

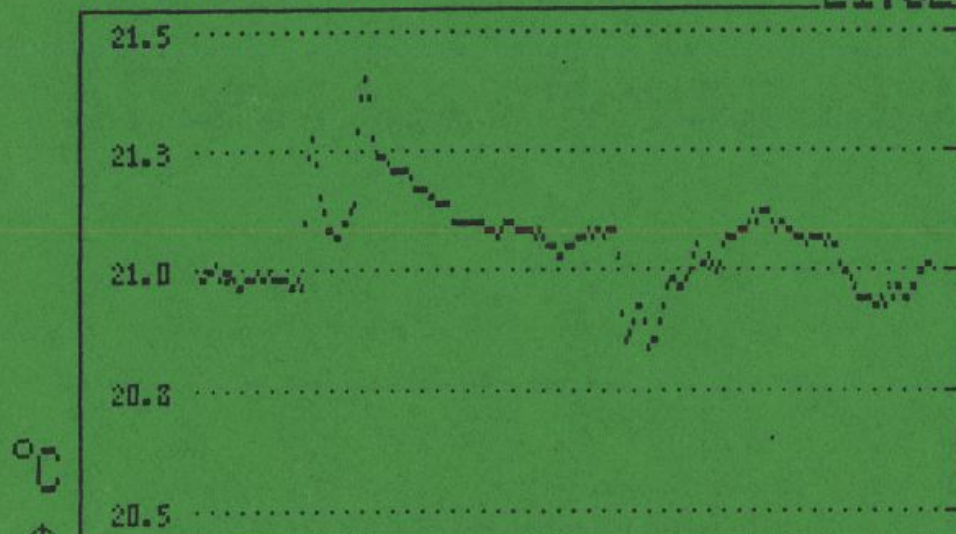
BULLETIN

SINCLAIR GEBRUIKERSGROEP GRONINGEN/ASSEN

Min: 20.84/Max: 21.40 °C

Start: 00:00 + 0

21.02



Nr: 1

COLLOFON

VOORZITTER:

Jan Dirk Burggraaf
Kluivingskampenweg 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

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.

SLUITINGSDATUM KOPIJ 2 MAART

VAN DE REDAKTIE



HALLO ALLEMAAL

We hebben deze keer een reuze leuk bulletin. Onze leden zijn enthousiast bezig met muziek, cursussen, mooie letters, leuke en of handige programma's, en nog meer. Dat Sir Clive Sinclair ook nog andere hobby's heeft kunt U zien op blz. 9 van ons bulletin. Voorlopig hoeven we dus van zijn kant geen nieuwe komputer te verwachten.

Op onze omslag ziet U een voorbeeld van Stef z'n metingen. Op 9 maart in Assen zullen we Stef zien meten aan Volts, licht en warmte.

Een van de redactieleden is zaterdag ook even op het N.A.T. wezen kijken. Dat was leuk maar ook nodig, want er liepen twee mannen uit Leeuwarden die het Sinclair spoor bijster waren. Dolblij met hun Spectrum, en dan nergens terecht kunnen met vragen en kapotte apparatuur.

Toen ze eenmaal van onze Vereniging hoorden besloten ze ter plekke lid te worden.

Het was leuk te zien dat Herman Vesper en Bert v.d. Zaag daar met Spectrum en toebehoren wat lieten zien. Waarom waren er niet meer van onze leden op zo'n gezellige dag?

Wat U in dit bulletin kunt lezen:

- Van de vice voorzitter : Relativiteit.
- Van Frans Postma : Big in Japan 128K muziek.
- Van Arjan v.d. Burg : D.O.M.
- Van de redactie : Sir Clive en z'n andere hobby.
- Van J.van Alteren : Machine-code stap voor stap 2.
- Van de redactie : Kalender met data.
- Van Martin : Basic voor beginners 9.
- Van Frans Postma : Bright on/off.
- Van Han van Abbe : Snel en precies 20b.

Binnenkort verwachten we van Stef voorbeelden van en uitleg over een plotter aan de Spectrum. Alvast bedankt Stef!



-----de redactie-----

GEBRUIKERSAVOND GRONINGEN



p donderdag 16 maart houden wij onze gebruikersavond in Groningen. De volgende gebruikersavond houden wij op donderdag 20 april.

Vooraf onze nieuwe leden raden wij aan te komen kijken, je zult versteld staan van de ideeën, hulp en informatie die je daar kunt krijgen. Maar neem vooral als het even mogelijk is je apparatuur mee, want dan gebruik je de avond pas goed.

Je treft daar medegebruikers waarvan sommige technische jongens die wel raad met je kapotte apparatuur weten, of net juist die aansluiting voor je kunnen maken die je net nodig hebt. Verder staat er een leestafel vol met Sinclair bladen uit het hele land.

Je kunt ook apparatuur huren, zoals bijvoorbeeld een AMX muis. Volgende maand zullen wij die lijst weer eens in ons bulletin zetten. Het adres van onze gebruikersavond is : School "de Wijert" van Schendelstraat 1 Groningen.



GEBRUIKERSAVOND ASSEN

De gebruikersavond in ASSEN is op donderdag 13 april in



Markehuus
Scharmberg
Assen (Peelo (wijk 7))

WIJ WILLEN DEZE



GRAAG

BEZET ZIEN

VAN DE VOORZITTER



> > * R E L A T I V I T E I T * < <

Relativiteit is een uitdrukking in onze nederlandse taal.

En netzo als zovele andere uitdrukkingen, heeft die een bepaalde betekenis.

Ik heb zaterdag 25 febr. mijn (grote) neus even laten zien op het N.A.T. en ik werd met de betekenis van deze uitdrukking geconfronteerd.

Natuurlijk heb ik ook op Das Treffen von die Radio-leute hin und her gelaufen. Ik heb er o.a. nog een collega getroffen. Hij stond n.b. naast onze eigen computerclub; ik heb er diverse 'koopjes' gezien. Iemand kreeg een doos met allerlei printen. Hij had een tientje in zijn hand, dus dat zal het hem wel gekost hebben, maar . . . hij had een glimlach voor wel 1000 gulden op zijn gezicht, dus hij zal er ongetwijfeld blij mee geweest zijn. Ik daarentegen had er geen 10 gulden, nog geen 1.--, nog geen duppie over voor die oude troep !! Ik kon er niets mee !!

De blijde glimlach was dus een relatieve uiting van een bepaalde stemming, nl. gebonden aan deze persoon op dat moment.

In de loop van de dag werd ik aangesproken door onze collega-GG uit Delfzijl/Appingedam.

Eerst even voorstellen enz., U weet hoe dat gaat. Toen kwam de verontschuldiging dat ze 4 maart helaas niet aanwezig konden zijn op onze open dag. 'Ja sinds zij op de zaterdag de bijeenkomsten hielden hadden ze 2 5 0 % vooruitgang in het aantal bezoekers geconstateerd.' Zo, zei ik, 'dat is een prachtig resultaat.' Ik dacht bij mijzelf; daar gaat onze SGG ! met maar ong. 30 bezoekers/avond. En dan te weten dat we er enkele keren geleden nog ong 60 hadden.

Maar mijn nieuwsgierigheid was net wat groter dan mijn schrik, en ik vroeg dus: 'Hoeveel hebben jullie er nu dan ?'

'Wel 20 die ook hun computer meenemen,' was het antwoord. U ziet: alles is relatief, en dus herhaal ik mijn oproep van vorige keer nog eens. Van de zolder die prachtige stukjes speelgoed, en naar het Denk-sport-centrum op 4 maart ! !

J. v. Alteren, v.v.

BIG IN JAPAN

Muziek van Frans Postma.



