<1 OR i>5 THEN GO TO 4080
4100 CLS : FOR n=1 TO aantnamen
4110 IF n$(n,1)=" " THEN GO TO wachten
4120 IF CODE n$(n,11)-48=i THEN GO SUB printnaam
4140 NEXT n: GO TO wachten
5000 REM zoeken op disknaam
5010 CLS : PRINT TAB 5;"Zoeken of indelen""in groepen op disknaa
m"
5020 PRINT ,,,,"Van welke disk"
5030 INPUT i$: IF i$="menu" THEN RETURN
5035 IF LEN i$=0 OR LEN i$>LEN w$(1) THEN GO TO 5030
5050 FOR w=1 TO aantdisk: IF w$(w,1 TO (LEN i$))=i$ THEN GO TO 510
0
5060 NEXT w
5070 PRINT ,,,,"Komt niet voor in dit programma.": GO TO wachten
5100 IF gr THEN GO TO 2600
5105 CLS : FOR n=1 TO aantnamen
5110 IF CODE n$(n,12)=w THEN GO SUB printnaam
5120 NEXT n: GO TO wachten
8005 FOR n=1 TO aantnamen
8010 IF n$(n,1 TO 10)=" " THEN RETURN
8020 NEXT n
8040 PRINT AT 5,0;"Programma vol": GO TO wachten
8110 PRINT #0;"Druk op een toets ": PAUSE 0: GO TO menu
8300 REM printen scherm
8305 PRINT n$(n,1 TO 10);: PAPER 5: PRINT TAB 11;s$(VAL n$(n,11))
;: PAPER 7: PRINT TAB 21;w$(CODE n$(n,12)): RETURN
8405 FOR w=1 TO aantdisk: IF w$(w)=" " OR w$(w)=d$ THEN
RETURN
8420 NEXT w
8430 PRINT AT 17,0;"Geen plaats meer voor een disk": GO TO wachten
8500 REM printen printer
8520 IF 1 THEN LPRINT "
```



```
8540 LPRINT " ";n$(n,1 TO 10);" ";s$(VAL (n$(n,11))); " ";w$(COD
E n$(n,12))
8550 IF 1 THEN LET 1=0: LPRINT CHR$(10): RETURN
8560 LET 1=1
8580 RETURN
9000 CLS : PRINT TAB 9;"HOOFD MENU": PRINT : PRINT
9010 PRINT "1 inlezen bestanden"
9020 PRINT "2 wijzigen"
9025 PRINT "h herschikken(na wijziging(en))"
9030 PRINT "3 zoeken op naam"
9040 PRINT "4 zoeken op soort"
9050 PRINT "5 zoeken op disk"
9055 PRINT "6 zoeken op groep"
9057 PRINT "i indelen bij een groep"
9060 PRINT "7 save bestand"
9070 PRINT "8 save prog."
9080 PRINT "9 laden bestand"
9092 PRINT "uitvoer sch. printer nu:";p$
9095 PRINT AT 16,0;"Er is ruimte voor 1000 namen""Daarvan is ";n
;" gebruikt"
9099 LET i$=INKEY$
9100 IF i$="1" THEN GO TO 10
9105 IF i$="2" THEN LET wijz=1: GO TO 10
9110 IF i$="3" THEN GO TO 3000
9112 IF i$="h" THEN GO TO 2100
9115 IF i$="4" THEN GO TO 4000
9120 IF i$="5" THEN GO TO 5000
9125 IF i$="6" THEN GO TO 2500
9127 IF i$="i" THEN LET gr=1: GO TO 5000
9130 IF i$="7" THEN GO TO 9310
9135 IF i$="8" THEN CLS : PRINT AT 18,0; BRIGHT 1;"Pasop bestand
wordt gewist"; BRIGHT 0;"Gebruik BREAK om te redden""elke ander
e toets gaat door": PAUSE 0: CLEAR : GO TO 9320
9136 IF i$="s" THEN LET p$="scherm ": PRINT AT 14,24;p$: LET prin
tnaam=8300
9137 IF i$="p" THEN LET p$="printer": PRINT AT 14,24;p$: CLOSE #3
: OPEN #3;"b": LET printnaam=8500
9140 IF i$="9" THEN GO TO 9300
9160 GO TO 9099
9230 CLEAR 61999: PRINT "LADEN VAN MACHINE CODE": LOAD *1;"nuttin
code"CODE
9240 DIM s$(5,10): LET s$(1, TO )="Basic": LET s$(2, TO )="Num.ar
ray": LET s$(3, TO )="Cha.array": LET s$(4, TO )="Mach.code": LET
s$(5, TO )="Printfile"
9250 LET p$="scherm ": LET aantnamen=1000: LET aantdisk=50: LET w
achten=8100: LET menu=9000: LET printnaam=8300
9270 LET n=1: LET gr=0: LET 1=0
9280 LET wijz=0: DIM n$(aantnamen,13): DIM w$(aantdisk,10): GO TO
menu
9300 LET aantnamen=aantnamen: LOAD *1;"diskdata" DATA n$(): LOAD
*1;"diskdaw" DATA w$(): GO SUB 8005: LET aantnamen=n: GO TO menu
9310 SAVE *1;"diskdata" DATA n$(): SAVE *1;"diskdaw" DATA w$(): G
O TO 9000
9320 SAVE *1;"DISKDATA" LINE 9230: SAVE *1;"nuttincode"CODE 64900
,100: VERIFY *1;"DISKDATA": VERIFY *1;"nuttincode"CODE : GO TO 92
40
9400 FOR n=1 TO aantnamen: PRINT n;TAB 4;n$(n,1 TO 10);TAB 15;n$(
n,11);" ";w$(CODE n$(n,12));TAB 31;n$(n,13): NEXT n: GO TO menu
```

BULLETIN SGG

Op deze twee bladzijden vindt U twee leuke programma's van
Bert v.d. Zaag.

Typ de programma's in en kijk toe.

VERGROOT

```

  SSSS      SSSS      SSSS
S  SSSS    S  SSSS  S  SSSS  S
S  SSSS    S  SSSS  S  SSSS  S
S  SSSS    S  SSSS  S  SSSS  S
  SSSS      SSSS      SSSS

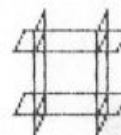
```

© 1988 BWZ SOFTWARE PUBLICATIONS

```

10 RESTORE 20: FOR L=0 TO 1E9: READ R: POKE 32000+L,R: IF
R<>201 THEN NEXT L
20 DATA 33,0,72,17,152,62,6,8,26,205,100,125,197,1,25,0,9
,193,19,5,32,-14,201
30 RESTORE 40: FOR L=0 TO 1E9: READ R: POKE 32100+L,R: IF
R<>201 THEN NEXT L
40 DATA 203,127,196,200,125,35,203,119,196,200,125,35,203
,111,196,200,125
50 DATA 35,203,103,196,200,125,35,203,95,196,200,125,35,2
03,87,196,200,125
60 DATA 35,203,79,196,200,125,35,203,71,196,200,125,201
70 RESTORE 80: FOR L=0 TO 1E9: READ R: POKE 32200+L,R: IF
R<>201 THEN NEXT L
80 DATA 213,229,245,17,88,255,14,8,26,119,36,19,13,32,-7,
241,225,209,201
90 RESTORE 100: FOR L=0 TO 55: READ R: POKE 32256+L,R: NEX
T L: PRINT AT 21,0,: RANDOMIZE USR 32256
100 DATA 33,24,126,6,32,126,245,62,2,229,213,197,205,1,22,1
93,209,225,241,215,35,16,-18,201
110 DATA 127,32,49,57,56,56,32,66,87,90,32,83,79,70,84,87,6
5,82,69,32,80,85,66,76,73,67,65,84,73,79,78,83
120 FOR A=0 TO 20: POKE 32204,88+A*8
130 POKE 32001,4: POKE 32004,152: RANDOMIZE USR 32000: POKE
32001,11: POKE 32004,56: RANDOMIZE USR 32000: POKE 32001,18
: RANDOMIZE USR 32000
140 PAUSE 20: NEXT A: GO TO 120
1000 LOAD *"m";1:"udgcode"CODE USR "a": RUN
1001 SAVE "GROOT": SAVE "udgcode"

```

3 D LETTERS

© 1988 BUZ SOFTWARE PUBLICATIONS