1>REM 128.1
2 LET a\$="T12005N3FD1CCD4F3&5&3FD1CCD4G3&F&FD1CCD4F3&5&3F
D1CCD4G3&FFFD1CCD3F3D1C3CD4AA3A06N4CC3C05N3FD1CCD3FD1C3CDFD1
CCD3GG1G3FFFD1CCD3FD1C3CD4AA3AA1G06N3C05N1G3A"
3 LET a\$=a\$+"5&3FF1E3F4E6D3FFGFEEFEEDDCC\$b7a&5&3FFE1F4E6D
3&1FF3G1F4E3FFEEDDCC6D5#C7&"
4 LET a\$=a\$+"5&1F3FE1D3CD6C1E3F4E1F4E3F5\$b1\$b4C1D4D3EFF1E
7E1&5&&1F3FE1D3CD6C1E3FEF4E3FE1D4D3D1G3F4E3D1FED5E1&&FFFFED6
D1&FF5\$b1FED6E1&EE5a1aaa5\$b1&4F1D3DEF5D3E7E"
5 LET a\$=a\$+"1FED6D1&AA5\$b3FDCEF1G5a1&3aFF1D4D3E4FE3D7E1&
FFFFED6D1&&&D4\$b5&"
6 LET a\$=a\$+"1C4g5&a&4AG5D1D4C3g7a5&3AA1A4D5&1D4\$b5&1C4g5
&a3aa4AG06N5D1D4C05N3G4A06N4C5D3FEE05N"
7 LET a\$=a\$+"1FED6D1&AA5\$b3FDCEF1G5a1&3aFF1D4D3E4FE3D7E1&
FFFFED"
8 LET a\$=a\$+"7D1\$b3\$b1\$b3\$b1\$b3\$b1C3C1C3C1CC1a3a1a3a1aa1\$b
3\$b1\$b3\$b1\$b3\$b1g3g1g3g1gg(1a3a1a3a1aa)8D5&"
9 LET b\$="05N9&&&&&4aa3a4CC3C9&&&4aa3aa1&&&&3&"
10 LET b\$=b\$+"5Aa6a\$b5DD\$bCCgg7e5&a6a\$b3&D5D\$bCCg7\$b5a7a"
11 LET b\$=b\$+"5&5a6\$b6g5g7a5ff\$b3b8C5&&a6\$b6g5C7C\$b6D3&5#C#
C1&aaaaga6a1&&&5&1&&&6C1&&&5&1&&&5&5&b6\$b7#C"
12 LET b\$=b\$+"1aga6a1&aa5f\$b1&6g5e1&3&a7\$b3b#C1&aaaaga6a1&
&a4f5&"
13 LET b\$=b\$+"1g4d5&e&4ED5a1a4g3d7e5&3EE1E4a5&1a4f5&1g4d5&
e3ee4ED5A1A4G3D4EG5A06N3C05N3AA"
14 LET b\$=b\$+"1aga6a1&aa5f\$b1&6g5e1&3&a7\$b3b#C1&aaaaga4a1a
3a1aaf3f1f3f1ff"
15 LET b\$=b\$+"1g3g1g3g1gg1e3e1e3e1ee1f3f1f3f1ff1d3d1d3d1dd
(1e3e1e3e1ee)8a5&"
16 LET c\$="03N3&&5&7&9&((4D1D5D)(4\$b1\$b5\$b)(4C1C5C)4a1a3a1
aala3a1a3a1aa)"
17 LET c\$=c\$+"4D1D5D4\$b1\$b5\$b4C1C5C4a1a5a4\$b1\$b5\$b4g1g5g4a
1a3a1aa4a1a3a1aa(4D1D5D4\$b1\$b5\$b4C1C5C4a1a5a4\$b1\$b5\$b4g1g5g4
a1a3a1aa4a1a3aa)"
18 LET c\$=c\$+"4D1D5D4\$b1\$b5\$b4C1C5C4a1a5a4\$b1\$b5\$b4g1g3g1g
g4a1a3a1aa4a1a3aa1D3D1D3D1DD1\$b3\$b1\$b3\$b1\$b3\$b1"
19 LET c\$=c\$+"1C3C1C3C1CC1a3a1a3a1aa4\$b1\$b3\$b1\$b3\$b1\$b3\$b1g1g3g1g
g(1a3a1a3a1aa)1D3D1D3D1DD1\$b3\$b1\$b3\$b1\$b3\$b1C3C1C3C1CC1a3a1a3
a1aa1\$b3\$b1\$b3\$b1\$b3\$b1g3g1g3g1gg1a3a1a3a1aa4a1a3aa"
20 LET c\$=c\$+"3aa4D1D5D4\$b1\$b5\$b4C1C5C4a1a5a4\$b1\$b5\$b4g1g3
g1gg4a1a3a1aa4a1a3aa1D3D1D3D1DD1\$b3\$b1\$b3\$b1\$b3\$b1"
21 LET c\$=c\$+"1C3C1C3C1CC1a3a1a3a1aa4\$b1\$b3\$b1\$b3\$b1\$b3\$b1g1g3g1g
g(1a3a1a3a1aa)1D3D1D3D1DD1\$b3\$b1\$b3\$b1\$b3\$b1C3C1C3C1CC1a3a1a3
a1aa1\$b3\$b1\$b3\$b1\$b3\$b1g3g1g3g1gg4a1a3a1aa4a1a3aa"
22 LET c\$=c\$+"04N4f1f3f1ff1d3d1d3d1dde3e1e3e1eec3c1c3c1ccd
3d1d3d1dd03N1b3b1b3b1bb04N(1#c3#c1#c3#c1#c#c)8f5&"
23 PLAY a\$,b\$,c\$

BULLETIN SGG

DISK OPERATING MENU

Na het kopen van mijn OPUS DISCOVERY viel mij gelijk op dat er voor dat je er wat mee kon doen veel moest intikken. Ook moest je voor het laden van een spelletje de Joystick nog even aanzetten, anders kon je nog niet echt veel beginnen. Hetzelfde gold ook voor de printer, waarbij je het kanaal ook nog moest openen.

Toen heb ik samen met een kameraad die ook een SPECTRUM heeft een OPUS utilitie gemaakt genaamd D.O.M. dat staat voor Disk Operating Menu.

We hebben een avond + nacht aan versie 3.0 besteedt om hem zo compleet mogelijk te maken wat niet echt gemakkelijk was, maar met een paar pilsjes erbij kom je een heel eind.

Na het intikken van het programma save je hem als volgt:

SAVE * 1:"run" LINE 1

Als je de computer dan aan zet en RUN geeft start deze utilitie zich zelf op.

Met enige aanpassing is deze utilitie ook geschikt voor andere diskdrive's.

SUCCEES!

Arjan van der Burg.

D.O.M. Version 3.0

- 1 - JOYSTICK (ON/OFF)
- 2 - PRINTER CHANNEL (ON/OFF)
- 3 - DIRECTORY FROM DISK
- 4 - LOAD FROM TAPE
- 5 - LOAD FROM DISK
- 6 - ERASE FILE FROM DISK
- 7 - FORMAT A DISK

```
1 REM ***** Multisound *****
2 REM ***** Presents *****
3 REM ** In Cooperation **
4 REM ***** With *****
5 REM ***** Soundtrack *****
6 REM ***** D.O.M. *****
7 REM ** Disk Operating **
8 REM ***** Menu *****
9 FORMAT "j";1: CLOSE #3
10 POKE 23609,50: LET b$="Y"
11 LET p$="C"
20 BORDER 3: PAPER 6: INK 0: CLS
30 PRINT AT 3,4;"1 - JOYSTICK (ON/OFF)"
32 PRINT AT 4,4;"2 - PRINTER CHANNEL (ON/OFF)"
35 PRINT " 3 - DIRECTORY FROM DISK"
40 PRINT " 4 - LOAD FROM TAPE"
45 PRINT " 5 - LOAD FROM DISK"
47 PRINT " 6 - ERASE FILE FROM DISK"
50 PRINT " 7 - FORMAT A DISK"
52 PRINT AT 1,7;"D.O.M. Version 3.0"
```



BULLETIN SGG

```
56 IF b$='Y' OR b$='y' THEN PRINT AT 3,25; FLASH 1; OVER 1;" "
57 IF b$='N' OR b$='n' THEN PRINT AT 3,28; FLASH 1; OVER 1;" "
58 IF P$='C' OR P$='c' THEN PRINT AT 4,28; FLASH 1; OVER 1;" "
59 IF P$='O' OR P$='o' THEN PRINT AT 4,25; FLASH 1; OVER 1;" "
60 INPUT "" CHOOSSE OPTION :";f
70 IF f<1 OR f>7 THEN GO TO 60
75 LET f=f+3
80 PRINT AT f-1,4; FLASH 1; OVER 1;" "
90 PAUSE 60: CLS
100 IF f=4 THEN GO TO 150
105 IF f=5 THEN GO TO 700
110 IF f=6 THEN GO TO 250
120 IF f=7 THEN GO TO 350
130 IF f=8 THEN GO TO 450
140 IF f=9 THEN GO TO 550
145 IF f=10 THEN GO TO 820
150 REM ** JOYSTICK **
160 PRINT AT 10,5;" JOYSTICK ON (Y/N)"
170 IF INKEY$<>"" THEN GO TO 170
180 IF INKEY$="" THEN GO TO 180
190 LET b$=INKEY$
200 IF b$='y' OR b$='Y' THEN GO TO 230
210 IF b$='n' OR b$='N' THEN GO TO 240
220 GO TO 170
230 FORMAT "j";1: GO TO 20
240 FORMAT "j";0: GO TO 20
250 REM ** DIRECTORY **
260 CLS : CAT 1
265 PRINT " PRESS <; FLASH 1;"O"; FLASH 0;"> FOR MAIN MENU"
270 IF INKEY$<>"" THEN GO TO 270
280 IF INKEY$="" THEN GO TO 280
290 IF INKEY$='O' THEN GO TO 20
300 GO TO 270

350 REM ** TAPE LOAD **
360 CLS : PRINT AT 8,3;" PRESS <; FLASH 1;"O"; FLASH 0;"> TO CONTINUE"
370 IF INKEY$<>"" THEN GO TO 370
380 IF INKEY$="" THEN GO TO 380
390 IF INKEY$='O' THEN GO TO 410
400 GO TO 370
410 CLS : PRINT AT 8,0;" D.O.M. IS LOADING FROM TAPE"
420 PRINT AT 12,0;" MULTISOUND & SOUNDTRACK": PRINT "
430 LOAD ""
450 REM ** DISK LOAD **
460 CLS
470 CAT 1: INPUT "PROGRAMNAME :";m$
480 CLS : PRINT AT 8,0;" D.O.M. IS LOADING FROM DISK"
490 PRINT AT 12,0;" MULTISOUND & SOUNDTRACK": PRINT "
500 LOAD *1;m$
550 CLS : CAT 1: PRINT ; FLASH 1;"ENTER '?' FOR MAIN-MENU."
560 INPUT "NAME OF PROGRAM TO ERASE :";w$
566 IF w$='?' THEN GO TO 20
570 ERASE 1;w$
580 CLS
585 PRINT AT 7,13; FLASH 1;w$
590 PRINT AT 10,10;" IS KILLED !"
600 PAUSE 100: GO TO 550
700 REM *** PRINTER ***
710 PRINT AT 8,7;"PRINTER CHANNEL"
720 PRINT AT 10,7;"OPEN or CLOSED ?"
```

ELECTRONICS 1989 Ltd."

ELECTRONICS 1989 Ltd."

BULLETIN SGG

```
730 IF INKEY$<>" " THEN GO TO 730
740 IF INKEY$="" THEN GO TO 740
750 LET P$=INKEY$
760 IF P$="C" OR P$="c" THEN GO TO 800
770 IF P$="O" OR P$="o" THEN GO TO 790
780 GO TO 730
790 OPEN #3;"t": GO TO 20
800 CLOSE #3: GO TO 20
810 REM ** FORMAT **
820 PRINT AT 3,4;"PRESS <"; FLASH 1;"O"; FLASH 0;"> FOR MAIN MENU"
830 PRINT AT 4,4;"OR PRESS <"; FLASH 1;"7"; FLASH 0;"> TO CONTINUE"
840 IF INKEY$<>" " THEN GO TO 840
850 IF INKEY$="" THEN GO TO 850
860 IF INKEY$="O" THEN GO TO 20
870 IF INKEY$="7" THEN GO TO 890
880 GO TO 840
890 CLS : PRINT AT 21,0;"ENTER DISK-ID (UP TO 10 CHR.) : "
900 DIM a$(10)
910 INPUT a$
915 GO SUB 950
917 CLS
920 PRINT AT 8,1;"FORMATTING CURRENT OPUS DISK : "
930 PRINT AT 10,9; FLASH 1;"[";a$;"]"
940 GO TO 980
948 REM * SUB-routine *
950 CLS : PRINT AT 4,1;" ARE YOU SURE TO FORMAT THE " :
PRINT " CURRENT OPUS DISK ("; FLASH 1 ;"Y/N"; FLASH 0;" )"
954 IF INKEY$<>" " THEN GO TO 954
956 IF INKEY$="" THEN GO TO 956
958 IF INKEY$="Y" OR INKEY$="y" THEN RETURN
959 IF INKEY$="N" OR INKEY$="n" THEN GO TO 20
960 GO TO 954
980 FORMAT 1;a$
990 GO TO 20
```



BERNADETTE'S IQ

De Britse bladen staan er vol van: Sir Clive Sinclair (48), vooral beroemd geworden met zijn vinding Z88-computer, is voornemens te trouwen met Bernadette Tynan (22). Niet om haar uiterlijk, zoals veelal beweerd wordt, maar om haar intelligentie. De zakenvrouw ontmoette de exentrieke Sinclair op een bijeenkomst van de Mensa, het genootschap voor intellectueel hoogbegaafden en in elk verhaaltje noemt men dus even de IQ-scores van het paar. Hij 159, zij 154. Volgens de sterren gaan zij een lang en gelukkig leven tegemoet, want astroloog Roger Elliot voorziet voor Sinclair 'verschillende momenten van geluk' vooral in Juli en Augustus.

MACHINE-CODE: STAP VOOR STAP 2

© J. v. Alteren, SGG; 200289.

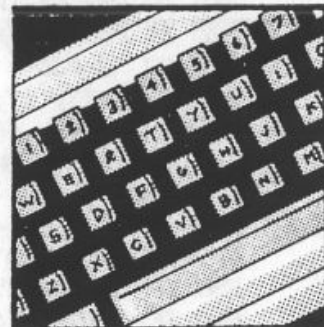
> > > MACHINE-CODE: Stap voor stap. < < <
door David Nowotnik.

Uit ZX-Computing afl.: dec/jan '85, blz. 25 e.v..

- - - - -

2e aflevering:

DEEL 1: BASIS-BEGRIPPEN. (vervolg)



Er zijn een groot aantal 8-bit opslag-plaatsen in de CPU. Deze zijn gelijk aan de bytes in het geheugen en worden 'registers' genoemd.

Zij slaan getallen op (en veranderen ze) in de CPU. Ze hebben 'namen' als A, F, B, C, D, E, H en L. Er zijn ook enkele 16 bit-registers, genaamd IX, IY, SP en PC. In de loop van deze serie artikelen zal ik (D.N. dus) ze behandelen.

De CPU wordt met de rest van de computer verbonden door een aantal verbindingen, die 'bussen' genoemd worden. Eenvoudig gezegd, deze beslaan 8 of 16 'draden' om elektrische signalen te transporteren naar en van de andere delen van de computer.

Om een opdracht te ontvangen is het eerste wat de CPU doet, de inhoud van het PC-(16 bits-)register op de adres-bus zetten. PC is de afkorting voor 'Program Counter' en het bevat het adres in het geheugen waar de volgende machinecode-opdracht wordt vastgehouden.

Dit zorgt al voor veel vragen, die ik later hoop te beantwoorden.

(Adres is m.i. hier gebruikt in de betekenis van 'opslag-plaats')

Bedrading buiten de CPU ontwaart de boodschap op de adres-bus om het juiste adres in het geheugen te activeren, of om hem toegankelijk te maken.

De inhoud van dat adres wordt op de DATA-bus gezet en dit getal wordt dan naar de CPU getransporteerd.

Dit getal is een gecodeerde instructie naar de CPU, (wordt een 'UPCODE' genoemd) die door de CPU wordt gedecodeerd en die dan volgens een vaste verwerkings-volgorde de opdrachten behorend bij die instructie, afwerkt.