```

10 GO SUB 9000: REM MACHINETAAL-ROUTINES
20 LET A=33: LET AD=32056: LET G=8: REM G=GROOTTE
30 PRINT AT 1,1;CHR$ A
40 PRINT AT 21,0;: RANDOMIZE USR 32000
50 FOR Y=8 TO 15
60 FOR X=8 TO 15
70 IF POINT (X,175-Y)=1 THEN LET X1=X: LET Y1=Y: GO SUB 130
80 NEXT X: NEXT Y: LET A=A+1-(95 AND A=127)
90 FOR L=0 TO 1
100 POKE AD+1,159 AND L: POKE AD+2,88+(2 AND L): POKE AD+6,56 AND L
D L: POKE AD+13,35+(8 AND L): POKE AD+29,9 AND L=0
110 RANDOMIZE USR AD: RANDOMIZE USR 32093+(13 AND L)
120 NEXT L: GO TO 30
130 PLOT X*G,150-Y*G: IF POINT (X+1,175-Y)=1 THEN DRAW G,0: PLOT
X*G+G/2,150-Y*G+G: DRAW G,0: GO SUB 200
140 PLOT X*G,150-Y*G: IF POINT (X,174-Y)=1 THEN DRAW 0,-G: PLOT
X*G+G/2,150-Y*G+G: DRAW 0,-G: GO SUB 230
150 PLOT X*G,150-Y*G: IF (POINT (X,174-Y)=1 OR POINT (X,176-Y)=1
) AND (POINT (X+1,175-Y)=1 OR POINT (X-1,175-Y)=1) THEN DRAW G/2,
G
160 PLOT X*G,150-Y*G: IF POINT (X+1,174-Y)=1 AND POINT (X+1,175-
Y)=0 AND POINT (X,174-Y)=0 THEN DRAW G,-G: PLOT X*G+G/2,150-Y*G+G
: DRAW G,-G: GO SUB 290
170 PLOT X*G,150-Y*G: IF POINT (X+1,176-Y)=1 AND POINT (X+1,175-
Y)=0 AND POINT (X,176-Y)=0 THEN DRAW G,G: PLOT X*G+G/2,150-Y*G+G:
DRAW G,G: GO SUB 260
180 PLOT X*G,150-Y*G: IF POINT (X-1,175-Y)=0 AND POINT (X,176-Y)
=0 AND POINT (X+1,175-Y)=0 AND POINT (X,174-Y)=0 AND POINT (X-1,1
76-Y)=0 AND POINT (X+1,176-Y)=0 AND POINT (X+1,174-Y)=0 AND POINT
(X-1,174-Y)=0 THEN DRAW 0,G: DRAW G/2,G: DRAW 0,-G: DRAW -G/2,-G
190 RETURN
200 IF POINT (X+2,175-Y)=0 THEN DRAW -G/2,-G
210 IF POINT (X-1,175-Y)=0 THEN PLOT X*G,150-Y*G: DRAW G/2,G
220 RETURN
230 IF POINT (X,173-Y)=0 THEN DRAW -G/2,-G
240 IF POINT (X,176-Y)=0 THEN PLOT X*G,150-Y*G: DRAW G/2,G
250 RETURN
260 IF POINT (X+2,177-Y)=0 THEN DRAW -G/2,-G
270 IF POINT (X-1,176-Y)=0 AND POINT (X-1,174-Y)=0 THEN PLOT X*G
,150-Y*G: DRAW G/2,G
280 RETURN
290 IF POINT (X+2,173-Y)=0 THEN DRAW -G/2,-G
300 IF POINT (X-1,176-Y)=0 THEN PLOT X*G,150-Y*G: DRAW G/2,G
310 RETURN
9000 RESTORE 9010: FOR L=0 TO 106: READ R: POKE 32000+L,R: NEXT L
: RETURN
9010 DATA 33,24,125,6,32,126,245,62,2,229,213,197,205,1,22,193,20
9,225,241,215,35,16,-18,201
9020 DATA 127,32,49,57,56,56,32,66,87,90,32,83,79,70,84,87,65,82,
69,32,80,85,66,76,73,67,65,84,73,79,78,83
9030 DATA 33,0,88,22,7,62,0,6,3,14,32,119,13,35,32,-5,197,14,50,6
,0,5,32,-3,13,32,-8,193,198,9,5,32,-24,21,32,-31,201
9040 DATA 33,0,64,17,1,64,1,255,23,54,0,237,176,201

```

JAAROVERZICHT BULLETIN 5^E JAARGANG

Hieronder volgt zoals, altijd in het juni-nummer, het overzicht van wat er het afgelopen jaar in ons bulletin gestaan heeft. Als er een artikel is wat je graag zou willen hebben, neem dan even contact op met de redactie. Wij zorgen dan dat je het gewenste artikel (tegen onkosten) in huis krijgt.

Flora en Rudy

* SPECTRUM

Spectrum +3	1 6
Woordzoeker	1 8
Handige tip Microdrive	1 16
Hoeveel K over	1 17
Instant screen	3 5
Geen BREAK	3 6
Tips voor de Spectrum	3 9
Nieuws over +3	3 10
Kalender	4 7
Tekening op het scherm	4 10
Geluidssensor	4 14
Afronden van getallen	5 4
Geluid voor de Spectrum	5 6
Spectrum repareren?	6 6
Hoe start Spectrum op?	6 12
TW3 printer aanpassing	7 8
Super Spectrum op komst	8 11
BASIC compilers	8 14
Masterfileprint	9 5
Voorbeelden en tips	9 6
MFprint	9 18
Ruitjes	10 6
64 kleuren	10 12
3D letters	10 12

* SERIES

Snel & precies (8)	1 10
Onze Spectrum (5)	2 7
Snel & precies (9)	2 13
Onze Spectrum (6)	3 7
Snel & precies (10)	3 11
Snel & precies (11)	4 17
Snel & precies (12)	5 12
Snel & precies (13)	6 14
Beginners BASIC (1)	7 15
Beginners BASIC (2)	8 7
Snel & precies (14)	8 16
Beginners BASIC (3)	9 8
Snel & precies (15)	9 14

* PRINTERS

TW3 printer aanpassing	7 8
Printoptie Spectrum 128K	7 11
Masterfileprint	9 5
Voorbeelden en tips	9 6
MFprint	9 6
Pseudo NLQ voor TW3 en	9 12
Smith Corona Fastext 80	10 24
De printer	10 24

* ZX81

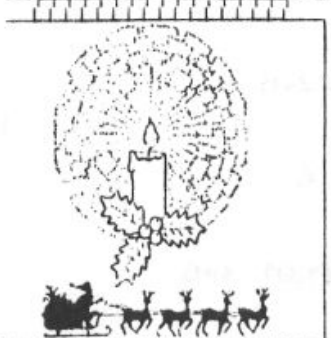
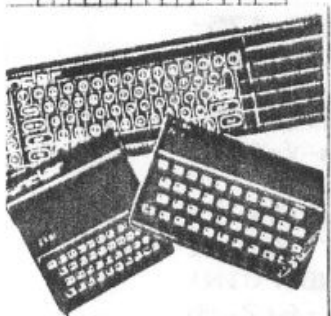
Coral BASIC voor ZX81	4 6
-----------------------	-----

* HARDWARE/TECHNIEK

Temperatuur meten	2 11
Ear indicator	4 15
Geheugen uitbreiding	5 7
naar 256K RAM en 96K ROM	5 7
Spectrum repareren	6 6
Vraag Lautenbach	6 9
Hoe start de Spectrum op	6 12
Digitale voltmeter	9 13

* 128K ONDERWERPEN

Muziek 128	4 9
Sinclairgids cassette	5 6
voor 128K gebruikers	5 6
Muziek voor de 128K	5 6
Spectaculair	5 6
Vader Jacob 128	6 4
Printoptie Spectrum 128	7 11
Jukebox voor de 128	8 19
128 Music	10 4
120 kleuren	10 6



* ALGEMEEN

* EVENEMENTEN

Redactie zoekt redactie	1	4
Eindelijk de plus 3	1	6
Handige tip Microdrive	1	16
Opus ROM	1	18
Shaduw ROM	1	19
Instructeurs gevraagd	2	6
Ben ik een komputernul?	2	9
Geen bericht, slecht bericht	2	11
5 jaar bulletin	2	17
Hobbyscoop nieuws	2	18
Nieuws over de +3	3	10
Kabel monitor<>128	4	3
Ledenvergadering	4	4
Contributie 1988	4	8
Prettige feestdagen	4	12
Handig stofzuigertje	4	15
Idee vraag en antwoord rubriek	4	16
Boeken bij de Slegte	4	23
Wat is BASICode	6	5
Vragen Arie Lautenbach	6	9
Brief van Arie Lautenbach	6	11
Griepvirus...komputer-virus	7	5
Verhuur modem	7	6
Bedankje	7	7
Spoken d.m.v. de komputer	7	12
Het ikonenbord	7	13
Iets over talen	7	14
Databanken e.d.	8	5
Voorbeeld pagina's databanken	8	6
Super Spectrum op komst	8	11
Ruilabonnementen erbij	8	13
Aanvulling spectaculair	8	13
Meer voorbeeld pagina's databanken	8	23
Masterfileprint	9	5
Voorbeelden en tips MFprint	9	6
Ledenvergadering	9	7
Talen	10	7
Databankje	10	8
Jaaroverzicht	10	14
De printer	10	24

Infodag SGGE	3	18
Hard- en softwaremarkt	3	19
Ledenvergadering '87	4	4
Verslag radiomarkt Assen	4	4
Demonstratie 256K	5	18
N.A.T. 1988	6	9
Verslag N.A.T. '88	7	7
Ledenvergadering '88	9	7

* HOBBYSCOOP/BASICODE

Hobbyscoop nieuws	2	18
Wat is BASICode	6	5

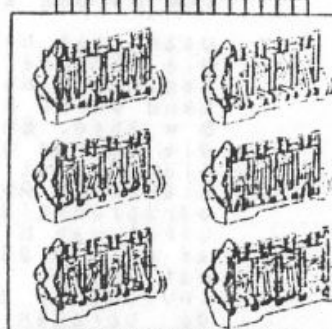
* REKTIFIKATIES

Snel & precies (9)	3	18
Snel & precies (10)	4	17
Snel & precies (11)	5	16
Telefoon Arie Lautenbach	8	13
Snel & precies (15)	10	24

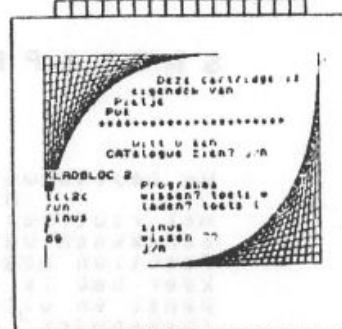
BULLETIN SINCLAIR GEBRUIKERSGROEP GRONINGEN/ASSEN



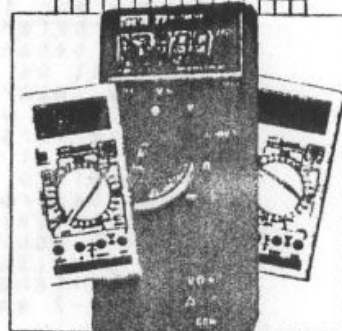
5e jaargang nummer 6 februari '88



5e jaargang nummer 7 maart '88



5e jaargang nummer 8 april '88



5e jaargang nummer 9 mei '88

BULLETIN SINCLAIR GEBRUIKERSGROEP GRONINGEN/ASSEN



5e jaargang nummer 10 juni '88

Een ongeluk komt zelden alleen is een "levenswet". Vrij vetaald: als er iets fout gaat beperkt zich dat ook vaak niet tot een enkele. De Redactie had wegens ruimtedebrek in het vorige nummer de LABELS van figuur 4B weggelaten. Dat maakte deze toch al niet gemakkelijke routine nog moeilijker te volgen. Anderzijds was ik bezig -zoals aangekondigd- een deel van de routines te gebruiken voor "REKENEN" deel 16, waarin het precies berekenen van zeer grote vermenigvuldigingen wordt behandeld. En toen ontdekte ik een foutje en later nog een. Beide hebben GEEN invloed op de gepubliceerde resultaten, maar dat had anders kunnen zijn. In overleg met de Redactie volgt daarom hieronder nogmaals het gehele artikel. Een aantal figuren en de tekst moesten namelijk worden aangepast.

Waar gaat het om? Om MDFD-CTR3, die afgeleid wordt van CTR3, beginnend op AD 40DB. Zoals bekend werkt B = 00 op DJNZ als B = 256d. En dus moet INC C voor die waarde van B = L achterwege blijven. Dat kost 4 bytes -zie hieronder AD 40DD/E0- maar in de oorspronkelijke, foute routine wordt van N = 284 af MDFD-CTR3 af en toe 256 te groot!! Dat heeft een verwaarloosbare invloed op de rekensnelheid. Bij de berekening van bijvoorbeeld 1000! treedt die fout 4-maal op. Dat kost $13\frac{1}{2} \times 256 \times 50$ T-cycles of te wel 0.05 s extra per keer, bepaald niet iets om over wakker

te liggen. Maar het werkgebied wordt door die fout met 100 hex bytes vergroot. Bij de uitvoering van LOOP3 vindt die uitbreiding "naar beneden" plaats dus op de adressen lager dan LO-(DE) en LO-(HL). Bij de uitvoering van SHIFT-L-4BITS is de gebiedsuitbreiding "naar boven", op de adressen hoger dan HI-(DE) dus. In beide gevallen KAN van buiten het geCLEARde gebied vervuiling van het werkgebied optreden en dat leidt vanzelfsprekend tot foute resultaten. Vandaar de noodzaak van de correctie.

Bij het onderzoek van de genoemde fout heb ik me ook gerealiseerd dat de adressen 7000 en 7001 tot de 'active area' van DE-STORE konden behoren, omdat CTR3 voor 2 'empty bytes' aan de bovenkant van dat gebied zorgde. En ook dat kan vervuiling veroorzaken. In de verbeterde routine is dat gebied geleegd van LO-(DE) - 2 t/m hoogstens 6FFF. Zie daartoe de -geen extra bytes kostende- verandering op AD 40D4/DA.

De verbetering van de foutjes in NFF-MC kan ook via vergroting van het CLEAR-gebied met 2-maal 100 hex bytes worden bereikt, maar dat zou meer dan 4 bytes extra kosten. Als er van wordt uitgegaan dat de betreffende RAM-gebieden altijd "schoon" beginnen na het LOADen, hadden de verbeteringen zelfs achterwege kunnen blijven. Fundamenteel is de gekozen weg echter de enige juiste.

Han van Abbe

SNEL & PRECIES REKENEN MET DE ZX 81 (15)

We beschouwen nu het programma NFF - MC met routines die het supersnel berekenen van $N! = N \cdot \text{Factorial}$ Function mogelijk maken. Vorige keer heb ik gezegd dat het berekenen en uitprinten van 1000!, 'benchmark' voor de rekensnelheid, met dit programma slechts 2 min 28 s kost, terwijl NFF-6 er 3 uur 47 min voor nodig had. In dat vorige artikel stelde ik ook dat herbeschouwing van een programma een tijdje later, weer tot een verbetering kan leiden. En daarom behandel ik in dit artikel niet de aangekondigde vierde versie, maar een nog recentere vijfde, die weer 9% sneller is. Daarmee verschijnt 1000! na 2 min 15 s op het scherm, wat dus ruim 100-maal zo snel is als met het reeds besproken BASIC-programma. Alhoewel beide programma's qua functie equivalent zijn, is de hier gevolgde methodiek geheel verschillend van de systematiek van NFF-7 en NFF-6.

METHODIEK

Behalve voor de conversie van het 4 hexdigits grote getal N-1 in een 4 decimale digits groot getal wordt geen gebruik gemaakt van de CALCULATOR. En wel om de meermaals genoemde reden: het bereiken van de grootst mogelijke rekensnelheid. Aan dit doel -vooral essentieel bij grotere waarden van N- wordt ook verder alles ondergeschikt gemaakt. Eventueel ten koste van de -toch zeer geringe- rekentijd voor kleine waarden van N.

Het vermenigvuldigen van een getal G met een integer N is qua principe in MC zeer simpel: tel N-1 keer G bij zichzelf op. Daarbij wordt gebruik gemaakt van 2 opslaggebieden, stores. G staat in (DE)-store, pointer DE en daarmee begint ook (HL)-store, pointer HL. Door N-1 keer (DE) bij (HL) op te tellen staat in die laatste store uiteindelijk $N \cdot G$. Dat optellen gebeurt decimaal met carry, instructies ADC/DAA, byte na byte beginnend bij de byte met het hoogste adres en vervolgens steeds een adres lager.

In (DE) en (HL) staan daardoor decimale getallen op de normale wijze: laagste adres 'most significant digits', hoogste adres 'least significant digits', 2 digits per byte.

In de praktische uitvoering wordt van de stores alleen het "nuttige" gebied betrokken bij de berekeningen. Deze beperking van het werkgebied is vanzelfsprekend weer om rekentijd te besparen. Daartoe zoeken we na iedere vermenigvuldiging het hoogste adres op in (HL)-store, met een byte waarvan de waarde niet-nul is: HI-(HL). En bepalen daarna het adres in (DE)-store dat hiermee correspondeert: HI-(DE). In (DE)-store zoeken we vervolgens het laagste adres op met een byte <> 00: LO-(DE). Het deel van (DE) van deze laatste PTR verminderd met 2, t/m de PTR HI-(DE) bepaalt de grootte van het nuttige werkgebied in de beide stores. Gewerkt wordt van hoog naar laag, waardoor aan het begin van iedere vermenigvuldiging twee lege bytes "aan de onderkant" van de gebieden ter beschikking staan, nodig en voldoende voor het vermenigvuldigen van G met een N van maximaal 9999.

Het vermenigvuldigen gebeurt het snelst - behalve voor zeer lage waarden van N - als we dat met ieder van de vier digits van de multiplier CTR separaat doen. Dan behoeft immers de betreffende optelling maximaal 9 keer te worden uitgevoerd. Hiertoe gebruiken we de (HL)-store ook van het adres HI-(HL) minus 1 af en daarna de lagere adressen. Dit geven we aan als (HL)), met twee sluithaakjes dus. En (DE) gebruiken we ook in de vorm "shifted left 4 BITS" wat we met (DE)) aanduiden. Het zal duidelijk zijn dat optellen van (DE) bij (HL)) overeenkomt met het vermeerderen van (HL) met 100*G. En zo betekent (DE)) optellen bij (HL) het toevoegen van 10*G en tenslotte (DE)) bij (HL)) is het bütellen van 1000*G bij (HL). We doen nu achtereenvolgens:

- . LOLO(=d4)-maal (DE) bij (HL)
- . LOHI(=d2)-maal (DE) bij (HL)
- . HILO(=d3)-maal (DE) bij (HL)
- . HIHI(=d1)-maal (DE) bij (HL)

waardoor het gewenste resultaat wordt verkregen: (HL) is 1000*d1 + 100*d2 + 10*d3 + d4 keer bij zichzelf opgeteld.

De beginwaarde van G is 01 en van CTR N-1. Nadat de vermenigvuldiging met N is uitgevoerd wordt (HL) naar (DE) overgebracht: TRFR. Achtereenvolgens wordt dan het zojuist beschreven proces herhaald met CTR = N-2, N-3, ..., 3, 2, 1 waardoor N! in decimale vorm ontstaat in (HL). Het uitvoeren van de berekening beginnend met CTR = N-1 en niet met 1 is weer een kwestie van keuze voor de snelst mogelijke routine. Nu worden namelijk de kleinste waarden van G vermenigvuldigd met de grootste waarden van de vermenigvuldiger.

FIG. 3 BASIC PROGRAMMA "NFF-MC"