Als het gehele PC-register is opgewaardeerd, (in BASIC zou dat inhouden: $LET\ PC = PC + 1$) wordt de volgende instructie/opdracht uit het geheugen gehaald.

De Z80-CPU begrijpt meer dan 800 verschillende instructies Omdat een byte (8 bits !!) al meer dan 256 verschillende getallen kan bevatten, is soms meer dan 1 byte nodig om een upcode te completeren. Er bestaan veel 2-bytes-upcodes.

Nadat de eerste byte gedecodeerd is, heeft de CPU in de gaten dat er tweede byte nodig is om de instructie volledig te maken.

Om dit getal terug te halen, wordt het PC-register met 1 vermeerderd en in dezelfde volgorde, zoals eerder beschreven, wordt het een en ander afgewerkt. Voor veel instructies naar de CPU zijn bepaalde gegevens nodig.

Deze komen in de vorm van getallen die onmiddellijk achter de instructie aankomen.

Ze worden vaak op dezelfde manier uit het geheugen gehaald als de upcode zelf, maar worden naar een ander deel in de CPU weggezet, afhankelijk van wat de upcode voorschrijft.

Deze informatie-bytes heten 'OPERANDS'. Er kunnen een of twee operands per upcode aanwezig zijn. Als gevolg daarvan kan een enkelvoudige instructie naar de CPU van een tot vier bytes lang zijn.

Andere 'draden/bussen' vanaf de CPU ontvangen en verzenden controle-signalen, die bevestigen dat alle handelingen op het juiste ogenblik plaatsvinden.

De CPU wordt gecontroleerd met een klok die bijna netzo werkt als een metronome. (dat is een apparaat dat bij het pianospelen de maat laat horen. De snelheid kan worden ingesteld.) dus die de tijd aftelt en waarmee de CPU in de pas moet lopen. Hoe sneller de klok, desto sneller de CPU zal werken, natuurlijk binnen de fabrieks-toleranties van de CPU.

ASSEMBLY LANGUAGE

(Het bovenstaande is een moeilijke term. Hij kan vertaald worden met: 'MC programma-taal' of 'MC-steno'.)

Een veel gemaakte vergissing in de terminologie is het door elkaar halen van de namen 'machine-code' en 'assembly-language'. Ik hoop dat uit mijn beschrijving duidelijk wordt wat machine code inhoudt: Het is een volgorde van getallen, die in het computergeheugen worden vastgehouden en die de CPU instructies geeft om uit te voeren.

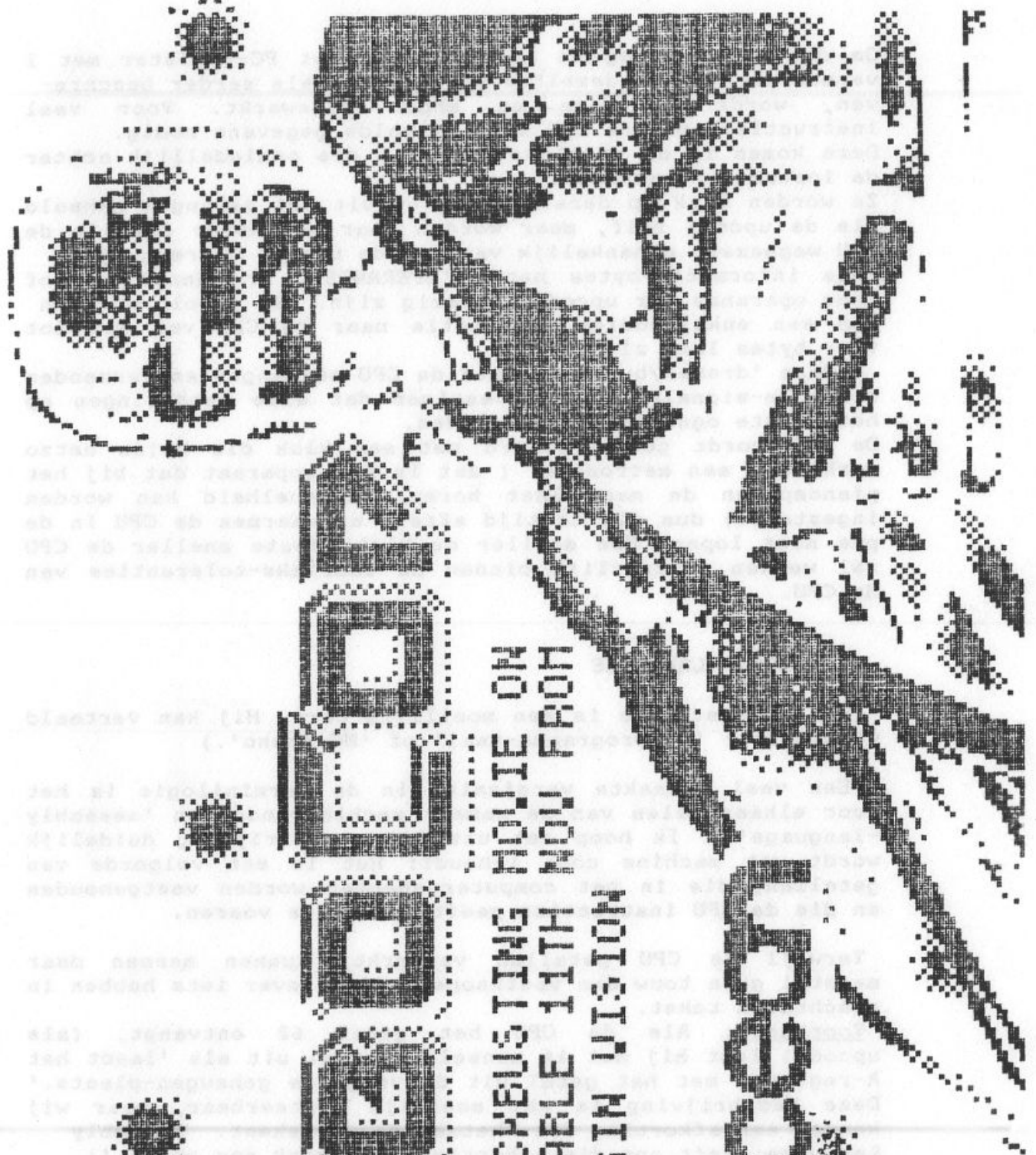
Terwijl de CPU getallen verwerkt, kunnen mensen daar meestal geen touw aan vastknopen, die liever iets hebben in geschreven tekst.

Voorbeeld: Als de CPU het getal 62 ontvangt, (als upcode) legt hij het in menselijke taal uit als 'laadt het A-register met het getal uit de volgende geheugen-plaats.' Deze beschrijving is wat moeilijk hanteerbaar, maar wij kennen een afkorting die hetzelfde betekent. Assembly Language geeft ons die afkorting in: (denk aan steno !)

- LD A,n. Betekenis: Laad A met het getal n. {In dit voorbeeld dus met 62}

Daarom is Assembly Language (AL) een beschrijvende afkorting van machinecode-handelingen.

Mc schrijven in AL is gemakkelijker dan met een speciaal programma, een 'ASSEMBLER' genoemd, AL omzetten in Machine Code. Die omgekeerde handeling heet 'Dis-Assemble.' Deze afleveringen zullen zowel MC als AL behandelen en van beide geven we voorbeelden.



LOCKED

SEE THE BLOCKBUSTING MOVIE ON
VIDEO AVAILABLE 11TH NOV FROM
VIRGIN VISION

VIDEO

11TH

HOE MC OPSLAAN EN GEBRUIKEN:

MC is een serie getallen, opgeslagen in een computer-geheugen. Deze getallen moeten op een veilige plaats worden opgeborgen, waar ze bewaard blijven, onafhankelijk van de onzichtbare bewerkingen in BASIC. Een van de veiligste plaatsen in de ZX 81 en bij de Spectrum, is boven de RAMTOP [RT] (= het hoogste adres dat door BASIC wordt gebruikt). Verlagen van het RT-adres zorgt voor een veilig gedeelte van het RAM, waarin MC kan worden geladen. Zodra het RT-adres is verlaagd, (doen we even verderop) kan MC worden ingevoerd. Nu kunnen we dat nog het beste doen door een BASIC-laadprogrammaatje te gebruiken.