```
0 REM-REGEL MET 303 BYTES MC
2 INPUT N
3 RAND N
4 PRINT N;" FACTORIAL=";USR 1
514;" DIGITS*1E";USR 16764
6 RAND USR 16769
```

FIG. 3A HEXDUMP "NFF-MC"

4082--ED4B	3240	0BCD	2015--287
408A--EFA4	2EA4	2EA4	2E30--395
4092--3500	C004	0FE0	040F--1FB
409A--E004	0F34	CD8A	15ED--380
40A2--4335	4001	0020	21FF--1FA
40AA--4F11	0050	EDB0	1B25--28E
40B2--5F34	0C18	032A	7740--411
40BA--E5ED	B8E1	232B	7EA7--1246
40C2--28FB	2277	407C	EE30--918
40CA--6722	7940	E513	1AA7--753
40D2--28FB	1B1B	23ED	52E5--928
40DA--D545	4C78	A028	010C--691
40E2--ED43	7B40	3A36	40CD--872
40EA--5541	35C0	5541	34E1--835
40F2--ED4B	7B40	AFED	6F2B--1065
40FA--8677	2323	10F6	0D20--630
4102--F33A	3640	CD51	4135--823
410A--CD51	4134	2136	407E--680
4112--3D27	7730	06AF	237E--609
411A--3D27	77C1	D130	9611--836
4122--0050	131A	A728	FBED--826
412A--5379	402A	7740	E5ED--959
4132--5223	29FE	1030	04C6--678
413A--A012	2B44	4DD1	21FF--863
4142--5FED	5229	1AE6	0F20--758
414A--020B	2322	7B40	C91F--501
4152--1F1F	1FE6	0F28	1C2A--448
415A--7740	ED5B	7940	ED4B--1008
4162--7B40	08AF	1A8E	2777--696
416A--2B1B	10F8	0D20	F508--632
4172--3D20	E421	7740	3A37--650
417A--40C9	ED4B	7B40	C92A--1007
4182--7940	ED5B	7740	7EE6--1052
418A--F0D6	A028	097E	E6F0--4EB
4192--1F1F	1F1F	C61C	D77E--2B3
419A--E60F	C61C	D7ED	5219--405
41A2--2322	7940	38E7	D61C--30F
41AA--2004	FD35	0ED7	C9 --304

16514-ED4B	3240	0BCD	2015- 695
16522-EFA4	2EA4	2EA4	2E30- 917
16530-3500	C004	0FE0	040F- 507
16538-E004	0F34	CD8A	15ED- 896
16546-4335	4001	0020	21FF- 506
16554-4F11	0050	EDB0	1B26- 654
16562-5F34	0C18	032A	7740- 411
16570-E5ED	B8E1	232B	7EA7-1246
16578-28FB	2277	407C	EE30- 918
16586-6722	7940	E513	1AA7- 753
16594-28FB	1B1B	23ED	52E5- 928
16602-D545	4C78	A028	010C- 691
16610-ED43	7B40	3A36	40CD- 872
16618-5541	35C0	5541	34E1- 835
16626-ED4B	7B40	AFED	6F2B-1065
16634-8677	2323	10F6	0D20- 630
16642-F33A	3640	CD51	4135- 823
16650-CD51	4134	2136	407E- 680
16658-3D27	7730	06AF	237E- 609
16666-3D27	77C1	D130	9611- 836
16674-0050	131A	A728	FBED- 826
16682-5379	402A	7740	E5ED- 959
16690-5223	29FE	1030	04C6- 678
16698-A012	2B44	4DD1	21FF- 863
16706-5FED	5229	1AE6	0F20- 758
16714-020B	2322	7B40	C91F- 501
16722-1F1F	1FE6	0F28	1C2A- 448
16730-7740	ED5B	7940	ED4B-1008
16738-7B40	08AF	1A8E	2777- 696
16746-2B1B	10F8	0D20	F508- 632
16754-3D20	E421	7740	3A37- 650
16762-40C9	ED4B	7B40	C92A-1007
16770-7940	ED5B	7740	7EE6-1052

16778-F0D6 A028 097E E6F0-1259
16786-1F1F 1F1F C61C D77E- 691
16794-E60F C61C D7ED 5219-1030
16802-2322 7940 38E7 D61C- 783
16810-2004 FD35 0ED7 C9 - 772

ROUTINES

In figuur 3 staat het BASIC-PG en in figuur 3A de HEXDUMP van regel 0. In figuur 4A worden de in NFF-MC gebruikte stores, counters en pointers gegeven van zowel de rekenroutines als de print routines. De routines zelf staan in figuur 4B. In het eerste deel van de routine wordt N-1, de beginwaarde van CTR, als 4 digits groot decimaal getal opgeslagen op AD(res) 4036/37. Via RAND N, regel 3, is N in hex vorm beland in SEED, AD 4032/33. Na vermindering met 1 wordt die waarde in de F(floating) P(oint) vorm op de Calculator Stack(STK) gezet. In de FPA(rithmetic)-RTN gebeurt nu het volgende:

. R4 brengt 10(decimaal) op de STK en na uitvoering van de instructie 2E(n-mod-m) met deze m(modulus) staat op de STK R(est)1 en INT((N-1)/10); R1=d4, de 4de en laagste digit.
. deze bewerking wordt 3 keer toegepast zodat dan op de STK staan R1=d4, R2=d3, R3=d2 en INT((N-1)/1000)=d1.
. vervolgens wordt 16d op de STK gebracht met 30 35 00(stk-16) en tevens opgeslagen in MEM-0 met de instructie C0(stk-mem-0).
. driemaal wordt met 16d vermenigvuldigd (04=multiplied) en opgeteld (0F=addition); E0 (get-mem-0) haalt 2 keer 16d weer op de STK.
. na de laatste bewerking staat op de STK 16*16*16*d1 + 16*16*d2 + 16*d3 + d4 en dat getal wordt via FP-TO-BC overgebracht naar het zoeven genoemde AD.
. de opslag in 4036-37 van dit d e c i m a l e getal met digits d1 d2 d3 d4 is in de volgorde HILO(d3), LOLO(d4), HIHI(d1), LOHI(d2).

In het tweede deel van de RTN maakt CLEAR de 'stores' schoon. INIT brengt de PTRs DE en HL op de initiële waarde van HI-(DE) en HI-(HL), SET (HL) op 01 en springt met BC = 0001 naar START van COMPUTE FOR CTR-VALUES, het derde deel. De volgende stap is (HL) naar (DE) over te brengen (TRFR). De byte op AD HI-(DE) heeft daardoor initieel ook de waarde 01. De eerste SEEK-loop brengt de PTRs (4077) en (4079) op de juiste waarde. De truo met XOR 30 (AD 40C8) verandert de 1ste nibble van H van een 5 in een 6. Natuurlijk doet ADD A, 10 hetzelfde, XOR 30 doet eventueel ook het omgekeerde. De tweede SEEK-loop begint met de waarde van DE resulterend na TRFR, dus altijd met een lege byte aan de onderkant van het (DE)-gebied en zoekt LO-(DE) op. Daarna wordt CTR3 bepaald, 2 groter dan het "gevulde" gebied. Zoals onder METHODIEK verklaard hebben we dan het gewenste werkgebied voor iedere vermenigvuldiging.

Vervolgens wordt CTR3 gemodificeerd: B neemt de waarde van L over en C die van H vermeerderd met 1. Tenzij B = 00 is want dat werkt op DJNZ als B = 256d, dus moet INC C achterwege blijven: AD 40DD/E1. BC en (407B) bevatten de MDFO-CTR3. Waarom deze schijnbaar vreemde omvorming? Het antwoord is weer: optimaliseren van de rekensnelheid en wel in de nog te bespreken LOOP3 en LOOP4. Vooral LOOP3 is uitermate belangrijk, omdat het aantal malen dat deze doorlopen wordt zeer groot is: bijna 8 miljoen keer bij 1000!. Door de modificatie kan DJNZ worden gebruikt, gekombineerd met DEC C/JRNZ resulterend in een LOOP3 van slechts een fractie meer dan 50 T-cycles. Na een "ronde" DJNZ is B = 00 = 256d, tenzij DEC C een Z oplevert. Omdat we C (behalve voor B = 00) eerst met 01 hebben verhoogd werkt JRNZ nu quasi als JRNC voor de oorspronkelijke waarde van C.

Het tweede gedeelte van COMPUTE FOR CTR-VAL voert met de subroutine COMPUTE PER DIGIT de eigenlijke berekening uit konform wat hierover eerder is verklaard. DEC (HL) en INC (HL) zorgen voor het op het juiste moment gebruiken van (HL) en (HL). Deze instructies werken alleen op de LSB van (4077), de PTR voor HI-(HL). En beperken daarmee de hoogste waarde van N tot die, waarbij HI-(HL) eindigt op AD 5F00 dus HI-(HL) op AD 5F01. Het maximale aantal nullen dat hiermee correspondeert is 509.

De overgang van (DE) naar (DE) gebeurt met SHIFT-L-4BITS toegepast op de bytes van (DE)-store door de LOOP4-routine. Initieel is A = 00 en HL = LO-(DE). Met in (DE) van HL-1 af: 00ab0d0lef...pql00 wordt (HL) = ab. De instructie RLD maakt hiervan A = 0a, (HL) = 00. DEC HL/ADD A, (HL)/LD (HL), A verandert (DE) in 0a0b0d0lef... Na INC HL/INC HL voert de volgende ronde tot (DE) = 0a0b0d0lef... Na nog 'n ronde is (DE) = 0a0b0d0lef... De laatste ronde doet (DE) eindigen met ...pql0, waarmee het gewenste resultaat is bereikt.

DECREMENT-CTR vermindert CTR, opgeslagen op 4036/37 decimaal met 1, waarna weer wordt begonnen met LOOP1 met de juiste waarden voor TRFR en de overige delen van deze RTN. Een en ander totdat in de laatste LOOP1-ronde CTR = 0 geen invloed meer heeft op het resultaat. Daarna wordt CTR = -1 (=9999) en de carryflag geSET, waardoor het rekenen stopt. LET OP: DEC A/DAA werkt alleen goed als uitgevoerd met No Carry; daarvoor zorgen de instructies op AD 4155 en 4117.

De subroutine COMPUTE PER DIGIT OF CTR-VALUE gebruikt A = CTR2 voor LOOP2 en A' voor de optelling in LOOP3.

BULLETIN SIGG

zoeven besproken Samen met de wordt gezorgd voor de juiste kombinaties van de digits van CTR met de gebieden (HL) of (HL) en (DE) of (DE)). Na de beëindiging van de berekening wordt in PREPARE PRINTOUT het laagste adres van (HL) opgezocht met een byte < > 00: LO-(HL). Dit is de 1ste byte voor de PRINTOUT 'FIRST' met PTR 4079. HI-(HL) = 'LAST' is al op AD 4077 vastgelegd. Vervolgens wordt het aantal te printen digits bepaald, rekening houdend met een eventuele 'leading zero' in LO-(HL).

Deze nul wordt veranderd in een "A", om later herkend te worden. Dan wordt het aantal 'trailing' nullen bepaald, zijnde de machten van 10 van het resultaat. Die worden niet geprint, maar via regel 4 en de subroutine op AD 417C/80 gemeld. Ook hier wordt weer rekening gehouden met een eventuele laatste 0 van HI-(HL), die het aantal met 1 verhoogd en het aantal te printen digits met 1 verlaagd. De twee berekende aantallen verschijnen op het scherm door de opdrachten in regel 4: PRINT.....;USR 16514; "DIGITS*1E";USR 16764.

FIG. 4A STORES, COUNTERS (CTR), POINTERS (PTR) OF MC-RTN "NFF-MC"

ADDRESS	USAGE IN COMPUTE-RTNS
(5000) ... 5FFF	(HL) -STORE
(5000) ... 6FFF	(DE) -STORE
4032	SEED = HEX N THROUGH RAND N
4036	(MAIN) CTR = MULTIPLIER VALUES: N-1, N-2, ..., 3, 2, 1
----	CTR2: FOR EACH ROUND 1 OF THE 4 DIGITS OF CTR
4077	HI-(HL): PTR TO HIGHEST ADDRESS IN (HL) -STORE WITH BYTE < > 00
4079	HI-(DE): PTR TO HIGHEST ADDRESS IN (DE) -STORE WITH BYTE < > 00
----	LO-(DE): PTR TO LOWEST ADDRESS IN (DE) -STORE WITH BYTE < > 00
----	CTR3: NUMBER OF BYTES USED IN STORES AS WORKING AREA
407B	MDFD-CTR3: MODIFIED CTR3 FOR FASTEST COMPUTATION
----	CTR4: MDFD-CTR3 IS ALSO CTR OF LOOP4, SHIFT LEFT 4 BITS

ADDRESS	POINTERS PRINTOUT-RTNS
407B	NR OF TRAILING ZERO'S
4079	FIRST BYTE PRINTOUT/ STORE-POINTER
4077	LAST BYTE PRINTOUT

FIG. 4B MC-ROUTINE "NFF-MC"

SET CTR AT N-1: 4 DIGITS DECIMAL

16514	4032--ED4B3240	LD	BC, (4032)	SEED, N REX
	4036--0B	DEC	BC	N-1 ALX
	4087--CD2015	CALL	1520	STK BC
	408A--EF	RST	28; FPA:	
	408B--A4/2E/A4/2E/A4/2E/30/35			see text
	4093--00/C0/04/0F/E0/04/0F/E0/			
	409B--04/0F/34			
	409E--CD8A15	CALL	158A	FP-T0-BC
	40A1--ED433640	LD	(4036), BC	CTR, N-1 dec

BULLETIN SGG

CLEAR + INITIALIZE

CLEAR	40A5--010020	LD	BC, 2000	
	40A3--21FF4F	LD	HL, 4FFF	(HL) = 00
	40AB--110050	LD	DE, 5000	
	40AE--EDB0	LDIR		
INIT	40B0--1B	DEC	DE	DE = 6FFF
	40B1--265F	LD	H, 5F	HL = 5FFF
	40B3--34	INC	(HL)	SET (5FFF) at 01
	40B4--0C	INC	C	BC = 0001
	40B5--1803	JR	40BA	START

COMPUTE FOR CTR-VALUES N-1 ... 1

LOOP/	40B7--2A7740	LD	HL, (4077)	HI = (HL)
START	40BA--E5	PUSH	HL	stack
TRFR	40BB--EDB8	LDOR		DE < LO-(DE)
	40BD--E1	POP	HL	retrieve HI-(HL)
	40BE--23	INC	HL	
SEEK (HL) <> 00	40BF--2B	DEC	HL	
	40C0--7E	LD	A, (HL)	
	40C1--A7	AND	A	
	40C2--28FB	JR	Z, 40BF	SEEK (HL) <> 00
	40C4--227740	LD	(4077), HL	NEXT HI-(HL)
	40C7--7C	LD	A, H	Right nibble of
	40C8--EE30	XOR	30	H from
	40CA--67	LD	H, A	5 to 6
	40CB--227940	LD	(4079), HL	NEXT HI-(DE)
	40CE--E5	PUSH	HL	stack
SEEK (DE) <> 00	40CF--13	INC	DE	DE <= LO-(DE)
	40D0--1A	LD	A, (DE)	
	40D1--A7	AND	A	
	40D2--28FB	JR	Z, 40CF	SEEK (DE) <> 00
	40D4--1B	DEC	DE	
	40D5--1B	DEC	DE	DE = LO-(DE) - 2
	40D6--23	INC	HL	HL = HI-(HL) + 1
	40D7--ED52	SBC	HL, DE	HL = CTR3
	40D9--E5	PUSH	HL	stack CTR3
	40DA--D5	PUSH	DE	stack LO-(DE) - 2
	40DB--45	LD	B, L	B = LO - CTR3
	40DC--4C	LD	C, H	
	40DD--78	LD	A, B	
	40DE--A0	AND	B	is LO-CTR3 = 00?
	40DF--2801	JR	Z, 40E2	NO-INC if yes
NO-INC	40E1--0C	INC	C	C = H + 1 = HI-
	40E2--ED437B40	LD	(407B), BC	MOD - CTR3
RIGHT-DIGS	40E8--3A3840	LD	A, (4036)	LO-CTR
	40E9--CD5541	CALL	4155	R-DIG: 1 * LCLO
	40EC--35	DEC	(HL)	→ (HL)
	40ED--CD5541	CALL	4155	R-DIG: 100 * LONI
	40F0--34	INC	(HL)	→ (HL) normal
SHIFT-L-4BITS	40F1--E1	POP	HL	retrieve LO-(DE) - 2
	40F2--ED4B7B40	LD	BC, (407B)	MOD - CTR3
LOOP4	40F6--AF	XOR	A	
	40F7--ED6F	RLD		(DE)
	40F9--2B	DEC	HL	
	40FA--86	ADD	A, (HL)	shifted to
	40FB--77	LD	(HL), A	(DE))
	40FC--23	INC	HL	
	40FD--23	INC	HL	
	40FE--10F6	DJNZ	40F6	LOOP4
	4100--00	DEC	C	HI-MOD-CTR3
LEFT-DIGS	4101--20F3	JR	NZ, 40F6	LOOP4
	4103--3A3840	LD	A, (4036)	LO-CTR
	4106--CD5141	CALL	4151	L-DIG: 10 * H1LO
	4109--35	DEC	(HL)	→ (HL)
	410A--CD5141	CALL	4151	L-DIG: 100 * H1HI
	410D--34	INC	(HL)	→ (HL) normal
DECREMENT-CTR	410E--213640	LD	HL, 4036	CTR
	4111--7E	LD	A, (HL)	A = LO-CTR
	4112--3D	DEC	A	
	4113--27	DAA		
	4114--77	LD	(HL), A	NEXT LO-CTR
DEC-HI	4115--3006	JR	NC, 411D	NO-DEC-HI
	4117--AF	XOR	A	→ No Carry
	4118--23	INC	HL	
	4119--7E	LD	A, (HL)	A = HI-CTR
	411A--3D	DEC	A	
	411B--27	DAA		
	411C--77	LD	(HL), A	NEXT HI-CTR
NO-DEC-HI	411D--C1	POP	BC	retrieve CTR3
	411E--D1	POP	DE	retrieve HI-(DE)
	411F--3096	JR	NC, 40B7	LOOP1

BULLETIN SIGG

PREPARE PRINTOUT

16673
SEEK LO-(HL)

4121--110050	LD	DE, 5000	begin of (HL)
4124--13	INC	DE	
4125--1A	LD	A, (DE)	
4126--A7	AND	A	
4127--28FB	JR	Z, 4124	SEEK LO-(HL)
4129--ED537940	LD	(4079), DE	FIRST
412D--2A7740	LD	HL, (4077)	LAST: HI-(HL)
4130--E5	PUSH	HL	Stack
4131--ED52	SBC	HL, DE	LAST minus FIRST
4133--23	INC	HL	+1
4134--29	ADD	HL, HL	* 2
4135--FE10	CP	10	>= 10 at LO-(HL)?
4137--3004	JR	NC, 413D	NO leading zero if so