Dat programma zal getals-waarden bovenop RT POKEN in een serie bytes, en die serie getallen vormen dus de machine-code 'boodschap'. De voorbeelden in dit artikel gebruiken allemaal dit BASIC laad-programma.

Als de MC op de goede plaats zit, moet je het nog activeren. De BASIC-term hiervoor is: 'USR'

USR is een functie, d. w. z. om het te completeren heeft het een zg. 'ARGUMENT' nodig. Het ARGUMENT is het start-adres van het mc-programma. De toegestane BASIC-opdrachten om mc 'aan de praat' te krijgen, zijn:

```
- PRINT USR 32000;  
- RAND USR 32000 ( op de ZX 81);  
- RANDOMIZE USR 32000 (voor de Spectrum);  
  LET T = 32000 ... IF USR = 32000 THEN . . . .
```

Er zijn er nog wel meer, maar dit zijn de meest gebruikte. De manier waarop USR werkt is als volgt:

1e opslaan van de aanwezige/lopende inhoud van het PC-register;

2e daarna het argument van USR in PC laden.

De reden dat het oorspronkelijke adres in PC wordt opgeborgen, is: de mogelijke terugkeer naar BASIC. (vooropgesteld dat terugkeren mogelijk is)

Een mc-programma lijkt vaak op een subroutine in BASIC, waarbij het BASIC-progr. wordt vervolgd vanaf de plaats waar naar BASIC wordt teruggesprongen, tengevolge van een 'terug-keer' -opdracht. (Zie 'RET' in de voorbeelden) Zoals 'RETURN' een BASIC-subroutine afsluit, is er een mc-instructie die een mc-subprogramma beeindigt. In AL is dit het codewoord 'RET'; de mc-getalswaarde = 201.

U zult dit in de meeste voorbeelden tegenkomen, omdat zonder die opdracht (of iets wat erop lijkt) de computer niet naar BASIC kan terugkeren.

Het gevolg daarvan zou zijn, dat de computer doorgaat zonder dat je in staat bent om hem te stoppen, of hij raakt totaal in de war en crasht. De enige oplossing voor beide toestanden is de voeding uitschakelen of een resetknop gebruiken. Dat gooit alles uit RAM en je moet opnieuw beginnen. Het is een van de meest frustrerende dingen bij mc, dat al bij de kleinste vergissing een crash aannemelijk is, en dat tot gevolg heeft dat je alles verliest wat in de computer is opgeslagen.

De 'gouden regel' bij mc is dan ook:

Alle mc EERST SAVEN, V O O R D A T je het laat RUNnen -voor het geval dat

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

BASICPROGRAMMA'S VOOR BEGINNERS 9.**PUNTENTELLING 4 (SLOT)**

Trouwe lezers zullen gemerkt hebben, dat in de listing van vorige maand (telling/3) een voor de hand liggende stap niet gezet is. Als we namelijk een deelnemerslijst gemaakt hebben, door uit de ledenlijst de namen te halen van diegenen die op dat moment ook werkelijk meedoen, dan is het logisch verder in het programma dan ook met de namen te werken en niet meer met de deelnemersnummers. Vorige maand lag dat ook al voor de hand, maar in de haast van het klaarmaken van de kopij, is het over het hoofd gezien. Maar nu dan toch:

In het programmablok PUNTEN INVOEREN (800 t/m 950) wordt in REGEL 860 en 870 "d" vervangen door "d\$(d)". "d" geeft het deelnemersnummer, "d\$(d)" geeft de naam. Het woord "deelnemer" kan dan vervallen. Verder wordt dezelfde wijziging toegepast in het programmablok PUNTENTELLING (1200 t/m 1360) in REGEL 1320 en in het programmablok LIJSTEN AFDRUKKEN (1600 t/m 1770) in REGEL 1670.

In dit vierde en laatste deel van het programma PUNTENTELLING zullen we nog wat TOETERS en BELLEN aanbrengen om er een mooi compleet en afgerond programma van te maken.

REMREGELS

Hiervoor wordt programmablok 100 t/m 190 vrijgehouden. We kunnen daarin allerlei gegevens van het programma vermelden. Wees hier niet te zuinig mee; na een paar maanden heb je meestal nog maar een heel vaag idee, wat je nu eigenlijk aan het doen was!

GO SUB REGELS

Programmablok 200 t/m 390 wordt gereserveerd voor GOSUB regels. Deze verwijzen naar subroutines, die aan het eind van het programma worden opgenomen.

SUBROUTINES

REGEL 210 springt naar subroutine 8000 t/m 8030, die een titelpagina op het scherm zet. Aan het eind van REGEL 8020 staat PAUSE 100. Deze instructie zorgt er voor dat de titelpagina 2 seconden op het scherm blijft staan, voordat het programma verder gaat met de volgende subroutine. Deze tijd kan natuurlijk naar believen gewijzigd worden. Het getal achter PAUSE komt overeen met de frequentie van het lichtnet. Of dat ergens op slaat weet ik niet, maar het IS wel zo. PAUSE 500 duurt dus 10 seconden, enz.

REGEL 220 verwijst naar subroutine 8100 t/m 8150 LEDENLIJST, die vorige maand uitgebreid is besproken.

Hierna volgt via REGEL 230 de subroutine PRINTER AANZETTEN (8200 t/m 8240). Deze subroutine is alleen maar nodig, als de resultaten van het programma m b v een Timex printer worden afgedrukt. Omdat deze printer apart moet worden aangezet en dit aanzetten nogal eens wordt vergeten, is deze subroutine ingevoerd als geheugensteuntje. Uiteraard kan deze subroutine later in het programma desgewenst nogmaals worden aangeroepen.

Tenslotte wordt d m v REGEL 240 naar programmablok 8300 t/m 8500 LEDENLIJST gesprongen, waarin de ledenlijst via het scherm of de printer gepresenteerd kan worden. REGEL 8310 PRINT'''' geeft 5 regels ruimte tussen het opschrift van de titelpagina en de vraag of je de ledenlijst wilt zien. Let er op dat hier geen INPUT instructie is gebruikt, zodat de vraag overal op het scherm geprint kan worden, waar we hem maar hebben willen.

REGEL 8320 is inmiddels bekend (zie blz 7 van het januarinummer) Toetsen we "n" in, dan heeft REGEL 8330 geen effect en wordt d m v REGEL 8340 het scherm leeggemaakt en de subroutine verlaten.

Toetsen we "j" in, dan wordt regel 8340 overgeslagen en gaan we via REGEL 8350 SCHERM LEEGMAKEN naar REGEL 8360. Deze print op de bovenste 2 regels de kop van de ledenlijst en geeft een regel tussenruimte.

Programmablokje 8370 t/m 8390 print de namen van de leden onder elkaar met het volgnummer er voor. De namen van de leden zijn reeds eerder aan de l\$ l t/m 24 toegekend in subroutine 8100.

Omdat de ledenlijst met meer dan 22 leden niet in één keer op het scherm kan, wordt de vraag gesteld: SCROLL ?. Uiteraard moet hier "j" worden ingetoetst. Mocht je "n" intoetsen, dan stopt het programma. Dit kan worden hersteld met CONTINUE (toets C) en ENTER. Hierna wordt met 3 regels tussenruimte (regel 8400) in REGEL 8410 gevraagd of je een uitdraai van de ledenlijst wilt hebben op de printer. Hierna kun je eventueel weer subroutine 8200 PRINTER AANZETTEN aanroepen.

REGELS 8420 t/m 8450 werken op dezelfde manier als de hierboven behandelde regels 8320 t/m 8350.

REGELS 8460 t/m 8490 werken hetzelfde als de regels 8360 t/m 8390, echter moeten we nu LPRINT gebruiken i p v PRINT ander zou de zaak weer op het scherm komen.

Zie voorbeeld 1.

LEDENLIJST FOTOCLOB februari 1989

- 1 klaas hof
- 2 jaap drup
- 3 ben gerards
- 4 arend steen
- 5 piet dam
- 6 gerrit blok
- 7 jan peters
- 8 albert stam
- 9 kees vrolijk
- 10 klaas alberts
- 11 leo smid
- 12 gerard wijdeveen
- 13 anne dikmans
- 14 martin kroon
- 15 kees van riijn
- 16 peter arkema
- 17 hetty durksma
- 18 joan smits
- 19 berend de boer
- 20 henk overeem
- 21 ans boersma
- 22 clara schoenmaker
- 23 onno koetje

voorbeeld 1

BULLETIN SGG

Met REGEL 8500 wordt de subroutine verlaten en komen we bij regel 400.

AANVULLING AANVANGSGEGEVENS

Hieraan wordt toegevoegd REGEL 430, waarmee de tellingsdatum wordt ingevoerd.

In REGEL 490 worden de aantallen beoordelaars en deelnemers op het scherm gezet, en met 2 regels tussenruimte de datum.

Na 5 regels ruimte wordt gevraagd of deze gegevens juist zijn en wordt op de bekende manier een mogelijkheid tot correctie gegeven (blz 6 januarinumner). Daar deze correctiemogelijkheid een aantal malen in het programma voorkomt, zou dat beter in een subroutine kunnen worden ondergebracht. Dit wordt verder aan de lezer overgelaten.

In REGEL 480 wordt een vierkant getekend, waarin de aantallen boordelaars en deelnemers worden afgedrukt. Dit vierkant of kader staat voor een deel op dezelfde plaats als de letters die er in horen. Omdat het kader eerst wordt afgedrukt en pas daarna de tekst, wordt deze tekst OVER het kader heen geplaatst en ontstaat het effect, zoals hier verschijnt.

AANVULLING DEELNEMERS INTOETSEN

Hier moet REGEL 605 CLEAR SCREEN worden toegevoegd om het scherm leeg te maken voor het intoetsen van de deelnemerslijst.

AANVULLING PUNTEN INVOEREN

In REGEL 830 wordt hetzelfde kader ingevoerd als in regel 480.

AANVULLING PUNTENTELLING en LIJSTEN AFDRUKKEN

REGEL 1220 geeft weer het bekende kader.

In de REGELS 1340 t/m 1360 wordt de mogelijkheid geboden al of niet een uitdraai van het puntenoverzicht te maken.

Toetsen we "j" in dan wordt regel 1360 overgeslagen en gaat het programma door met REGEL 1610, die subroutine 8600 t/m 8660 aanroept, waarmee het geheel netjes van een kop wordt voorzien. Daarna volgt REGEL 1620 COPY, waardoor bovengenoemde uitdraai wordt geregeld. Zie voorbeeld 2.

PUNTENTELLING
FOTOCUB

Datum: 27 februari 1989

3 beoordelaars
2 deelnemers

voorbeeld 2

piet dam	: 68 pnt
ben gerards	: 70 pnt

Hierna volgt het programmablok 1650 t/m 1770 voor het uitprinten van de puntenlijstjes. Zie voorbeeld 3.

Toetsen we "n" in dan gaat het programma naar REGEL 1630, waardoor ook weer de kop wordt afgedrukt, waarna de puntenlijstjes volgen.

Martin den Hollander

BULLETIN SGG

Piet dam

23

21

24

68

voorbeeld 3

ben gerards

24

22

24

70

```

100 REM *****
120 REM titel "telling/4"
140 REM © MARTIN DEN HOLLANDER
150 REM VEENDAM FEBRUARI 1989
160 REM Telling puntenbeoordeling fotoclub
170 REM Afdrukken resultaat puntentelling
190 REM *****
210 GO SUB 8000: REM (titelpagina)
220 GO SUB 8100: REM (leden invoeren)
230 GO SUB 8200: REM (printer aanzetten)
240 GO SUB 8300: REM (ledenlijst)
400 REM AANVANGSGEGEVENS*****
410 INPUT "Hoeveel beoordelaars? ";x
420 INPUT "Hoeveel deelnemers? ";y
430 INPUT "Datum ?";a$
480 PLOT 5,155: DRAW 120,0: DRAW 0,18: DRAW -120,0: DRAW 0,-18
490 PRINT " ";x;" beoordelaars": PRINT " ";y;" deelnemers": PRINT " ": PRINT a$
510 PRINT " "
520 PRINT "Zijn deze gegevens juist? (j/n)"
530 IF INKEY$<>"j" AND INKEY$<>"n" THEN GO TO 530
540 IF INKEY$="j" THEN GO TO 600
550 CLS : GO TO 400
600 REM DEELNEMERS INTOETSEN (y)*****
605 CLS
610 DIM d$(y,20)
620 FOR p=1 TO y
630 INPUT ("Toets ledennummer van deelnemer ";p;" in: ");o
640 LET d$(p)=l$(o)
650 PRINT "Dlnr ";p;" : ";d$(p)
660 NEXT p
670 PRINT : PRINT "Is deze lijst correct ? (j/n)"
680 IF INKEY$<>"j" AND INKEY$<>"n" THEN GO TO 680
690 IF INKEY$="j" THEN GO TO 710
700 CLS : GO TO 600
710 CLS
800 REM PUNTEN INVOEREN*****
810 DIM C(x,y)
820 FOR b=1 TO x
830 PLOT 5,155: DRAW 120,0: DRAW 0,18: DRAW -120,0: DRAW 0,-18
840 PRINT " ";x;" beoordelaars": PRINT " ";y;" deelnemers": PRINT " ": PRINT "beoordelaar ";
: PRINT
850 FOR d=1 TO y
860 INPUT ("Hoeveel punten voor ";d$(d);" ?");C(b,d)
870 PRINT d$(d);" : ";C(b,d);" pnt"
880 NEXT d
890 PRINT
900 PRINT "Is deze lijst correct ? (j/n)"
910 IF INKEY$<>"j" AND INKEY$<>"n" THEN GO TO 910

```

```
920 IF INKEY$="j" THEN GO TO 940
930 CLS : GO TO 830
940 CLS
950 NEXT b
1200 REM PUNTENTELLING*****
1220 PLOT 5,155: DRAW 120,0: DRAW 0,18: DRAW -120,0: DRAW 0,-18
1230 PRINT " ";x;" beoordelaars"
1240 PRINT " ";y;" deelnemers"
1250 PRINT ""
1260 FOR d=1 TO y
1270 LET punten=0: LET totaal=0
1280 FOR b=1 TO x
1290 LET punten=C(b,d)
1300 LET totaal=totaal + punten
1310 NEXT b
1320 PRINT d$(d);" : ";totaal;" pnt"
1330 NEXT d
1340 INPUT "Uitdraai van dit scherm maken ? (j/n)";r$
1350 IF r$<>"j" AND r$<>"n" THEN GO TO 1350
1360 IF r$="n" THEN GO TO 1630
1600 REM LIJSTEN AFDRUKKEN*****
1610 GO SUB 8600: REM (kop lijsten afdrukken)
1620 COPY : GO TO 1650
1630 GO SUB 8600
1650 LPRINT
1660 FOR d=1 TO y
1670 LPRINT d$(d)
1680 LET punten=0: LET totaal=0
1690 FOR b=1 TO x
1700 LET punten=C(b,d)
1710 LPRINT punten
1720 LET totaal=totaal + punten
1730 NEXT b
1740 LPRINT "----"
1750 LPRINT totaal
1760 LPRINT
1770 NEXT d
7995 STOP
8000 REM TITELPAGINA*****
8010 PRINT AT 10,10;"PUNTENTELLING": PRINT AT 11,13;"FOToclub"
8020 PRINT AT 13,6;"©Martin den Hollander": PRINT AT 14,11;"okt 88/feb 89": PAUSE 100
8030 RETURN
8100 REM LEDEN INVOEREN (23 leden)*****
8110 DIM l$(23,20)
8120 FOR n=1 TO 23
8130 READ l$(n)
8140 NEXT n
8150 RETURN
8200 REM PRINTER AANZETTEN*****
8220 INPUT "Staat de Timex printer aan? (j/n)";q$
8230 IF q$<>"j" AND q$<>"n" THEN GO TO 8220
8240 RETURN
8300 REM LEDENLIJST*****
8310 PRINT "''': PRINT "Wilt U de ledenlijst zien? (j/n)"
8320 IF INKEY$<>"j" AND INKEY$<>"n" THEN GO TO 8320
8330 IF INKEY$="j" THEN GO TO 8350
8340 CLS : RETURN
8350 CLS
8360 PRINT "LEDENLIJST FOToclub": PRINT "februari 1989": PRINT
8370 FOR s=1 TO 23
```

BULLETIN SGG

```

8380 PRINT s;" ";l$(s)
8390 NEXT s
8400 PRINT ""
8410 PRINT "Wilt u een uitdraai van de ledenlijst? (j/n)"
8420 IF INKEY$<>"j" AND INKEY$<>"n" THEN GO TO 8420
8430 IF INKEY$="j" THEN GO TO 8450
8440 CLS : RETURN
8450 CLS
8460 LPRINT "LEDENLIJST FOTOCLUB": LPRINT "februari 1989": LPRINT
8470 FOR t=1 TO 23
8480 LPRINT t;" ";l$(t)
8490 NEXT t
8500 RETURN
8600 REM KOP LIJSTEN AFDRUKKEN*****
8610 LPRINT "PUNTENTELLING": LPRINT "FOTOCUB": LPRINT
8620 LPRINT "Datum:";a$: LPRINT
8660 RETURN
9500>REM DATAREGELS LEDENLIJST*****
9510 DATA "klaas hof","jaap drup","ben gerards","arend steen","piet dam"
9520 DATA "gerrit blok","jan peters","albert stam","kees vrolijk","klaas alberts"
9530 DATA "leo smid","gerard wijdeveen","anne dikmans","martin kroon","kees van rijm"
9540 DATA "peter arkema","hetty durksma","joan smits","berend de boer","henk overeem"
9550 DATA "ans boersma","clara schoenmakers","onno koetje"

```

BRIGHT ON/OFF

```

1 CLS : REM BRIGHT ON /OFF
2 PRINT "BRIGHT SWITCHERF.POSTMA 1988"
3 PRINT ""INPUT ADRES:";: INPUT ADR: IF ADR+30>65367 OR ADR+3
0<23800 THEN PRINT FLASH 1;"ERROR": GO TO 3
4 PRINT ADR: PRINT AT 21,10; FLASH 1;"POKING"
10 DEF FN A(X$,B)=CODE X$(B)-48-7*(CODE X$(B)>57)
20 DEF FN B(X$)=16*FN A(X$,1)+FN A(X$,2)
30 DATA "21","00","58","0E"
40 DATA "02","06","C2","CB"
50 DATA "76","20","04","CB"
60 DATA "F6","28","02","CB"
70 DATA "B6","23","10","F3"
80 DATA "79","FE","00","C8"
90 DATA "0D","06","FF","18"
100 DATA "EA","00","00","00"
110 LET T=0: FOR F=ADR TO ADR+30
120 READ X$: LET X=FN B(X$): PRINT AT 10,9;(ADR+31)-F,: POKE F,X
: LET T=T+X: NEXT F
125 IF T<>2971 THEN PRINT "ERROR IN DATA.....": STOP
130 CLS : PRINT "POKED.....""SAVE MET:CODE ";ADR;","30""RUNNEN
MET ~RANDOMIZE USR ";ADR;""~"
140 STOP
150 SAVE "BRIGHT" LINE 1
155 STOP
160 SAVE *"M";1;"BRIGHT" LINE 1

```


FOR/NEXT-LOOPS BASIC VERSUS MC

In figuur 4 staat het BASIC-PG voor 9 FOR/NEXT-LOOPS 9....1. Dit programma doet hetzelfde als het eerste deel van de zojuist besproken MC-routine zonder CALL TEST-G\$, maar met tussenvoeging van DEC C/JP NZ 4086/RST 08; 9 na AD 4146 en voor de subroutine WRITE-\$. Vanzelfsprekend kunnen dan de PUSH- en POP-instructies uit die subroutine worden weggelaten en moet JR Z, 411F worden: JR 411F met aangepaste sprong. Regel 100 is equivalent met het schrijven in \$-STORE dat die subroutine doet. De variabelen zijn geluk aan de registers wat C, B, E en D betreft, de overige Q, P, S, R en Z korresponderen met de registers C', B', E', D' en A'.

Het aantal malen dat een G\$ met 9 als 1ste digit wordt gevormd is $3 \times 7! = 15120$. De laatste digit kan namelijk alleen 7, 3 of 1 zijn. Dit aantal geldt ook voor 7, 3 en 1 als beginndigit. Met een 0 beginnend kan de laatste digit 9, 7, 3 of 1 zijn, dus het aantal mogelijkheden is $4 \times 7! = 20160$. Evenzo voor 6, 5, 4 of 3 als 1ste digit. Totaal zijn er dus $(4 \times 3 + 5 \times 4) \times 7! = 161280$ verschillende waarden van G\$. De MC-RTN berekent en schrijft die in 1min 58s, het BASIC-PG doet er meer dan 92 uur over! Eens te meer blijkt hoeveel sneller het rekenen in machinetaal verloopt, in dit geval ruim 2800-maal. Met een 9 als eerste digit heeft de MC-RTN 12.6s nodig, terwijl dat met een 0 beginnend 13.6s is. Die laatste tijd is slechts een kleine 9% meer dan de eerste, alhoewel de "output" $1/3$ meer is. Dat komt doordat de "kans" op een goede laatste digit ook $1/3$ meer is.

RESULTATEN

In figuur 5 staan de 9 paren met de oplossingen van ons probleem geprint. Zoals gezegd eerst de G\$ en dan priem N. Aangezien het PRINTEN van N via de instructie PRINT-FP (AD 42FC) volgt, worden slechts 8 digits korrekt weergegeven, een welbekende eigenschap van onze ZX 81. Daarom zijn de laatste 2 cijfers van de eerste twee waarden van N afgerond. In de figuur vermeld ik de juiste er achter. Vanzelfsprekend is het mogelijk ook die waarden van N exact te PRINTEN met behulp van de CALCULATOR subroutine 'n-mod-m', zoals ik in eerdere artikelen heb gedaan. Maar daar de PRINTOUT van G\$/9 eigenlijk "redundant" is, heb ik dat hier achterwege gelaten.

Nu de TIMING van de diverse subroutines en de hoofdroutine. Het maken van de PRIMES TABLE met RAND USR 17179 duurt ruim 14s. De invloed van het gebruik van zo'n tabel wordt gedemonstreerd door het vergelijken van

de rekestijd met het programma van figuur 2A, ex REKENEN (8):
 $-123456679 = 9 \times 13717631$ gaat van 2.0s zonder tabel naar 1.0s met tabel
 $-987654231 = 9 \times 109739359$ van 3.9s naar 1.7s.

Het zoeken en PRINTEN van de 9 paren duurt 41s. Per oplossing varieert het aantal te onderzoeken waarden van G\$ van 1 tot 17; zie de laatste kolom van figuur 5, waar ik die aantallen heb aangegeven. De oplossing 298765431 wordt het snelst gevonden, in ruim 1s namelijk. Die met de uitkomst 498753621 kost het meeste tijd: 4 1/2 s. Dat ik zo precies weet wat het aantal te onderzoeken waarden is, komt doordat de CALCULATOR subroutine 'val' (AD 18A4 in de ROM) iedere string, behalve de eerste (!?) kopieert naar de 'workspace', te vinden in de RAM met de pointer E-LINE, AD 4014/15 h. Na afloop van het programma zijn alle G\$s behalve 987654321 keurig af te lezen. En te tellen zoals ik heb gedaan.

De volgende keer behandel ik een BASIC-programma ter oplossing van een HCC probleem.