4139--C6A0	ADD	A, A0	SET first
413B--12	LD	(DE), A	digit at A
413C--2B	DEC	HL	-1
413D--44	LD	B, H	BC is number of
413E--4D	LD	C, L	printed digits
413F--D1	POP	DE	retrieve HI-(HL)
4140--21FF5F	LD	HL, 5FFF	end of (HL)
4143--ED52	SBC	HL, DE	HL = trailing 0
4145--29	ADD	HL, HL	* 2
4146--1A	LD	A, (DE)	DE = HI-(HL)
4147--E60F	AND	0F	trailing zero?
4149--2002	JR	NZ, 414D	NOT if not
414B--0B	DEC	BC	-1 printed
414C--23	INC	HL	+1 trailing
414D--227B40	LD	(407B), HL	trailing 0
4150--C9	RET		to BASIC with no of printed digits

NO

NOT

COMPUTE PER DIGIT OF CTR-VALUE

LEFT-DIGIT

4151--1F	RRR		
4152--1F	RRR		
4153--1F	RRR		
4154--1F	RRR		
RIGHT-DIGIT			
4155--E60F	AND	0F	A = CTR2
4157--281C	JR	Z, 4175	NEXT-DIGIT
LOOP2			
4159--2A7740	LD	HL, (4077)	HI-(HL)
415C--ED5B7940	LD	DE, (4079)	HI-(DE)
4160--ED4B7B40	LD	BC, (407B)	NOFD - CTR3
4164--08	EX	AF, AF	
LOOP3			
4165--AF	XOR	A	→ No Carry
4166--1A	LD	A, (DE)	
4167--8E	ADC	A, (HL)	ADC (DE)
4168--27	DAA		decimally to
4169--77	LD	(HL), A	(HL)
416A--2B	DEC	HL	next lower
416B--1B	DEC	DE	bytes
416C--10F8	DJNZ	416B	LOOP3
416E--0D	DEC	C	HI - NOFD - CTR3
416F--20F5	JR	NZ, 416B	LOOP3
4171--08	EX	AF, AF	
4172--3D	DEC	A	NEXT CTR2
4173--20E4	JR	NZ, 4159	LOOP2
NEXT-DIGIT			
4175--217740	LD	HL, 4077	AD PTR HI-(HL)
4178--3A3740	LD	A, (4037)	HI-CTR
417B--C9	RET		

NUMBER OF TRAILING ZERO'S

16764

417C--ED4B7B40	LD	BC, (407B)	no of trailing 0
4180--C9	RET		

PRINT DIGS (<) LEADING/TRAILING 0

16769

4181--2A7940	LD	HL, (4079)	FIRST/STORE-PTR
4184--ED5B7740	LD	DE, (4077)	LAST
4188--7E	LD	A, (HL)	
4189--E6F0	AND	F0	1st digit
418B--D6A0	SUB	A0	SET A?
418D--2809	JR	Z, 419B	PRINT-SPACE if yes
418F--7E	LD	A, (HL)	
4190--E6F0	AND	F0	Left digit
4192--1F	RRR		
4193--1F	RRR		
4194--1F	RRR		
4195--1F	RRR		
4196--C61C	ADD	A, 1C	Right
4198--D7	RST	10	make CTR-CODE
4199--7E	LD	A, (HL)	PRINT L-dig

NXT-2DIGS

PRINT-SPACE

BULLETIN SGG

419A--E60F	AND	OF	<i>Right digit</i>
419C--C61C	ADD	A,1C	<i>make CHR-CODE</i>
419E--D7	RST	10	<i>PRINT R-dig</i>
419F--ED52	SBC	HL,DE	<i>check for</i>
41A1--19	ADD	HL,DE	<i>LAST byte</i>
41A2--23	INC	HL	
41A3--227940	LD	(4079),HL	<i>STORE-PTR</i>
41A6--38E7	JR	C,418F	<i>NXT - 2 DIES</i>
41A8--D61C	SUB	1C	<i>last CHR "0"?</i>
41AA--2004	JR	NZ,41B0	<i>END if not</i>
41AC--FD350E	DEC	(IY+0E)	<i>last PRINT-POS</i>
41AF--D7	RST	10	<i>PRINT SPACE</i>
41B0--C9	RET		

FINAL-DIGIT

END

De laatste subroutine is de PRINTOUT-RTN die de significante digits op het scherm brengt. De standaard methode om een eventuele 1ste nul van het resultaat te onderdrukken is met AND F0 en dan een sprong naar het gedeelte van AD 4199 af voor de rechter digit. Dat kan echter om twee redenen niet: die onderdrukking zou ook gebeuren bij de van 358! af voorkomende volgende pagina's met resultaat en dat mag natuurlijk niet, maar bovendien zouden we aan het eind van de eerste pagina uitkomen op een halve byte en dan zou de laatste digit herhaald worden op de volgende bladzij. Daarom wordt een leading zero in LO-(HL) geSET op A. Die is uniek en wordt omgezet in een spatie, zodat alles op z'n pootjes terecht komt. Ook aan het eind van de routine wordt een eventuele laatste nul als spatie gePRINT, zie de truc toegepast in FINAL DIGIT AD 41A8/AF.

Er zij op gewezen dat de niet noodzakelijke onderdrukking van die -inderdaad niet fraaie-nullen relatief veel bytes kost: 15 voor de leading en 15 voor de trailing zero's. Ondanks het gebruiken van de relatief trage RST 10 ROM-routine is de PRINTOUT "ogenblikkelijk". In het in deel (14) van de serie "REKENEN" behandelde programma NFF-7/6 was dat meer dan 25 s per pagina. Dankzij de separate regel 6, waarin de PRINTOUT-RTN wordt aangeroepen en het gebruiken van (4079) als STORE-PTR, worden eventuele volgende bladzijden van de uitkomst verkregen met CONT/ENTER. Wil men een herhaling van de PRINTOUT zien, dan moet eerst worden ingetoetst PRINT USR 16673 (= 4121). Op het scherm komt het aantal digits van de PRINTOUT, waarbij nu de eventuele beginspatie is meegerekend, omdat dan de 1ste nibble van LO-(HL) al geSET is op A. Met PRINT USR 16764 wordt de macht van 10 gevonden en met GOTO 6 verschijnen de digits op het scherm, inclusief de beginspatie indien nodig.

VOORBEELDEN EN TIJDEN

In figuur 5A worden een aantal voorbeelden gegeven van het PRINT-resultaat, de laatste vier in vergelijking met figuur 2. 9! en 10! demonstreren duidelijk een laatste respectievelijk een 1ste nul in de "nuttige" bytes. 200! heeft zowel het een als het ander.

In figuur 5B staat een overzicht van de resultaten voor een aantal waarden van N. De tijden worden ook gegeven in vergelijking met NFF-7 en/of NFF-6. Voor h (uren), m (min) en s worden -bij gebrek aan beter- hoofdletters gebruikt. De enorme tijdwinst met NFF-MC wordt in deze tabel overduidelijk getoond.

Zoals al gezegd bij de bespreking van NFF-7/6 kost NFF-6 7/6-maal zoveel tijd als NFF-7. De relatie dat de rekentijd ongeveer evenredig is met $N \times 2.19$ gaat niet meer op voor NFF-MC. Die macht klopt nog wel tot $N = 600$, neemt daarna echter toe. De maximale N wordt, zoals hiervoor besproken, bepaald door de laagste toelaatbare waarde van HI-(HL) en die N! heeft 509 nullen. Dit blijkt voor $N = 2045$ het geval te zijn. 2045! heeft 5376 printed digits, de rekentijd is 11 min 25 s.

De ruimte van 1000 hex voor (DE) en (HL) mag worden benut op 2 bytes na, 4094d dus, waarin 8188 digits kunnen worden ondergebracht. De maximale N! die dan nog net past laat zich berekenen met de formule van Stirling, die in deel (14) is behandeld. Dat is 2727! Om dit getal te bepalen moeten we in plaats van met SEEK (HL)<>00 te werken de vaste waarden 5FFF voor HI-(HL) en 5FFF voor HI-(DE) gebruiken. Evenzo in PREPARE PRINTOUT en COMPUTE PER DIGIT. Dat kost bijna 10% extra rekentijd. 2727! bevat 7508 printed digits en heeft 679 nullen. De gemodificeerde RTN is 9 bytes korter en doet er 23 min 32 s over. De gedeelten die anders zijn dan in figuur 4B staan in figuur 6.

In het volgende artikel zal ik laten zien hoe we de besproken routines kunnen gebruiken voor zeer grote vermenigvuldigingen. Daarbij worden ook ten dele de zoeven genoemde gemodificeerde routines benut en behandeld.

H A N V A N A B B E

BULLETIN SGG

FIG. 5A VOORBEELDEN PRINTOUT