HAN VAN ABBE

FIG. 4 9 FOR/NEXT LOOPS 9....1

```

5 FOR C=9 TO 1 STEP -1
10 FOR B=9 TO 1 STEP -1
15 IF B=C THEN GOTO 180
20 FOR E=9 TO 1 STEP -1
25 IF E=C OR E=B THEN GOTO 170
30 FOR D=9 TO 1 STEP -1
35 IF D=C OR D=B OR D=E THEN GOTO 160
40 FOR Q=9 TO 1 STEP -1
45 IF Q=C OR Q=B OR Q=E OR Q=D THEN GOTO 150
50 FOR P=9 TO 1 STEP -1
55 IF P=Q OR P=C OR P=B OR P=E OR P=D THEN GOTO 140
60 FOR S=9 TO 1 STEP -1
65 IF S=Q OR S=P OR S=C OR S=B OR S=E OR S=D THEN GOTO 130
70 FOR R=9 TO 1 STEP -1
75 IF R=Q OR R=P OR R=S OR R=C OR R=B OR R=E OR R=D THEN GOTO 120
80 FOR Z=9 TO 1 STEP -2
85 IF Z=5 OR Z=C OR Z=B OR Z=E OR Z=D OR Z=Q OR Z=P OR Z=S OR Z=R THEN GOTO 110
100 LET G$=STR$ C+STR$ B+STR$ E+STR$ D+STR$ Q+STR$ P+STR$ S+STR$ R+STR$ Z
110 NEXT Z
120 NEXT R
130 NEXT S
140 NEXT P
150 NEXT Q
160 NEXT D
170 NEXT E
180 NEXT B
190 NEXT C
    
```

FIG. 5 PRINTOUT "9 * PRIME"

987654231	109739360 (59)	2
912345687	101371760 (63)	22
897652143	99739127	10
812345769	90260641	9
798664321	88739369	1
712364889	79151881	14
698763491	77639289	6
612364987	68839443	16
598763241	66629249	8
512346987	66927443	4
498763021	65417069	17
412369867	45817763	12
398764821	44307189	3
312468897	34717633	8
298766431	33196159	1
213466987	23717443	4
198765243	22085027	2
123468679	13717631	7

41C8--D9	EXX	r'
41C8--5D	LD E,L	
41CC--7B	LD A,E	
LOOP-E' 41CD--89	CP A,C	A = E' = "1"... "9"
41CE--285A	JR NZ,422A	E'-NEXT if E' = C'
41D0--B8	CP B,B	if E' = B'
41D1--2857	JR NZ,422A	r
41D3--D9	EXX	
41D4--89	CP C,C	E'-NXT if E' = C
41D5--2852	JR NZ,4229	if E' = B
41D7--B8	CP B,B	if E' = E
41D8--284F	JR NZ,4229	if E' = D
41DA--B8	CP E,E	r'
41DB--284C	JR NZ,4229	
41DD--8A	CP D,D	A = D' = "1"... "9"
41DE--2849	JR NZ,4229	D'-NEXT if D' = C'
41E0--D9	EXX	if D' = B'
41E1--55	LD D,L	if D' = E'
41E2--7A	LD A,D	r
LOOP-D' 41E3--89	CP C,C	D'-NXT if D' = C
41E4--283D	JR NZ,4223	if D' = B
41E6--B8	CP B,B	if D' = E
41E7--283A	JR NZ,4223	if D' = D
41E9--B8	CP E,E	
41EA--2837	JR NZ,4223	
41EC--D9	EXX	
41ED--89	CP C,C	
41EE--2832	JR NZ,4222	
41F0--B8	CP B,B	
41F1--282F	JR NZ,4222	
41F3--B8	CP E,E	
41F4--282C	JR NZ,4222	
41F6--8A	CP D,D	
41F7--2829	JR NZ,4222	
41F9--88	EX AF,AF"	a'
41FA--7D	LD A,L	
LOOP-A' 41FB--FE21	CP A1	A' = "5" ?
41FD--281D	JR NZ,421C	A'-NEXT if A' = "5"
41FF--B9	CP C,C	A' = "1"/"3"/"7"/"9"
4200--281A	JR NZ,421C	A'-NEXT if A' = C
4202--88	CP D,D	if A' = B
4203--2817	JR NZ,421C	

BULLETIN SGG

4205--BB	CP	E	
4206--2814	JR	Z,421C	if A' = E
4208--BA	CP	D	
4209--2811	JR	Z,421C	if A' = D
420B--D9	EXX		r'
420C--B9	CP	C	
420D--280C	JR	NZ,421B	A'-NXT if A' = C'
420F--B8	CP	B	
4210--2809	JR	Z,421B	if A' = B'
4212--BB	CP	E	
4213--2806	JR	Z,421B	if A' = E'
4215--BA	CP	D	
4216--2803	JR	Z,421B	if A' = D'
TO WRITE 4218--1837	JR	4251	WRITE
NEXT-G\$ 421A--08	EX	AF,AF"	a'
A'-NXT 421B--D9	EXX		r
A'-NEXT 421C--3C	INC	A	
421D--3C	INC	A	A'-CHR "STEP +2"
421E--BC	CP	H	A' < "A" ?
421F--38DA	JR	C,41FB	LOOP-A' if so
4221--08	EX	AF,AF"	a'
D'-NXT 4222--D9	EXX		r'
D'-NEXT 4223--14	INC	D	
4224--7A	LD	A,D	
4225--BC	CP	H	D' < "A" ?
4226--20BB	JR	NZ,41E3	LOOP-D' if so
4228--D9	EXX		r'
E'-NXT 4229--D9	EXX		r'
E'-NEXT 422A--1C	INC	E	
422B--7B	LD	A,E	
422C--BC	CP	H	E' < "A" ?
422D--209E	JR	NZ,41CD	LOOP-E' if so
422F--D9	EXX		r'
B'-NXT 4230--D9	EXX		r'
B'-NEXT 4231--04	INC	B	
4232--78	LD	A,B	
4233--BC	CP	H	B' < "A" ?
4234--2084	JR	NZ,41BA	LOOP-B' if so
4236--D9	EXX		r'
C'-NXT 4237--D9	EXX		r'
4238--0C	INC	C	
4239--79	LD	A,C	
423A--BC	CP	H	C' < "A" ?
423B--C2A641	JP	NZ,41A6	LOOP-C' if so
423E--D9	EXX		r
D-NEXT 423F--14	INC	D	
4240--7A	LD	A,D	
4241--BC	CP	H	D < "A" ?
4242--C29741	JP	NZ,4197	LOOP-D if so
E-NEXT 4243--1C	INC	E	
4244--7B	LD	A,E	
4245--BC	CP	H	E < "A" ?
4246--C28D41	JP	NZ,418D	LOOP-E if so
B-NEXT 4248--04	INC	B	
4249--78	LD	A,B	
424C--BC	CP	H	B < "A" ?
424E--C28741	JP	NZ,4187	LOOP-B if so
WRITE 4251--320950	LD	(5009),A	store 9. CHR
4254--08	EX	AF,AF"	a
4255--ED530750	LD	(5007),DE	store 7. & 8. CHR
4259--ED430550	LD	(5005),BC	store 5. & 6. CHR
4260--05	PUSH	DE	stack D'E'
426E--C5	PUSH	BC	stack B'C'
426F--D9	EXX		r
4260--ED530350	LD	(5003),DE	store 3. & 4. CHR
4264--ED430150	LD	(5001),BC	store 1. & 2. CHR
4268--C5	PUSH	BC	stack BC
4269--05	PUSH	DE	stack DE
426A--CD8342	CALL	4283	TEST-G\$
426D--211D26	LD	HL,261D	H = "A" L = "1"
4270--D1	POP	DE	retrieve DE
4271--C1	POP	BC	retrieve BC
4272--D9	EXX		r'
4273--C1	POP	BC	retrieve B'C'
4274--D1	POP	DE	retrieve D'E'
4276--211D26	LD	HL,261D	H' = "A" L' = "1"
4278--28A0	JR	Z,421A	NEXT-G\$
CONV + NEXT 427A--D9	EXX		r
427B--79	LD	A,C	convert C-CHR
427C--95	SUB	L	into next-lower
427D--4F	LD	C,A	digit C
427E--C25540	JP	NZ,4086	C-LOOP if C > 00
4281--CF08	RST	08.9	Stop with Report-Code 9

PORT BETAALD
GRONINGEN

DRUKWERK

AFZ:

REDAKTIE SGG:

Mevr. F. Elstrodt

Kam. Onnesstraat 172

9727 HS Groningen