```

1 FACTORIAL=1 DIGITS*1E0
1
9 FACTORIAL=5 DIGITS*1E1
36288
10 FACTORIAL=5 DIGITS*1E2
36288
20 FACTORIAL=15 DIGITS*1E4
243290200817664
50 FACTORIAL=53 DIGITS*1E12
3041409320171337804361260816608
4768844377641568960512
100 FACTORIAL=134 DIGITS*1E24
93326215443944152681699238856266
70049071596826438162146859296389
52175999932299156089414639761685
18286253697920827223758251185210
916864
200 FACTORIAL=326 DIGITS*1E49
7886578673647905035523632139321
85062295135977687173263294742533
24435944996340334292030428401198
46239041772121389196388302576427
90242637105061926624952829931113
46285727076331723739698894392244
56214516642402540332918641312274
28294853277524242407573903240321
25740557956660222603190417032405
23517008587961789222227896237038
9737472

```

COMPUTE FOR CTR-VALUES N-1 ... 1

```

40B7--E5      PUSH HL
40B8--11FF6F  LD DE,6FFF
40B9--D5      PUSH DE
40BC--E0B8    LDDR
40BE--F1      POP HL
40BF--00      NOP
40C0--00      NOP
40C1--00      NOP
40C2--00      NOP
40C3--00      NOP
40C4--00      NOP
40C5--00      NOP
40C6--00      NOP
40C7--13      INC DE
40C8--1A      LD A,(DE)
40C9--A7      AND A
40CA--28FB    JR Z,40C7
40CB--1B      DEC DE
40CC--1B      DEC DE
40CD--23      INC HL
40CE--ED52    SBC HL,DE
40D1--E5      PUSH HL
40D2--D5      PUSH DE
40D3--45      LD B,L
40D4--4C      LD C,H
40D5--78      LD A,B
40D6--A7      AND A
40D7--2801    JR Z,40DA
40D9--0C      INC C
40DA--ED437B40 LD (407B),BC

```

2-DE EN 3-DE STUK ONGEWIJZIGD
BEHALVE DE LAATSTE 2 INSTRUKTIES

```

4116--E1      POP HL
4117--309E    JR NC,40B7

```

PREPARE PRINTOUT

```

4119--210050  LD HL,5000
411C--28      DEC HL
411D--7E      LD A,(HL)
411E--A7      AND A
411F--28FB    JR Z,411C
4121--227740  LD (4077),HL
4124--E5      PUSH HL
4125--110050  LD DE,5000
4128--13      INC DE
4129--1A      LD A,(DE)
412A--A7      AND A
412B--28FB    JR Z,4128
412D--ED537940 LD (4079),DE
4131--ED52    SBC HL,DE
4133--23      INC HL
4134--20      ADD HL,HL
4135--7E10    CP 10
4137--3004    JR NC,4130
4139--C6A0    ADD A,A0
413B--12      LD (DE),A
413C--28      DEC HL
413D--44      LD B,H
413E--40      LD C,L
413F--D1      POP DE
4140--21FF5F  LD HL,5FFF
4143--ED52    SBC HL,DE
4145--29      ADD HL,HL
4146--1A      LD A,(DE)
4147--E60F    AND 0F
4149--2002    JR NZ,4140
414B--0B      DEC BC
414C--23      INC HL
414D--227B40  LD (407B),HL
4150--C9      RET

```

FIG. 5B DIGITS/TIMING NFF-PG"S

N	NR OF DIGS PRINT	*1E	T I M I N G I N S O R H (O U R S) M (I N)		
			NFF-7	NFF-6	NFF-MC
50	53	12	16		<<
100	134	24	1M17		<1.5
200	326	49	5M44		<4
300	541	74	13M59	16M16	8.5
400	770	99	26M11	30M32	16
500	1261	148		73M58	40.5
1E3	2319	249		3H47M	2M15
2E3	5237	499			10M50

FIG. 6 "NFF-MC" MET VASTE
HI-(HL) EN HI-(DE)

DE T.O.U. FIG. 4A VERANDERDE
GEDEELTEN WORDEN GEGEVEN

(INITIALIZE)

```

40B0--265F    LD H,5F
40B2--227740  LD (4077),HL
40B5--34      INC (HL)
40B6--0C      INC C

```

COMPUTE PER DIGIT OF CTR-VALUE

ENIGE WIJZIGING:
415C--ED5B7940 WORDT 00/11FF6F

DE PRINTER

Dit artikel is geschreven voor die mensen die DE PRINTER gaan huren, maar ik acht het ook zeer lezenswaardig voor diegenen die al een eigen printer hebben. Hier en daar heb ik moeten kiezen tussen een begrijpelijk verhaal en een verhaal dat voor alle printers in alle gevallen opgaat. Namens de redactie: reageer a.u.b. op dit artikel als anderen wat aan uw kanttekeningen kunnen hebben.

De meeste grotere printers (80 koloms of meer; A4) moeten met behulp van een serieel- of parallel interface worden aangestuurd. De meest bekende standaarden daarvoor zijn respectievelijk de RS232c en de CENTRONICS. Meestal kan dan niet volstaan worden met het COPY commando om een schermafdruck te krijgen. Daar is software (een programma) voor nodig.

DE PRINTER die voor leden van de SGG te huur is, is zo'n 80 koloms printer. Het is de Sinclair QL-printer, die identiek is aan de Seikosha SP-1000. Ondermeer Philips verkoopt dit succesnummer met een ander kastje ook. Er kan naar keuze gewoon papier in (bijvoorbeeld A4) of papier met transport gaatjes aan de zijkant (pinfeed; tractorfeed). Het is een NLQ printer (near letter quality). De printer moet serieel worden aangestuurd.

Als serieel interface dat nodig is om de Spectrum te koppelen met een seriele printer, is het Sinclair Interface 1 zeer geschikt. Veel leden hebben zo'n interface (gehad). Ook het Interface 1 is eventueel te huur. In de Spectrum 128k zit een serieel interface (niet gelijk aan Interface 1) ingebouwd.

1. Wat zitten er voor lampjes en schakelaars op?

Wanneer ook de kabel in orde is, kunnen computer en printer worden aangezet. Bijna standaard zit de aan/uit schakelaar tegenwoordig ergens rechtsachter. Op alle printers zit een ON-LINE lampje en een bijbehorende schakelaar. Brandt het lampje dan is de printer aan, maar behalve dat: hij staat op de lijn. Oftewel klaar om via de lijn (de kabel) data te verwerken tot iets zichtbaars. Zet u dat lampje uit, dan kan met een tweede knopje het papier regel voor regel worden getransporteerd (LF= line feed), met een derde knopje kan een heel vel worden getransporteerd (FF= Form Feed) en met een vierde knopje kan de NLQ stand worden gekozen. De kwaliteit van de afgedrukte letter is dan veel mooier en zwarter, maar het kost ook veel meer tijd. Overigens, de laatstgenoemde drie functies (LF, FF, NLQ) zijn ook via de lijn, dus vanuit de computer te besturen. Op de printer zitten verder Dipschakelaars. Daarmee worden ondermeer de baudrate (9600) en de lengte van het papier (12 inch) ingesteld.

2. Tractorfeed of gewoon papier?

Dat moet u zelf weten. Voor gewoon papier zie vooral de afbeeldingen in de handleiding op pagina 7, 8, 9 en 10. Voor tractorfeed (oftewel Pin-feed, continuous forms) zie de pagina's 11 en 12. Voor gewoon papier moet de tractor unit worden verwijderd. Het papier wordt anders niet getransporteerd.

3. INTERFACES

a. SPECTRUM 128k

In het eerste menu zit meen ik een soort installatie optie. De computer wordt dan als het ware ook ON-LINE gezet volgens een protocol (baudrate, aantal bits, etc.) dat met de printer instelling overeenkomt. Op ieder gewenst moment kan een copy van het beeldscherm gemaakt worden. In 48k mode moet gehandeld worden als bij de 48k Spectrum, dus met Interface 1 of iets dergelijks.

b. INTERFACE 1

Voordat de printer en de computer weten dat ze met elkaar verbonden zijn moet de regel

```
FORMAT "t";9600:OPEN#3;"t"
```

worden uitgevoerd. Dat kan zo, of in een programma. LPRINT en LLIST zijn nu bruikbaar. Zie ook pagina 35-38 van de Interface 1 handleiding.

Voor het uitvoeren van een COPY moet aanzienlijk meer gebeuren. Bij sommige moderne programma's hoort een INSTALL PROGRAM (bijvoorbeeld bij ART STUDIO). Zo'n hulpprogramma past uw randapparatuur aan aan het hoofdprogramma. Bij dat INSTALL PROGRAM kies je voor Interface 1 (nr 10), geef je als baudrate 9600 op en verder tig keer ENTER. Prachtige kopieën van het scherm op verschillende formaten. Maar helaas zijn de mogelijkheden in andere programma's meestal minder fraai.

4. Programma's voor RS232c (serieel)

Aangezien de mogelijkheden om COPY 's te maken van het scherm meestal maar beperkt zijn hier twee oplossingen in BASIC. Ze zijn erg traag, maar ook zeer leerzaam. Maar door ze te bestuderen leer je de printer te begrijpen en te laten doen wat je niet voor mogelijk hield.

Het eerste programma is alleen bruikbaar wanneer de standaardkarakterset (32 koloms) van de Spectrum in gebruik is. Dus grafische toepassingen zijn er niet mee mogelijk.

```
10 FORMAT "t";9600 : OPEN#3;"t"  
20 FOR r=0 TO 19 : FOR k=0 TO 31 : LPRINT SCREEN$(r,k); :  
NEXT k : LPRINT : NEXT r
```

Het programma tast dus alle posities af en herkent daarbij de standaard karakterset. In regel 10 is de stream #3 opengezet. Bij het gebruik van LPRINT en LLIST worden de opdrachten naar stream 3 (#3) toegestuurd. Bij gebruik van een andere stream, bijvoorbeeld stream 5, moet worden opgegeven PRINT #5; in plaats van LPRINT. LPRINT is hetzelfde als PRINT #3. Zie ook pagina 21, 22 van de Interface 1 handleiding.

Voor grafische COPY 's is een programma nodig dat eventueel als subroutine in andere programma's kan worden ondergebracht. Hier een basicversie van zo'n programma. Voor een andere printer dan de Seikosha SP1000 zullen kleine aanpassingen misschien nodig zijn:

```
100 CLOSE #3
1000 FORMAT "b";9600 : OPEN #3;"b"
1020 LPRINT CHR$ 27; CHR$ 64; CHR$ 27; CHR$ 60;
      CHR$ 27;"A";CHR$ 8;
1030 FOR y=175 TO 0 STEP -8
1040 LPRINT CHR$ 27;"*";CHR$ 5;CHR$ 0;CHR$ 1;
1050 FOR x=0 TO 255
1060 LPRINT CHR$ (128*POINT (x,y)+64*POINT (x,y-1)+ 32*POINT
      (x,y-2)+16*POINT (x,y-3)+8*POINT (x,y-4)+4*POINT (x,y-5)
      +2*POINT (x,y-6)+POINT (x,y-7));
1070 NEXT x
1080 LPRINT CHR$ 13;CHR$ 10;
1090 NEXT y
```

Het programma maakt van iedere pixel op het scherm een dot op de printer. Afmeting circa 10x12cm.

Toelichting:

```
100 Sluit een eventueel geopende stream 3.
1000 Koppel stream 3 aan channel b. Channel b is een RS232
      kanaal voor het transporteren van 8 bits codes.
1020 Reset printer en printkop naar beginpositie.
      Het papiertransport wordt ingesteld op 8/72 inch. Zo
      sluiten de regels op elkaar aan. Per dot is dat 1/72 inch.
1030 De teller voor y. Met x en y wordt een bepaalde positie op
      het scherm aangegeven. Bij de Spectrum is linksonder 0,0
      en rechtsboven 255,175. Het aftasten begint dus linksboven.
1040 Ook de dichtheid in horizontale richting wordt nu 1/72. Er
      kunnen maximaal 576 dots naast elkaar. Er maar 256 dots
      nodig. Er wordt met 0 en 1 opgegeven dat er 0 + 1*256 zijn.
1050 Teller voor x.
1060 Iedere keer als dit statement wordt uitgevoerd, worden er
      door de printer acht dots onder elkaar gezet. De POINT
      functie geeft, als de pixel gezet is een 1, anders een 0.
1080 Na iedere regel moet het papier getransporteerd worden en
      moet de printkop naar het begin van de regel.
```

Als je een A4 kopy wilt zul je moeten vergroten. Dat kan door een pixel van het scherm als 3*3 dots af te drukken. Een programma daarvoor volgt hier in Basic. Als regel 1060 nu uitgevoerd wordt, worden er slechts 2 pixels verwerkt. Daar worden 6 dots van gemaakt. Er worden 2 dots niet gebruikt (jammer, want het kost al zoveel tijd), sneller kon ik het niet bedenken. Omwille van die snelheid heb ik enkele regels juist wel en enkele juist niet gecombineerd. Erg leuk om beide programma's te combineren.

```
1000 CLOSE #3 : FORMAT "b";9600 : OPEN #3;"b"
1020 LPRINT CHR$ 27; CHR$ 64; CHR$ 27; CHR$ 60;
      CHR$ 27;"A"; CHR$ 6;
1030 FOR x=0 TO 255 STEP 2
1040 LPRINT CHR$ 27;"*"; CHR$ 5; CHR$ 13; CHR$ 2;
1060 FOR y=0 TO 175 : FOR d=0 TO 2 : LPRINT CHR$ (128*POINT
      (x,y)+64*POINT (x,y)+32*POINT (x,y)+16*POINT (x+1,y)
      +8*POINT (x+1,y)+4*POINT (x+1,y));: NEXT d : NEXT y
1080 LPRINT CHR$ 10: NEXT x
```


Erg belangrijk vind ik het dat de verhoudingen intact blijven. Helaas gaat juist dat bij veel machinecode (of andere talen) de mist in. Let wel: dat ligt niet aan de apparatuur, maar aan de programmatuur. Het kan (gezien bovenstaande) wel degelijk.

6. Lettertypes

In de printer zit behalve software voor de besturing vaak ook nog software om allerlei lettertypes af te drukken. Stuur je dus vanuit de computer een code die een bepaald karakter voorstelt dan is het dus afhankelijk van de toestand van de printer hoe die letter wordt afgedrukt. De SP-1000 kan door combinaties van schuin, vergroot, verkleind, NLQ, vet en proportioneel (niet alle karakters even breed) zo'n 27 verschillende lettertypes maken.

Daarnaast zijn de mogelijkheden onbeperkt als je in een van de zeven grafische toestanden van de printer gaat werken. Je kunt dan de computer lettersets naar de printer laten sturen. Dat werkt voor alleen tekst langzamer, maar de toekomst is dat je op je beeldscherm versierde tekst maakt, en het plaatje als geheel naar de printer stuurt.

Kees van Krimpen

Verhuur

Zo vlak voor de vakantie wil ik alle leden nog eens wijzen op de mogelijkheid om bij mij apparatuur te huren. De digitizer is vrijwel continue onder de diverse leden en ook voor de andere apparatuur is redelijke belangstelling. Bij de AMX-muis is extended ART-STUDIO bij de huurprijs inbegrepen. Op verzoek kunt u bij het modem enkele dagen een toegangscode voor Viditel krijgen. Wel worden de gemaakte toegangs- en beeldkosten in rekening gebracht.

	weekprijs	borg	opm.
AMX muis	2.50	25	incl. alle benodigdh.
Modem (VTX 5000)	2.50	25	"
Speech-synthesizer	2.50	25	"
Joystick + interface	2.50	25	
printer (Seik SP1000)	10.00	200	zie tekst.
printer (Seik GP 50s)	2.50	50	
digitizer	2.50	50	
Interface 1	2.50	50	
spectrum	2.50	50	
QL	5.00	100	

Bij de printer is 20 vel papier vrij. Inktlint wordt alleen tegen kostprijs verrekend indien van zichtbare achteruitgang sprake is.

De speech synthesizer en het VTX-modem zijn nog niet geschikt gemaakt voor de 128k.

Telefoon 05920-70093 voor reservering.

Kees van Krimpen

DRUKWERK

PORT BETAALD
Groningen

Afz.:

SGG

redactieadres

F. Elstrodt
Kam. Onnesstraat 172
9727 HS Groningen

AAN:
