

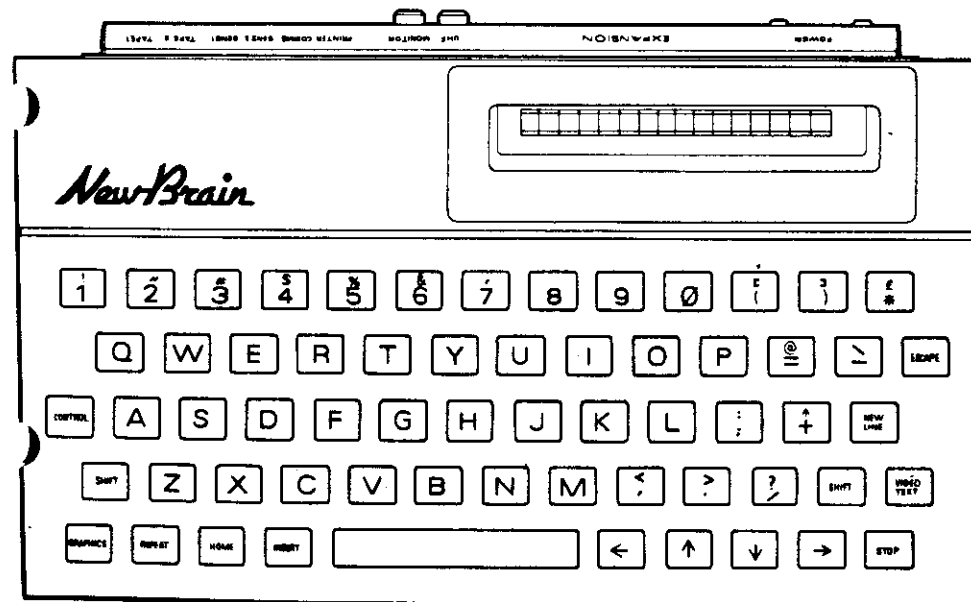
Smed 9257 RS :Smit 7524 CW :Snoo 9351 DB :Sni, 1273 AP :Sonn 5922 XJ :Spal 3431 SK :Spek 6531 CK :
 Spie 7522 HL :Steij 5914 VR :Steu 1381 AJ :Stev 1092 JV :Stev 7061 BH :Stol 1504 AT :Stro 4384 LD :
 Tax, 7545 MC :Tech 1025 RX :Tech 2105 SK :Tech 1034 LL :Thie 2316 AC :Thij 7003 DT :Tilb 4902 AE :
 Timm 7151 MK :Tolk 6602 XA :Toni 5368 AJ :Trui 3512 NP :Unia 3972 XB :Uyld 1353 CH :Valk 2713 CE :
 Vals 1215 RL :Veen 3295 VC :Veld B-2600 :Veld 4475 AG :Veld 5343 XK :Veld 5051 BT :Velt 4384 BP :
 Verb 3061 GN :Verb 1628 HK :Verb 3811 LR :Verc 5144 CK :Verh 9724 AW :Verh 8070 AB :Verh 6545 NK :
 Vera 7944 VM :Vera 3174 EN :Vera B-8200 :Vinc B-2730 :Vish 1093 TP :Visa 9247 BG :Voor 3641 AK :
 Vooij 2722 SC :Vos, DK V-16 :Voss 5212 PH :Vree 1402 GA :Vreu 2582 XK :Vrev B-2400 :Vrie 1484 PC :
 Vrie 3581 SX :Vrij 3116 AJ :Wair 7626 LW :Wann 7221 AC :Weid 7514 CR :Weij 5014 BL :Welm 7152 GP :
 Welt 5708 JD :Weni 3203 BG :Werk 1078 GL :Wess 9741 AE :West 4333 EM :Wiel 1214 CP :Wiel 4561 AC :
 Wier 9602 NP :Wier 5161 GH :Wijg 4905 AX :Wijk 5042 BH :Wijn 7201 HX :Wijn 3828 VC :Wilk 7431 CG :
 Wilk 8608 VV :Wilm 2402 AL :Wiss 7511 JK :Wiss 7531 HP :Wit, 7511 LK :Wit, 6532 XL :Wit, 7316 DS :
 Woer 7558 NK :Wort 8442 BJ :Woud 1121 CP :Wout 1703 KB :Zane 1509 ZA :Zant 7311 AA :Zeeg 2034 SP :
 Zome 1315 CL :Zuur 1000 BP :Zwaai 2811 AN :Zwan 3435 BH :Zwar 2201 BN :

New Brain

on-line

UITGAVE VAN DE
 NEWBRAIN
 GEBRUIKERS GROEP

84-3



NEWBRAIN-ON-LINE is een officiële uitgave van de **NewBrain Gebruikers Groep Nederland**. Zodra er voldoende, voor de NewBrain gebruiker, nuttige kopij voorhanden is volgt er een nieuwe uitgave. De bladen zijn licht gelijmd zodat ze eenvoudig los te maken en daarna in een klapper op te bergen zijn. De perforatie is afgestemd op de klapper met H.C.C. opdruk. Deze is tijdens de landelijke NewBrain gebruikers dagen of bij de H.C.C.te koop.

Er bestaat de mogelijkheid om zich te abonneren op regelmatige toezending. Zie het aanmeldingskaartje achter in de uitgave. De prijsstelling zal ongeveer liggen op Fl. 5.50 voor 50 pagina's, plus Fl. 2.50 voor administratie- en verzendkosten.

De redactie ziet graag kopij van de leden tegemoet. Het liefst getypt op A4 formaat; laat links en rechts een marge van 3.5 cm. en boven en onder een marge van 4.5 cm. blank.

Geplaatste artikelen mogen alleen voor niet-commerciële doeleinden onder bronvermelding overgenomen worden.

Voor informatie omtrent commerciële advertenties of advertenties van leden kunt U contact opnemen met de redactie.

Het is voor de redactie nu eenmaal onmogelijk om alle ingezonden artikelen en programma's te controleren. De aansprakelijkheid voor de ingezonden stukken ligt derhalve dan ook bij de inzender.

Hoofredacteur:

C.L. Truijens
Hamburgerstraat 19
3512 NP Utrecht
030-311087

**LET OP: Nieuwe penningmeester; nieuw gironummer
zie pagina 3**

PRIJS PER EXEMPLAAR :

**Kopij voor OnLine 84-4 voorlopig
opsturen naar Bert Hovius**

De NewBrain gebruikersgroep

Voorzitter	E. Hovius Bitterstraat 58 8011 XM Zwolle 038-228252
Secretaris	R. van Albada Talmastraat 20-II 1073 JX Amsterdam 020-739671
Penningmeester	J. de Vries Haagstraat 16 3581 SX Utrecht 030-511243 postrek: 2700380
Redacteur HCC-Nieuwsbrief NewBrain On-line	vakature

Kontaktpersonen

Onderwijs	Pieter van Dijk	01751-12367
CP/M	Paul Reinking	02510-44687/36493
Tekstverwerking	Rob van Albada	020-739671
Viditel	Bob Rijnbeek	02995-2264
Basicode	Pieter van Dijk	01751-12367
Hardware	Rob Maris	055-424485
Data comm.	Theo Kamperman	03438-17084
Pascal	Vakature	
Assembler	Bob Rijnbeek	02995-2264
Inkoop	Jan de Vries	030-511243

Softwarebiblio- theek	Jos Hermans	Postbus 2286 5001 CG Tilburg
CP/M bibliotheek	Paul Reinking	02510-44687/36493

Regionale bijeenkomsten

Utrecht	030-511243
Limburg	013-552530 / 04498-53507
Overige	030-511243

Voorwoord

Fantastisch, iedereen heeft weer goed zijn best gedaan. Ondanks het zomerseizoen een flink deeltje On-line 84-3. Dit keer vooral veel artikelen over de standaard NewBrain, d.w.z. de 32k cassette-uitvoering

Dit boekje is als volgt ingedeeld. Eerst komt er wat 'clubnieuws', dan artikelen over de NewBrain hard-ware, vervolgens tips en software voor de un-expanded NewBrain en dan hetzelfde voor expanded/disk machines. Er is nog een gebruiksaanwijzing voor een programma uit de gebruikersbibliotheek, n.l. voor het hulpprogramma PROTO. Dan zijn er drie artikelen over programmeertalen, Forth, Basicode en Pascal en we besluiten met de ledenlijst. Deze is weer op postcode gesorteerd; dat is de vorige keer goed bevallen; zoek je een naam dan kun je gebruik maken van de verkorte lijst op naam.

Als nu de leden van de gebruikersgroep niet de hele zomer achter hun NewBraintje hebben doorgebracht, dan hebben ze nu vast nog genoeg energie en frisse ideeën om nummer 84-4 te vullen. Stukken graag toesturen aan Bert Hovius, zolang er nog geen nieuwe redacteur gevonden is. Dit is n.l. het laatste nummer waarvan ik de redactie heb gevoerd. Houd je s.v.p. aan de typaanwijzingen op pagina 2 van dit nummer. Doe als het even kan een vers lint in printer of schrijfmachine en stuur het origineel op (dus geen kopie). Handgeschreven tekst kan eventueel ook, die wordt dan door de redactie getypt. Stuur van programma's een directe listing op; typ de listing dus niet over t.b.v. het artikel. Dit geeft de beste garantie voor de afwezigheid van typefouten.

Kees Truijens

LET OP: Nieuwe penningmeester; nieuw gironummer zie pagina 3

Kopij voor OnLine 84-4 voorlopig opsturen naar Bert Hovius

NewBrain Gebruikers Groep Software Bibliotheek

Postbus 2286 5001 CG Tilburg

Het is nu weer 4 maanden geleden dat de vorige software-catalogus is verschenen. Als men de nieuwe software-catalogus bekijkt dan valt meteen op dat er niet zo heel veel nieuws te melden is. Waardoor zou dit toch komen? Ligt het misschien aan de tussenliggende vakantie? Aan nieuwe programma's, geschreven door de leden, kwam gedurende deze 4 maanden, nog geen 'handvol' programma's binnen. Toch wel gek, bij meer als 450 leden!

Zouden er mensen zijn, die leuke, leerzame, handige, interessante programma's ontwikkeld hebben, hou deze dan niet voor U alleen. Laat andere NewBrain-gebruikers ook meeprofiteren. Het is vaak ontzettend tijdrovend en ook demotiverend, als je 'langs elkaar zit te werken'. We zijn, als het ware toch een 'grote NewBrain familie'. Als je programma's, naar de software-bibliotheek opstuurt, vermeld dan wel duidelijk de volgende punten:

- Naam, adres etc.
 - Toepassing van het programma (wat doet het programma of behoort het programma te doen).
 - Geef ook een korte handleiding/beschrijving bij het programma.
 - Geef aan op welk systeem het programma werkt (UNEXPANDED-DISC-EXP.)
- Dit alles kan heel eenvoudig in de vorm van REM-statements in het programma worden opgenomen. Je kunt je programma's opsturen, zowel op cassette als op diskette (vermeld wel het formaat!), naar de NewBrain Software-Bibliotheek. Na ontvangst van de programma's, wordt in overleg met de redactie van On-Line, bekeken of de programma's in de software-bibliotheek kunnen worden opgenomen. Als er geen nieuwe programma's binnenkomen, kan er ook niets nieuws worden besteld.

Er zijn toch nog enkele leuke nieuwigheden te melden:
Per 22 september 1984 zijn o.a. de volgende programma's aan het bestand toegevoegd.

- Cursus machineschrijven. (2 uitvoeringen)
- Muziekcursus (met 'echte' noten, ladders en sleutels)
4 delig met demo-programma.
- Grafische cursus, eigenlijk voor elk NewBrain-gebruiker een 'must'.
3 delig. Mensen die in het bezit zijn van de 20 delige NewBrain Basic Cursus zullen hieruit weinig nieuws leren.
- Anthill: een Pacman-achtig spel met 3 speelvelden.
- Globe: tekent de wereldbol vanuit verschillende gezichtspunten.
- Sioritme: met leuke grafische weergave.
- Hanoi: 2 uitvoeringen, met 6 en 7 ringen en een programma dat de snelste 'weg' zoekt en demonstreert.
- Houthakker: vindt de verstopte goudklompjes in het bos en verzamelt zoveel mogelijk punten. Met 3 moeilijkheidsgraden en machinetaal-routines.
- Contest: stopwatch-programma. Maakt gebruik van de ingebouwde klok van de NewBrain.
- etc.

Alle in de Software-catalogus genoemde programma's, kunnen worden geleverd op cassette en op diskette (zowel 200k, 400kss, 400kds en 800k formaat). Geef bij bestelling wel duidelijk aan, welk formaat gewenst wordt. Alle bestelde cassettes worden verwerkt, met Sanyo Data-recorder. Als U een of meerdere programma's onverhoopt niet niet zou kunnen inlezen, grijp dan niet DIRECT naar de telefoon of 'klim in de pen'. Probeer dan eerst een ander VOLUME of TOONHOOGTE. Blijft U problemen hebben met inlezen van de cassettes, lees ze dan in op een 'vreemde' cassette-recorder en save ze dan opnieuw op Uw eigen recorder. Zouden al deze handelingen nog steeds geen effect hebben, laat dan even iets weten.

September 1984,

Jos Hermans
Software-Bibliothecaris
NewBrain Gebruikers Groep.

VAN HET SECRETARIAAT

Zoals je weet is het lidmaatschap van de HCC voorwaarde voor de toetreding tot de NewBrain gebruikersgroep. Verruweg de meeste NBGG leden zijn daarom ingeschreven bij de HCC en staan op de ledenlijsten die de HCC ledenadministratie ons regelmatig opstuurt.

Maar van de 460 bij ons ingeschreven leden zijn er ca. 80 niet te vinden op de HCC lijsten. Voor een deel zijn dat nieuwe leden, die nog niet door de HCC-ledenadministratie verwerkt zijn. Anderen echter zijn opzeggers, wanbetalers of mensen die eenvoudig vergeten zijn zich bij de HCC te melden. Al degenen die bericht hebben gehad dat hun HCC inschrijving niet in orde is, moeten maar voor zichzelf uitmaken tot welke groep ze behoren en dienovereenkomstig maatregelen treffen.

Toezending van de HCC Nieuwsbrief is geen bewijs van lidmaatschap. Let op het codenummer op de adresstrook. De eerste twee tekens geven je uitsluitel.

Verklaring: L op de 2e plaats (bijv. 4L18936) gewoon lid.

C op de 2e plaats (bijv. 4C18256) idem, wil geen mailings.

D op de 2e plaats: donateur.

XP waarna nummer, bijv. XP12345: heeft proefnummer gehad.

XN waarna nummer, bijv. XN07903: lidmaatschap beëindigd.

Wij hebben al degenen die niet een L of een C op de 2e plaats hadden bericht gestuurd (met excuses aan een klein aantal donateurs).

Opzeggingen of adreswijzigingen s.v.p. doorgeven aan zowel de HCC als aan de NBGG ledenadministratie.

Binnenkort zullen we het hele bestand doorlichten op ongeldige inschrijvingen en al degenen die langer dan een nader vast te stellen termijn bij ons bekend waren maar nooit ingeschreven waren bij de HCC, uit het bestand verwijderen.

Abonnementen op on_line (196 medeleden gingen U voor!) kun je opgeven bij de NBGG ledenadministratie.

Dude nummers: uitsluitend te bestellen bij de PENNINGMEESTER (Jan de Vries). Ook voor andere eenmalige bestellingen, zoals Softwarecatalogi e.d. moet je bij de penningmeester zijn en niet bij mij.

On_line 1 is overigens uitverkocht en de voorraad van on_line 2 bedraagt nog slechts enkele tientallen.

De secretaris,

Rob van Albada.

The continuing story of NewBrain

Over de geschiedenis van de NewBrain is al veel gezegd en geschreven. In 1978 geboren in het brain van Sir Sinclair heeft deze baby een turbulente jeugd gekend. De research afdeling van Newbury Laboratories adopteerde dit geesteskind en kondigde in de P.C.W. van juni 1980 de werkelijke geboorte aan. De opzienbare specificaties verbaasde de computer insiders. De benodigde kracht om op grote schaal de wereld te veroveren moest echter gevonden worden in een nieuw op te richten dochter van het Grundy concern "Grundy Business Systems". De voortvarende aanpak vanaf midden 1982 leidde tot een explosieve groei en torenhoge investeringen. In eerste instantie herkenden insiders de kracht en mogelijkheden van het operating systeem met de uitgebreide systeem software. Toen echter de eerste succes-golven voorbij waren was het de beurt aan de gewone man, die niet op specificaties maar presentatie kocht. In de roerige wereld van spectaculaire spelletjes computers met kleur en geluid verloor de nog altijd intelligentere NewBrain snel terrein. De door professionals verwachte randapparatuur liet langer op zich wachten dan was aangekondigd en de verkoop klapte volledig in elkaar. 1 augustus 1983 was het vonnis getekend en de briljante wonder-baby was niet meer. Een schokgolf ging door gebruikerskringen toen de pers op het debacle sprong. Ook bij Tradecom stond de telefoon niet stil en was de meest gehoorde vraag "wat moet ik nu met mijn NewBrain?". Het geloof in het product en de kracht ervan bracht Tradecom er toe de mogelijkheden van een reddings-operatie te onderzoeken. Deze operatie was niet eenvoudig doordat Grundy Business Systems zich ongelooflijk diep in de problemen had gewerkt zonder tijtjds maatregelen te nemen waardoor een gigantische voorraad was ontstaan bij de fabrikant die voor Grundy de NewBrains produceerde. Om uitverkoop tegen dumprijzen te voorkomen, waardoor de NewBrain nooit meer terug zou kunnen komen, moest Tradecom enorme verplichtingen aangaan. Ook de buitenlandse distributeurs zagen hun verkoop volledig stagneren en bleven met voorraden zitten. Enkelen volgden al spoedig het voorbeeld van Grundy. Door versneld op grote schaal productie van de al zo lang verwachte uitbreidingen te starten herwon Tradecom tegen alle negatieve publiciteit in het vertrouwen van NewBrain eigenaars en de distributeurs. Door een duidelijke marketing filosofie en optimale support aan de buitenlandse distributeurs kwam de verkoop al spoedig weer op gang en werd de voorraad snel afgebouwd. Met belangstelling keek de wereld naar Tradecom en werd al binnen twee maanden een aantal nieuwe ontwikkelingen verwacht uit de research afdeling. Dat dit iets meer tijd zou vergen zal duidelijk zijn. Tradecom wil echter het herwonnen vertrouwen niet verliezen.

De NewBrain gebruiker van vandaag zijn overal te vinden. Op scholen, universiteiten, fabrieken en in ziekenhuizen verricht de NewBrain zeer specifieke functies. Ontwikkelingen op hard- en software gebied geven de gebruikers een breed scala aan gebruikersmogelijkheden.

Beschikbaar zijn op het ogenblik:

Pascal, Viditel, Comal, Basicode Forth, Elan, Lisp, Animator etc.

In de hardware ontwikkeling ontstond Masternet, de nieuwe expansion, harddisk controller, multi-user netwerk, multiple choice systeem, communicatie pakket en diverse I/O apparaten zoals A/D-D/A converters en parallel drivers.

Export van NewBrain producten neemt nog steeds de grootste plaats in. Mede door de uitgebreide karakterset is de buitenlandse markt erg gelukkig met de NewBrain.

Libanon is als nieuwe distributeur in de lijst opgenomen. De NewBrain wordt daar als arabisch van rechts naar links schrijvende computer verkocht.

De laatste ontwikkeling is een joined-venture met India, alwaar men zich gaat bezighouden met assemblage van NewBrains voor de eigen markt en productie van kleine serie's nieuwe peripherals. De verwachtingen voor de toekomst zijn optimistisch gezien het grote aantal aanvragen uit alle delen van de wereld.

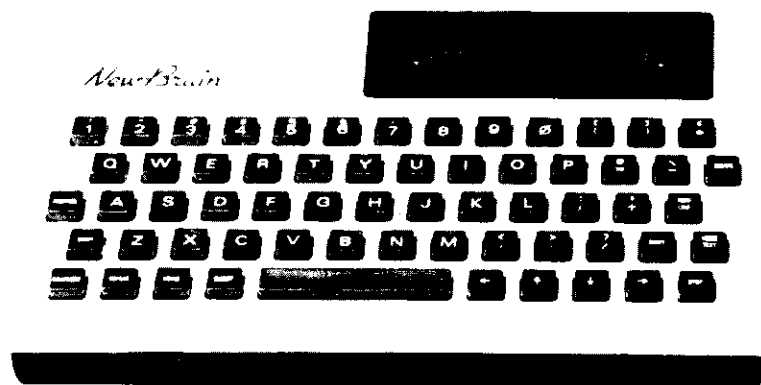
Een computer is niet alleen een stuk hardware met technische specificaties. Wat je ermee kunt doen is veel belangrijker, want daar is het uiteindelijk voor gemaakt.

J.O. Rooymans

Product Manager NewBrain.

NewBrain

11



PROFESSIONELE MICRO-COMPUTER IN KLASSIKAAL NETWERK, NU BEREIKBAAR MET EEN KLEIN BUDGET!

Deze micro-computer met Z 80A processor is al leverbaar voor **f 1190,-**
Een geavanceerde netwerk module (f 2500,-) integreert 16 New Brain's tot een volledige informatica klas met alle onderwijs faciliteiten.

Enkele technische gegevens die het weten waard zijn:

De NEW BRAIN micro-computer:

- ☐ is uitgerust met een zgn. ANSI BASIC compiler; een standaard programmeertaal;
- ☐ heeft een volledig geïntegreerd schrijfmachine-toetsenbord van normale afmetingen;
- ☐ hij is standaard uitgerust met een 32k RAM geheugen dat echter met inplugbare modules is op te voeren tot twee megabytes;
- ☐ hij beschikt over aansluitmogelijkheden voor twee cassetterecorders, TV, monitor, printer en modem;
- ☐ biedt de mogelijkheid om andere software-pakketten te laden vanaf cassette of onder CP/M vanaf schijf;
- ☐ beschikt voorts over 80 tekens maal 30 regels video;
- ☐ eveneens biedt hij de mogelijkheid voor grafische toepassingen (zgn. high resolution graphics);
- ☐ alle interfaces en uitgangen zijn standaard;
- ☐ bescheiden afmetingen: 28 x 15 x 5 cm;
- ☐ is ook te leveren met nikkel-cadmium batterijen voor automatische beveiliging tegen kortdurende stroomuitval!

Nadere inlichtingen worden u desgewenst graag verstrekt door:

Prijzen excl. BTW



Tradecom International B.V.
Hondsdijk 3,
Postbus 60,
2396 ZH Koudekerk a.d. Rijn
Tel. 01714-43 00
Telex 39466

Engelse NewBrain Users Group

Sinds enkele maanden ben ik ook lid van de Engelse NewBrain Users Group (NBUG). Deze gebruikersgroep geeft ongeveer tweemaandelijks een blaadje uit met allerlei welenswaardigheden voor de echte NewBrainers.

Al in het eerste blad bleek dat dit een non-commerciële groep was;

Van alle kopiën met extra info die was bijgevoegd waren de onder-, boven- en zijranden afgeknipt om porto kosten te sparen. Het eerste blad behandelt voornamelijk de historie van de NewBrain en met name de blunders in de afd. Marketing die al twee keer (hijna) voor de ondegang van NewBrain'tje hebben gezorgd. De twee volgende bladen waren algemener, terwijl ook de uitvoering en layout aanmerkelijk beter werden.

De hele adm./ledenwerving/redactie etc. wordt gedaan door 1 persoon; te weten Gerald McMullon, welke voorheen bij Grundy werkzaam was. Het blad is een aanrader (zeker voor de prijs van 6 pond/jaar).

Adres : GFG Microsystems
36 Anaitage Way
Cambridge CB4 2UE
England

Ton Tonies
Grotestraat 21
5368 AJ Haren (Gem. Megen) NBr

LET OP: Nieuwe penningmeester; nieuw gironummer zie pagina 3

Kopij voor OnLine 84-4 voorlopig opsturen naar Bert Hovius

THE GANG OF FOUR: UNITED NEWBRAIN USER GROUPS

At present there are four NewBrain user groups in different countries coöperating in the following manner:

- 1) exchange of Public Domain Software (meaning software which is not subject to copyright);
- 2) exchange of technical information;
- 3) exchange of articles for publication;
- 4) wherever possible: personal contacts and consultations.

The availability of Public Domain Software and of membership lists (in order to facilitate contacts between members) are important aspects of these groups. Coöperation in the field of hardware development (e.g. a cheap modem) will be stimulated and members who are active in this field are invited to send me their addresses.

Public Domain CP/M Software is available in the Netherlands through our contact with the very active Netherlands CP/M u.g. which also takes part in the Hobby Computer Club.

Denmark has not less than five NewBrain user groups, the Copenhagen user group acting as the coordinator.

France: through a letter from one of its members we know of the existence of a French NewBrain user group, but we do not know its address yet, neither anything else.

UK: several other user groups exist in the UK, the most important being NBUG (Gerald F.G. McMullon, GFG Microsystems, 36 Armitage Way, Cambridge CB4 2UE). Gerald is a ex Grundy software engineer. He publishes a newsletter with much information on Grundy and less (but well informed) on the NewBrain. They have no PDS and do not publish membership lists. The lists of GFG software in the newsletter give it a somewhat commercial appearance. GFG produce a ROM box and will blow EPROMS with your own programmes. (Not free).

Another UK group is INGROUP, run by Anthony Hodge. The newsletter is not quite legible (microscopic print) and very uninformative. Not recommended.

We know nothing of the Brighton NewBrain user group. It is said to have been founded by disappointed INGROUP members. We sent for information, but our letter has not been answered.

- 1) UK: OPEN STREAM, CH Philip Crookes, 26 College Road, BROMLEY BR1 3PE KENT. T 01-290-5262. Secr.: Brian Miller. 24 College Road, BROMLEY BR1 3PE, KENT. T 01-290-5299 à PDS(C,D, CPM), NL2, MLX, BBX, REPX, M40. AMF £10.00
- 2) NETHERLANDS: NEWBRAIN GEBRUIKERSGROEP, CH Bert Hovius, Bitterstraat 58, 8011 XN ZWOLLE, T 038-228252; SE Robert van Albada, Talmestraat 20", 1073 JX AMSTERDAM, T 020-739671. TR Jan de Vries, Haagstraat 16, 3581 SX UTRECHT, T 030-511243; à PDS(C,D,CPM), NL1 (pages in HCC Nieuwsbrief), ML, M460. Takes part in Hobby Computer Club (M1B,000). AMF Dfl.45.00 (HCC).

- 3) ITALY: NEWBRAIN USER GROUP, CH Giovanni Mello, Via Corne 1/A, 31030 BIGOLINO (Tv). à PDS (C)
- 4) DENMARK: COPENHAGEN NEWBRAIN USER GROUP, CH Bo Grønland, Sprydet 59, V-3070 SNEKKERSTEN, T 02-225181; SE Niels Troels-Smith, Brudevanget 19, 3520 FARUM; TR Erik Pürschel, Løvenborg Alle 6, 2820 GENTOFTE. @PDS(C,D,CPM), NL(Brains), MLX. Address Brains: V.B. Lauridsen, Tagensvej 89 4th, 2200 KØBENHAVN N. Fee Dkr. 75 for 5 issues. Postgiro No. 2071460. ROM listings at cost-price.

Abbreviations:

PDS = Public Domain Software (C: cassette, D: disk, CPM:CP/M)
 NLn = Newsletter every n months
 ML = Membership lists
 BB = Bulletin Board (via modem)
 REP = repair service
 X = forthcoming or under development
 CH = chairman
 SE = secretary
 TR = treasurer

Vraag en aanbod.

Wie heeft er op de unexpanded NewBrain al een spreadsheet programma, zoals Procalc, VUcalc, Braincalc of zelf iets gemaakt? Ik ben zeer geïnteresseerd in de ervaringen hiermee.

Wim Luijt, Troelstrapad 22 4463 VM Goes tel.
01100-28197 na 20 h.

Zijn er nog meer mensen die belangstelling hebben voor een digitale bandrecorder voor de NewBrain? Ik denk hierbij aan wat beschreven staat in de HCC Nieuwsbrief nr. 59 voor de Edixy Sorcerer, de Hobbit voor de BBC of de Memotech voor de DAI. Ik heb niet de kennis om zoiets te ontwikkelen maar ben wel bereid een dergelijk project actief te steunen.

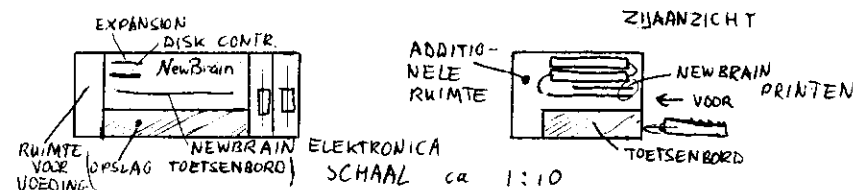
W.M.Luijt Troelstrapad 22 4463 VM Goes tel.
01100-28197 na 20 h.

Werkgroep hardware

Hoe is nu globaal de stand van zaken?

- facelift

Hiermee zijn we nog niet concreet verder gekomen dan de volgende punten: Er moet een kunststof kast komen voor het geheel, omdat dat vriendelijker is en goedkoper dan metaal. Bovendien is het gemakkelijker te verwerken. Er moet ruimte zijn voor alle uitbreidingen. Om de kast niet te hoog te laten worden is bedacht om de kast wat in de breedte te laten groeien. Het heeft geleid tot de volgende ontwerpschets:



Er is op een elegante wijze plaats gegeven aan de diverse onderdelen. Het is uit oogpunt van montage, flexibiliteit en betrouwbaarheid min of meer vereist dat de computer zelf ook gesloopt wordt, zodat de hoofdprinten een vaste plaats in de kast krijgen. Met een kabeltje en misschien een beetje interface elektronica wordt het toetsenbord met display verbonden met de hoofdkast.

Gezien het werk dat met realisatie gemoeid is, wordt het volgende voorgesteld:

De definitieve ontwerpen (maatschetsen, details en eventuele schema's) worden gepubliceerd. De kunststof kast kan bij voldoende belangstelling geleverd worden door mijn bedrijf, al dan niet inclusief montage van alle onderdelen voor gebruikers die dat willen. Ik kan dit uitsluitend tegen een redelijke prijs verzorgen als er voldoende belangstelling is (mal voor x kunststof kasten ca. f 1500,-, tegen kostprijs kast zelf ca. f 30,- kaal).

Vrijblijvend kunt u schriftelijk laten blijken of u geïnteresseerd bent, en of u zelf wilt/kunt monteren van uw nieuwe geheel of alleen bepaalde gedeeltes. Dit laatste is erg belangrijk, want het demonteren van een computer en dan het inbouwen in een grote kast kan vrij veel tijd kosten. Aan de hand van de verkregen gegevens kunnen we in het bestuur bekijken wat haalbaar is.

- functioneren werkgroep

Bij de laatste vergadering van de kerngroep van de NewBrain gg is geconstateerd dat werkgroepen in het algemeen onvoldoende als zodanig functioneren (niet meer dan 1 man in werkgroep enz.). Wel, dit geldt ook voor de werkgroep hardware. In plaats van werkgroep, zal het meer vraagbaak gaan heten. Dus alles blijft wat betreft hardware zoals het was, alleen wordt zo de situatie reeler weergegeven.

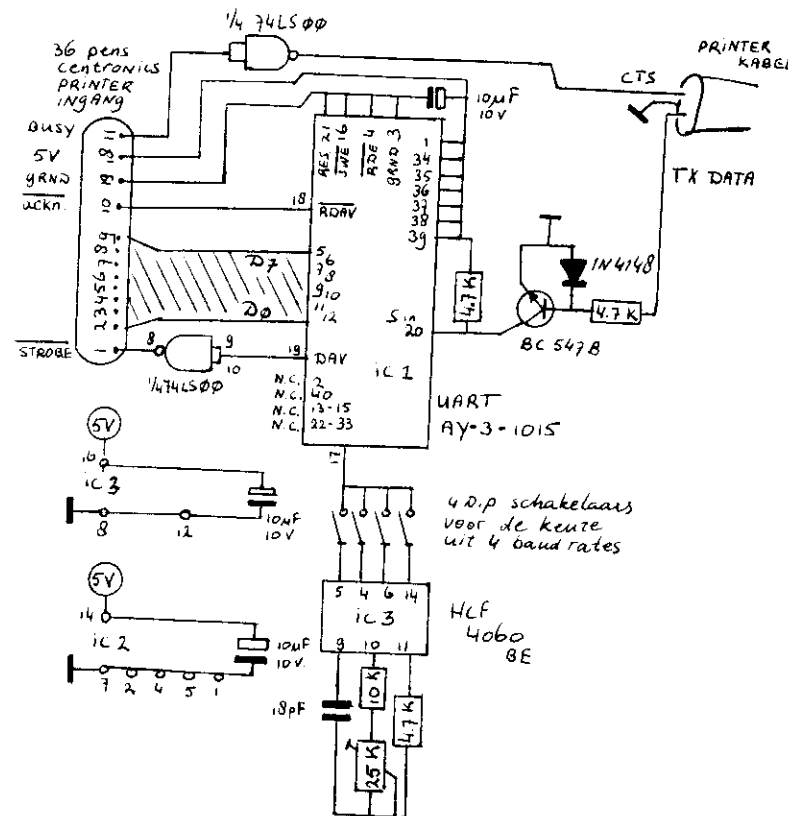
Rob Maris

Om een Brother 8300 printer (met een centronics ingang) aan te sluiten op de NewBrain heb ik bijgaande schakeling ontworpen. Ik heb die interface meer dan een jaar in gebruik en ondanks de vrij eenvoudige opzet bleek hij in praktijk zeer betrouwbaar. Met enige handigheid is een printje zelfs in het achtercompartiment van de 8300 te bouwen. De Brother blijft ten alle tijde als schrijfmachine bruikbaar. De kosten bedragen ca. f35.- (met een UART van Tandy) tot f55.- . Zonder 36-polige connector.

De voeding (+5V) wordt betrokken van de Brother. Dit heeft als nadeel dat het CTS-signaal niet op RS232-nivo gebracht kan worden. Dit bleek voor de NewBrain (die is uitgerust met de bekende 1488 en 1489 drivers) geen bezwaar. De maximale baudrate is bij aanpassing van C4 naar 12 à 15 pF 4800 baud - in ieder geval als de aangegeven onderdelen worden gebruikt. Voor degenen met een frekwentieteller om de baudrate met de potmeter in te kunnen stellen zij vermeld dat de frekwentie op pen 17 van de UART 16 maal de baudrate moet bedragen. Wees er bovendien van overtuigd dat de UART AY-3-1015 (of equivalent) wordt genomen en NIET de AY-5-1013. Deze laatste heeft naast +5 ook -12V nodig.

De pennen 35 t/m 39 zijn ingesteld voor de NewBrain
(1 start-, 8 data-, geen pariteit- en twee stopbits).

Wat betreft de joystick-interface zoals die op 26 mei in Utrecht was te zien, wil ik graag resultaten van mensen die deze hebben nagebouwd en hebben geprobeerd die te gebruiken met een newbrain + expansion-box. Op zo'n systeem heb ik hem niet kunnen testen. Voor mensen die alsnog een schema van de interface en een listing van de bijbehorende routine willen, heb ik nog een 20-tal stencils liggen. Deze mensen kunnen, met een aan zichzelf gerichte enveloppe (gefrankeerd met f0.70 aan zegels) tezamen met een extra postzegel van f0.50, zich een exemplaar laten doen toekomen. A.u.b geen telefonische reacties of vragen; schrijft u naar: H. v.d. Linden
Hoofdstraat 88
5171 DG Kaatsheuvel



PS

Met boodschappen die voor meerdere gebruikers geschikt zijn kunt het beste bij online-4 terecht.

Voor mensen die bij nabouw zes pennen over hebben: TTL ic's in casu 2 x 74ls30 en 74ls32 moeten ook gevoed worden (pen 7 is 0 volt, pen 14 is +5 volt). De ingangen van de 74ls30 zijn 1,2,3,4,5,6,11,12. Pen 1 is die met dat puntje in het huis. Helaas blijkt deze additionele informatie niet overbodig. Mocht u met de bouw nog niet uitkomen raadpleegt u dan een handige kennis.

De kosten komen op ong. f45.- in het geval dat u het mooi afwerkt , en zorgvuldig te werk gaat; het kan f15.- minder en een z80 meer kosten.

RESET DE NEWBRAIN MET BEHOUD VAN PROGRAMMA

Een reset voor de NewBrain is mogelijk m.b.v. het N.M.I.-signaal van de Z80 (Non Maskable Interrupt). Dit is een geïnverteerd signaal. Bij normaal gebruik is het signaal hoog (Pull Up). Maken we het signaal laag dan wordt het door de Z80 gedetecteerd en die geeft dan direct een interrupt, wat wil zeggen dat hij direct stopt waarmee hij op dat moment mee bezig is. (NMI is pin 43 van de Expansion poort) Intern van de NewBrain wordt NMI niet gebruikt, maar wordt NMI laag dan voert hij direct de instructie uit die gevormd wordt door de drie bytes op de locaties 102, 103, 104 (decimaal). E.e.a. is als volgt te realiseren.

poke	102,24	}	JR 119
poke	103,15		
poke	119,49	}	LD SP,616
poke	120,104		
poke	121,2	}	JP 494ZZ
poke	122,195		
poke	123,14	}	
poke	124,193		

De reset zal in de meeste gevallen een uitkomst bieden. Wordt echter door een echte "crash" de inhoud van page zero verminkt dan zal het uitschakelen van de voeding als enige uitkomst bieden.

Tip : Zelf heb ik deze reset intern aan de "vidiotext"-toets gekoppeld zodat geen extra drukknop nodig was.

A. van der Schrier

(advertentie)

RESET - SCHAKELAAR

Het is gelukkig mogelijk gebleken een reset-schakeling te ontwerpen voor alle versies van de New-Brain. De schakelaar kan ingebouwd worden aan de zijkant van het toetsenbord en geeft zowel een hard-ware als een soft-ware reset. Dit houdt in, dat bij een storing of 'op hol slaan' van de computer het indrukken van de schakelaar hetzelfde effect sorteert als het uit- en inschakelen van de netspanning. De floppies kunnen echter blijven zitten en zoals gebruikelijk wordt het RAM-geheugen gewist. Informatie over de inbouw kan ingewonnen worden bij de firma Telcom Engineering te Bussum, tel. 02159-17306.

advertentie

MARIS ELEKTRONICA



Nu verkrijgbaar: 6 polige stekertjes voor de NewBrain!

Voor voeding, printer, coms en cassette, incl. soldeeraanwijzingen.

Als de voeding doorgelust wordt via expansions zal de machine betrouwbaarder werken.

plastic oshulling f 1,70

vergulde contacten, per contact f 0,50



Screendump programma's voor high resolution afdrukken.

Voor de volgende printers:

IDS Micropriss printers

Star printers

Epson printers

MPI Printmate 99

Seikosha CP250

Trend JP80

NEC 8023

f 18,- per exemplaar

SRCCD CP/M assembler editor. Met onovertroffen eigenschappen....

- ca. 1000 regels source code (32K NewBrain)

- gedrag als zeer lang scherm! (overzichtelijk)

- autoformat en autocompressing

- high speed load, save, scroll op en neer, find label etc.

- page buffer

f 64,- incl. support

SRCCUT source code cutter. Analyseert Z80 of 6502 source code door het in programma'secties te snijden en tot slot een overzichtelijke structuur te laten zien met subroutine niveau's, zodat het onderhoud of wijzigen van een assemblerprogramma sensationeel sneller kan geschieden. Men ziet direct wat voor consequenties een verandering heeft. Zien waar variabelen voorkomen? Met SRCCUT geen probleem!

Tevens is SRCCUT een unieke documentatiehulp. Doordat het een BASIC programma is, is de prijs al even uniek: van f 40,- tot f 80,-, afhankelijk van gewenste support.

Arnhemseweg 27-B, 7331 BB Apeldoorn.

055 - 424485

bestellingen alleen vooruitbetaling (postrekening 5639839) of onder rembours. prijzen incl. BTW, excl. verz. kosten en magnetische drager.

Ingeschreven bij de Kamer van Koophandel te Zutphen onder nr. 40731. Kantooruren maandag t/m vrijdag 8.30 tot 16.30 u., bezoek uitsluitend woensdag t/m vrijdag of volgens afspraak.

STANDAARD CONNECTOREN

Aangezien er steeds meer mensen er toe overgaan, hun NEWBRAIN om te bouwen in een kast gezamenlijk met controller, drive's, e.d. leek het mij verstandig om een standaard voor te stellen, voor de connectoren op die nieuwe kast, om een chaos te voorkomen, en dat de apparatuur evt. uitwisselbaar/aansluitbaar wordt/blijft.

Er wordt gebruik gemaakt van de bekende "DIN-stekers"
Die overal makkelijk en goedkoop verkrijgbaar zijn.

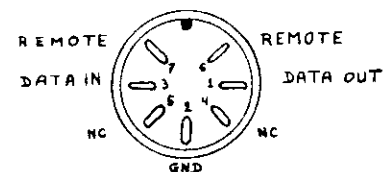
TAPE 1&2	pen	signaal
	1	DATA OUT (n. r.a.)
	2	GROUND
	3	DATA IN
	4/5	N.C.
PRINTER	6/7	Remote sturing recorder
	1	DATA
	2	GROUND
	3/5	N.C.
	4	CTS: (Busy)
COMMS	1	DATA OUT
	2	GROUND
	3	DATA IN
	4	CTS
	5	RTS

Indien men nu op de printer een dergelijke connector monteerd kan men een gewone 'diode-kabel' als printerkabel gebruiken: Prijs? naar ik een c.a. f7,50 voor een complete kabel, bij zelfbouw: f1,50-f2,00 voor een plug, en de kabel c.a. f1,50/m. Dat is nog eens wat anders dan f50,00-f75,00 voor een kabeltje. Datzelfde geldt natuurlijk ook voor de comms- en recorderkabel.

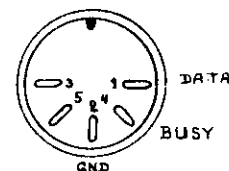
E. Drijver
Aldebaranstraat 49
9742 LH GRONINGEN
050 - 778415

DE NIEUWE NEWBRAIN STANDAARD CONNECTOREN

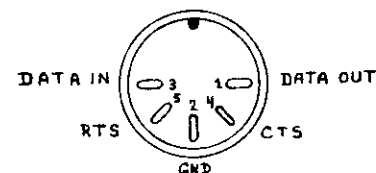
TAPE 1&2



PRINTER



COMMS



STORINGEN bij gebruik van diskdrive.

Hetzelfde probleem als Rob van Albada had, (NB.On Line-84-2) nl. "NO RESPONSE FROM DISK CONTROLLER", heb ik ook gehad, en ook ik heb in het begin toen de storing zich aanmeldde, enig effect gescoord met wat 'knoeien' aan de datakabel, maar na wat speurwerk hoorde ik dat op pen 6 van IC 449 van de controller een frequentie moet staan van 1 MHz, wat in mijn geval slechts c.a. 700kHz bleek te zijn.

Het blijkt nl. vaker voor te komen, dat de controller te laag staat afgeregeld.

Deze frequentie is af te regelen met de instelpot aan de smalle zijde van de print.

Om deze frequentie af te regelen sluit U een frequentieteller aan op pen 6 van IC 449, en draait langzaam aan de instelpot tot de teller een frequentie aangeeft van 1 MHz.

Dat is alles!

Daarna zult U ook geen verschil meer merken wanneer U aan de DATA-kabel 'knoeit'.

Mocht U niet in het bezit zijn van een frequentie-teller, maar wel een oscilloskoop ter beschikking hebben, dan kunt U die ook goed gebruiken, gewoon afregelen tot de periodetijd 1µs bedraagt. Indien U problemen heeft, en die afregeling niet zelf kunt verrichten en in het noorden woont, of er een reisje voor over heeft mag U mij bellen voor een afspraak, kan ik die afregeling voor U verrichten.

Deze controle/afregeling 'kost' U natuurlijk 'gratis'.

Evert Drijver
Aldebaranstraat 49
9742 LH Groningen
050-778415

DE STORINGS GEVOELIGHEID VAN DE E.I.M

Door storingen van buitenaf via het lichtnet, kan de E.I.M. van slag raken waardoor opnieuw opstarten noodzakelijk wordt, vooral bij in- en uitschakelen van apparatuur zoals koelkasten, printers, etc.

Deze storingen komen door op de voedingsspanning van de IC's, nu is gebleken dat met name de busdrivers in de E.I.M. hiervoor het gevoeligst zijn.

Zo'n busdriver (74LS245) kan door een storing, in combinatie met de toch altijd al aanwezige "ruis" op de voedingsspanning, een foutieve waarde doorgeven, hetgeen "tilt" tot gevolg heeft.

Deze gevoeligheid kan aanzienlijk verminderd worden door direkt over de aansluitpinnen 10 en 20 van deze I.C.'s een keramische condensator te plaatsen van 0,1 F 50 V. (pin 10=0 V. ,pin 20=5 V.).

Storings ongevoelig wordt hij hierdoor niet, maar het wordt in ieder geval wel een stuk minder.

A. van der Schrier

(Voor verdere informatie Rob Maris, Hardware)

Voor hobby en voor bedrijfsmatig gebruik.

Computersystemen in alle soorten en maten 8 - 16 en 32 Bits en randapparatuur.

Beeldschermmonitoren Fabr. SANYO

Type	2112 CX groen	Fl.	395,=
	2212 CX amber	Fl.	405,=
	8112 CX groen	Fl.	695,=
	8212 CX amber	Fl.	705,=

Beeldschermmonitoren Fabr. ZENITH

Type	ZVM-123-E groen	Fl.	369,=
	ZVM-122-E amber	Fl.	389,=

Alle monitoren worden geleverd inclusief video kabel.

Printers Fabr. STAR

Type	GEMINI 10X 120 T/sec.	96 kol	Fl.	1250,=
	GEMINI 15X 120 T/sec.	136 kol	Fl.	1810,=
	DELTA 10 160 T/sec.	96 kol	Fl.	1920,=
	DELTA 15 160 T/sec.	136 kol	Fl.	2630,=
	RADIX 10 200 T/sec.	96 kol	Fl.	2850,=
	RADIX 15 200 T/sec.	136 kol	Fl.	3450,=
	STX-80 Thermische printer		Fl.	695,=
	POWER TYPE Daisywheel printer		Fl.	1750,=

Alle printers worden geleverd inclusief GENTRONICS kabel.

Papier blanko 12 " X 240 mm		
doos met 2000 vel (afgehaald)	Fl.	39,60

Diskettes Fabr. FLEXETTE

48 tpi SS	Fl.	99,50
48 tpi DS	Fl.	132,=
96 tpi SS	Fl.	157,50
96 tpi DS	Fl.	175,=

Diskettes Fabr. GEHA idem aan FLEXETTE diskettes.

NIEUW OP DISKETTE GEBIED.

Gekleurde diskettes te verkrijgen in 12 kleuren		
48 tpi DS	Fl.	158,=
96 tpi DS	Fl.	189,=

Alle prijzen voor diskettes per 10 stuks, inclusief luxe opbergbox.

Alle prijzen zijn exclusief 19 % B.T.W.

Voor meer informatie bel of schrijf naar :

ATEBO

Adviesburo voor automatisering

Hofstraat 14
Postbus 91
7120 AB Aalten
tel. 05437-5341

PEN(7), PEN(8), PEN(9)

Naar aanleiding van de oproep van Sietze Knijpstra in Online 84-2, nemen we hier een stukje over uit het blad van NBUG, de Engelse gebruikersgroep.

De vroegste versies van de grafische ROM kennen deze functies niet. De volgende routine geeft het startadres (st), het eindadres (en) van de grafische stream (sr). De routine berekent ook de breedte van het scherm in pixels. Deel dit door 8 om de breedte in bytes te krijgen.

```
100 sr=i1:sa=0:REM verander sr in het nummer van de
    betreffende grafische stream
110 DEF FNa(n)=PEEK(n)+256*PEEK(n+1)
120 t1=FNa(86):t2=FNa(100):REM begin en einde van
    de stream table
130 FOR i=t1 to t2 STEP 6
140 IF sr=PEEK(i) THEN sa=FNa(i+4):REM vind plaats
    in de stream table
150 NEXT i
160 IF sa=0 THEN PRINT "FOUT: geen startadres voor
    stream":sr:"in stream table":END
170 st=FNa(sa+42):en=FNa(sa+44):wd=FNa(sa+26)
180 en=en-1-(wd/8)
190 PRINT "BEGIN, EINDE, BREEDTE",st,en;wd
200 END
```

Het gebruik van het geheugen lijkt op dat van een normaal tekstscherm. De eerste wd/8 bytes worden gebruikt voor de bovenste beeldlijn, de volgende wd/8 voor de tweede etc. Een grafisch scherm hoeft niet geïnitieerd te worden; evenmin wordt er aan het einde van een beeldlijn extra geheugen gereserveerd, zoals bij een tekstscherm.

Het bitpatroon van elk byte is zo dat het minst significante bit het meest linkse pixel bestuurt van een groepje van acht pixels. De bytes worden dus als het ware in spiegelbeeld afgebeeld.

Je kunt met een en ander experimenteren door allerlei waarden naar adres st te poken (POKE st,x). Het resultaat wordt zichtbaar in de linkerbovenhoek van het scherm.

GEHEUGEN TE KLEIN ?

Als tijdens de uitvoering van een programma blijkt dat het geheugen net te klein is, dan kan je op de volgende wijzen geheugen ruimte besparen.

A Programma zo klein mogelijk houden, zonder opdrachten weg te laten:

- laat overbodige spaties in de programma regels weg; dus `PUT31 i.p.v. PUT 31`

- zet tussen het regelnummer en de eerste opdracht geen spatie; dus `10PUT31 i.p.v. 10 PUT31`. Met LIST verschijnt de spatie weer, maar hij staat echt niet in de source table (de plaats waar het programma is opgeslagen).

- haal spaties achter de regel weg met `Sh/→`. Ze zijn onzichtbaar, dus controle heb je er niet op.

- wanneer de 0^e elementen van een array niet gebruikt worden neem dan `OPTION BASE 1` op. Bij 2-dimensionele arrays kan dat flink wat schelen.

- spaties in strings vervangen door `TAB( )` of , (komma). `TAB` kost 4 of 5 bytes, een komma slechts één; dus `"12345",,"12345" i.p.v.`

`"12345 12345".`

- opeenvolgende regels met print opdrachten vervangen met , of ; tussen de te printen variabelen/strings; dus `PRINTA$,B$,C$ i.p.v. PRINTA$:PRINTB$:PRINTC$`. Is de string te lang om op de regel te passen, dan schuift hij automatisch op naar de volgende regel, wat het aantal komma's beperkt.

(zie ook de quick brown fox in het vorige nummer).

- afkorte van de keywords is zinloos, omdat ze toch als één byte token in de source code table komen.

- arrays `CLEAR`en wanneer de inhoud niet meer nodig is.

B Het console verkleinen (in NewBrain termen: stream 0 kleiner maken dan default)

Als er geen communicatie met het scherm nodig is tijdens de uitvoering van een gedeelte van een programma verklein dan in het programma stream 0 met `OPEN # 0,3,"16" (model AD)` of `OPEN # 0,0,"1" (model A en AD)`. Het eerste voorbeeld levert een geheugen winst van 1,5 kbyte op ! Is communicatie weer noodzakelijk, maak dan ruimte door bijv. arrays te `CLEAR`en en `OPEN #0,0`.

Zelf heb ik ook geprobeerd om de printer als console te laten fungeren. Het kan wel maar is niet aan te bevelen. De foutmelding `ERROR 109` werd keurig afgedrukt en werd steeds herhaald. Verder is het een gok wat gebeurt. Ik heb al helemaal overnieuw moeten beginnen door de stroom uit te schakelen. Een andere keer luste het me om te stoppen met de `STOP`-toets waarna `ERROR 0` op het default scherm (`# 0,0,"24"`) verscheen en het programma zelfs nog aanwezig was.

C Programma-deien `DELETE`n.

Is een bepaald gedeelte van een programma uitgevoerd en wordt het niet meer gebruikt `DELETE` dit gedeelte. Een nadeel is dat het programma voor iedere `RUN` op-nieuw moet worden geladen.

D Uitwendig geheugen gebruiken.

Bij het op alfabet zetten van mijn bibliotheek, meer dan 1000 titels, kan ik met een klein sorteer programma 1000 (!) auteursnamen (1 veld per record) tegelijk op alfabet zetten. De rest van de titelrecord was op band opgeslagen en kon steeds stapsgewijs worden ingevoerd, verwerkt en op band worden gezet.

Enkel-toets of Graphics-toets commando's op de New Brain

Tijdens de laatste NB GG dag is duidelijk gebleken dat de enkel-toets commandos van de NB bij vele gebruikers niet bekend zijn. Deze commandos kunnen soms toch handig zijn. Een voorbeeldje: je kan een programma oproepen op de "oude" manier dus door LOAD "PIMS", maar je kunt dit ook als volgt doen GRAPHICS/U "PIMS". Scheelt toch een beetje. De meeste van deze enkel of graphics-toets commandos hebben twee functies. Functie 1 = Puur als commando (NEW, LIST, RUN, LOAD, enz) Functie 2 = Gebruik in expresies (INT, <>, >=, PI, enz).

Een overzicht, van de mij bekende, enkel-toets commando's:

GRAPHICS/	FUNCTIE 1	FUNCTIE 2
A	LIST	+
B	NEW	-
C	PRINT	*
D	RUN	/
E	GOTO	^
F	GO TO	&
G	GOSUB	AND
H	GO SUB	OR
I	IF	<>
J	INPUT	<=
K	FOR	>=
L	NEXT	=
M	RETURN	>
N	REM	<
O	END	RND
P	STOP	PI
Q	DIM	POS
R	ON	ABS
S	OPTION	ATN
T	SAVE	COS
U	LOAD	EXP
V	RANDOMIZE	INT
W	OPEN	LOG
X	CLOSE	SGN
Y	? (print)	SIN
Z	RET	SGR
(	RESTORE	TAN
)	DATA	FALSE
=	LET	NOT
-	READ	TRUE
+	POKE	ASC

Dit zijn alle mij tot nu toe bekende commando's. Wat opvalt is dat de graphics plot commando's ontbreken (zijn die er zo ja waar ??). De in de tabel opgenomen commando's zijn heb ik als volgt gevonden: Typ een regelnummer in gevolgd door twee maal het enkel-toets commando list hierna de regel !

Ivan van kampen

Auto-nummering en 1-toets commando's.

=====

Met behulp van onderstaand programma kun je een automatische regelnummering en 1-toets commando's in je programma's inbouwen. De nummering begint bij regel 10 (zie 60000) en hoogt zich steeds op met 10 (zie 60030).

De meest voorkomende Basic-statements kun je nu met behulp van 1 toets invoeren. Je doet dit door het intoetsen van het gewenste commando en de Shift-toets (Sh).

vb. Sh-P = print

Sh-R = return

Sh-G = gosub

Je kunt dit programma natuurlijk zelf nog verder uitbreiden.

```

60000 PUT6,31,10,10:CLOSE#6:OPEN#6,6:a=10
60010 ?a;
60020 GET#6,a$:IFa$=CHR$(0)THEN 60020
60030 IFa$=CHR$(13)THEN PUT5:LINPUT#0,a$:PUT12,10,30:?a$:PUT
30,27,4,3:MERGE#0:a=a+10:GOTO 60010
60040 IF a$="P"THEN a$="print"
60050 IF a$="N"THEN a$="next"
60060 IF a$="G"THEN a$="gosub"
60070 IF a$="O"THEN a$="goto"
60080 IF a$="R"THEN a$="return"
60090 IF a$="F"THEN a$="for"
60100 IF a$="M"THEN a$="mid$("
60110 IF a$="I"THEN a$="input"
60120 IF a$="C"THEN a$="chr$("
60130 IF a$="T"THEN a$="then"
60140 FOR b=1 TO LEN(a$):PUT ASC(MID$(a$,b,1)):NEXT b
60150 GOTO 60020

```

Jos Hermans

Rekenen tijdens een programma

Onderstaand programma laat zien hoe een programma onderbroken kan worden om de onderste regel van het scherm als een eenvoudige rekenmachine te gebruiken. Na 'r' getypt te hebben kan men b.v. invoeren: '23.6x345='.
 Voor het oorspronkelijke programma ('Oefenen chemische berekeningen') was alleen x en : nodig maar uitbreiding met + en - bewerking is niet zo moeilijk.
 Het afgedrukte hoofdprogramma is niet zo zinvol; het gaat in feite om de subroutines 500, 600 en 700.

Een korte toelichting:

- De eerste 6 regels zijn in feite overbodig, maar wel handig bij het ontwikkelen van een wat groter programma dat vaak tussentijds gelist of op disc gesaved moet worden. STOP toets of goto 2 geeft een groot scherm om op te listen. Om het programma, (inclusief datum in korte notatie, b.v. 0309), snel te saven kan men de opdracht goto 4 gebruiken. Ook de gebruikte geheugenruimte wordt afgedrukt.
- In de initialisatie-subroutine wordt in regel 20010 een functie gedefinieerd die de cursor (in de 'off' toestand) naar een bepaalde regel stuurt en tegelijkertijd die regel wist als hij al beschreven was. (Handig bij foute invoer van de gebruiker).
- Als er onder en boven de cursor tekst kan zijn, is de cursor van de NewBrain wat te bescheiden om op te vallen. Dat is in regel 20020 ondervangen door cr\$. Op de plaats van de cursor is dan afwisselend een streepje en een blokje te zien.

- In het programma worden veel Boolese variabelen gebruikt (deze kunnen dus TRUE of FALSE zijn):
 tk: x of : teken gebruikt?
 vm: vermenigvuldiging? xl: eerste keer r ingevoerd?
 dl: deling? r: rekenmachine gevraagd?
 is: = teken? cr: NEW LINE?
 Verder is in\$ de invoer, g1 en g2 de twee getallen en aw het juiste antwoord.

```

1 ON BREAK GOTO 2:GOTO 1000
2 FOR i=1 TO 255:CLOSE#i:NEXT i:OPEN#0,0,"1100":PUT 23,5:END
4 FOR i=1 TO 255:CLOSE#i:NEXT i:OPEN#0,0:CLEAR
5 LINPUT("Programma 'TEST.BAS'; save datum: ")dd$
6 SAVE "TEST"+dd$+".BAS":?:?"Gebruikt";40882-FREE;"bytes":END
7
498 REM      stuurmodule voor invoer
500 in$="":tk=FALSE:vm=FALSE:dl=FALSE:is=FALSE:PUT 21:GET x:GET y
510 cr=FALSE:GOSUB 600:IF cr AND LEN(in$)>0 THEN 590
520 IF NOT r THEN 510
530 PUT 22,x,y,32:FNc$(24);"Rekenmachine: ";cr$;in$=""
540 GOSUB 600
550 IF e$<>"" AND e$<>"x" AND e$ " THEN 540
560 GOSUB 700
570 IF NOT is THEN 540
590 RET

```

```

598 REM      invoer karakter en afdrucken numerieke invoer
600 IN$=""
610 PUT 6:GET#5,e$
620 IF (e$="0" AND e$<="9") OR e$="." THEN ?e$;in$=in$+e$:GOTO 610
630 IF e$="r" AND xl THEN r=TRUE
640 IF ASC(e$)=13 AND LEN(in$) 0 THEN cr=TRUE
650 IF ASC(e$)<>8 THEN 690
660 l=LEN(in$):IF l>0 THEN in$=LEFT$(in$,l-1):PUT 24:GOTO 610
690 RET
691
698 REM      controle en verwerking invoer x,: en =
700 IF tk AND e$<>"" THEN 790
710 IF tk AND e$="" THEN is=TRUE:g2=VAL(in$):GOTO 770
720 IF e$="x" THEN vm=TRUE:GOTO 750
730 IF e$=":" THEN dl=TRUE:GOTO 750
740 GOTO 790
750 IF vm OR dl THEN ?e$;
760 g1=VAL(in$):tk=TRUE:in$="":GOTO 790
770 IF vm THEN ?="";g1*g2;
780 IF dl THEN ?="";g1/g2;
790 RET
791
998 REM      hoofdprogramma
1000 GOSUB 10000
1010 xl=TRUE:GOSUB 500
1040 xl=FALSE:IF r THEN PUT 22,x,y:r=FALSE:GOSUB 500
1050 gt=VAL(in$):?FNc$(7);
1060 IF gt>=.999*aw AND gt<=1.001*aw THEN ?"Akkoord.":GOTO 1080
1070 ?"Fout."
1080 ?FNc$(14);"NEW LINE=nog eens/ESCAPE=stoppen";cr$
1100 GET#5,c:IF c=27 THEN 2
1110 ?FNc$(7):?FNc$(24):GOTO 1000
1111
9998 REM      genereren van een som
10000 g1=INT(1000*RND):g2=INT(1000*RND):a=RND
10020 ?FNc$(1);"Maak de volgende berekening:"
10030 IF a<0.5 THEN a$="":aw=g1/g2
10040 IF a>0.5 THEN a$="x":aw=g1*g2
10050 ?FNc$(24);"Type 'r' voor rekenmachine";
10060 ?FNc$(4);g1;a$g2;"":cr$;
10090 RET
10091
19998 REM      initialisatie
20000 FOR i=1 TO 25:CLOSE#i:NEXT i:OPEN#0,0:OPEN#5,5
20010 DEF FNc$(y)=CHR$(7)+CHR$(22)+CHR$(1)+CHR$(y)+CHR$(30)
20020 cr$=CHR$(6)+CHR$(127)+CHR$(8)
20090 RET

```

Henk Hiddink
 Ereprijs 53
 1273 XJ Huizen

SCROLL

SCROLL is een basic-programma dat de karakters op het scherm horizontaal laat draaien, en is zeer geschikt voor spelletjes. Het programma is in basic geschreven zodat iedereen het op een unexpanded Newbrain kan draaien. De data wordt ingelezen en dan gePOKEd in een daartoe geRESERVED geheugen. Als laatste wordt de machinetaal-routine aangeroepen. Hierdoor worden de karakters een plaats verschoven. Natuurlijk is het ook mogelijk de CALL TOP-statement in een FOR/NEXT lus te gebruiken.

Een voorwaarde voor een goede afloop van het programma (en voor een terugkeer naar het operating system) is dat stream 0 als eerste in de stream-table staat genoteerd. Dit is te bewerkstelligen door het programma te RUNnen direkt na het opstarten van de NewBrain, of door het sluiten van alle andere streams. Als bij een poke642,65 een A links boven op het scherm staat kunt u met een gerust hart verder - voor een breed scherm (80 kolommen) geldt poke644,65.

Aan het einde van deze soft-kopij een tip:
Indien men veel geredigeerd heeft en men niet weet of de tekst zoals die op het scherm staat ook als zodanig in het programma-geheugen staat kan men zich op de volgende manier hierover zekerheid verschaffen:

```
10 eerste regel
100 laatste regel
end
```

```
new
(control+)newline of (control+)c
```

Zorg er wel voor dat alles wat ingelezen/gemerged moet worden op het scherm afgedrukt staat zonder een lege regel.

Hans van der Linden
Hoofdstraat 88
5171 DG Kaatsheuvel

```
5 REM xxxxx SCROLL 40/80 : H. van der Linden. xxxxx
6 REM xxx Voor 'n 80-koloms scherm vindt men in de xxx
7 REM x REM-regels de nieuwe data. x
8 REM xxx De x bepaalt het aantal regels. xxx
9 REM xxxxx Zorg dat stream 0 bovenaan staat !!! xxxxx
10 DATA 197,213,245,253,229,229
20 DATA 17,25,0
25 REM 49
30 DATA 6,21
35 REM x
40 DATA 253,33,130,02
45 REM 132
50 DATA 14,39
55 REM 79
60 DATA 253,102,0
70 DATA 253,126,1
80 DATA 254,0
90 DATA 32,2
100 DATA 62,32
110 DATA 253,119,0
120 DATA 253,35
130 DATA 13
140 DATA 32,239
150 DATA 124
160 DATA 254,0
170 DATA 32,2
180 DATA 62,32
190 DATA 253,119,0
200 DATA 253,25
210 DATA 16,220
220 DATA 225,253,225,241,209,193,201
300 IF TOP ` 32050 RESERVE 766
310 RESTORE
320 FOR i=0 TO 57 : READ a: POKE TOP+i,a:NEXT i
400 CALL TOP
410 END
```



CENTRUM VOOR TOEGEPASTE MICRO-ELECTRONIC

PRINTERS VOOR NEWBRAIN

=====

VOOR HET AANSLUITEN VAN EEN PRINTER OP DE NEWBRAIN COMPUTER
IS EEN SPECIALE KABEL NODIG EN EEN SERIE-INTERFACE IN DE PRINTER.

PRIJS GEMIDDELD f 325,-

C.T.M. HEEFT EEN KABEL ONTWERPEN VOOR DE AANSluitING OP DOOR ONS
GELEVERDE PRINTERS, WAARBIJ DE INTERFACE IS INGEBOUWD.

PRIJS f 150,-

SPECIFICATIES: SERIE-INGANG, 9600 BAUD, MET NEWBRAIN CONNECTOR
CENTRONICS PARALLEL UITGANG, 36-POLIGE CONNECTOR
VOEDINGSSPANNING UIT DE PRINTER.

LEVERBAAR:

BROTHER DAISYWHEEL SCHRIJFMACHINES, MET O.A. PROPORTIONEEL SCHRIFT
LOGIC SEEKING, BIDIRECTIONEEL, PLOTMOGELIJKHEDEN, VERWISSELBAAR
LETTERWIEL.

CE-40BS	f 1395,--
CE-50BS	f 1475,--
CE-50BT	f 1895,--

MATRIXPRINTERS

AVT 80A	f 875,--
AVT 100B	f 990,--
TREND JP 80A	f 860,--
STAR GEMINI 10X	f 995,--

ALLE GENOEMDE PRIJZEN ZIJN EXCLUSIEF B.T.W

VERDER LEVEREN WIJ DIVERSE INTERFACES, OOK OP MAAT, VOOR DE SERIE-
COMMUNICATIEPOORT VAN DE NEWBRAIN: ZOALS DE ADM-8 KANAALS MILLIVOLT-
METER MET EEN RESOLUTIE VAN +/- 2000 MV EN DE MCI-1 VOOR METEN EN
REGELEN MET EEN RESOLUTIE VAN 8 BITS.

C.T.M. JAMES WATTSTRAAT 66 1097 DM AMSTERDAM 020-941654

T A P E / T A P E - programma

Onderstaand programma is oorspronkelijk geschreven om
machinecode-programma's te kopiëren. Naderhand is gebleken,
dat het programma met kleine wijzigingen algemeen toepas-
baar is wanneer data van een stream naar een andere stream
moeven worden overgebracht.

Toepassing 1: Kopiëren van tapes (zowel programma's als data)
zie listing 1.

Het programma gebruikt de 2 cassettereorderpoorten (device
1 en 2) van de NewBrain (voor 1 recorder zie toepassing 2).
Alles wat vanaf recorder 1 wordt ingelezen, wordt onder de-
zelfde file-naam weggeschreven naar recorder 2.

Toepassing 2: als 1 met één recorder, zie listing 2.
Is langzamer maar werkt goed.

Toepassing 3: "Lezen" van tapes

Vervang regel 50 door:

```
50 GET 1 1,x:PUT x:GOTO 50      of:  
50 GET 1 1,x:PRINT x;; GOTO 50
```

Bij het "lezen" van BASIC-programma's geeft de eerste wij-
ziging de keywords als token weer. Control-codes blijven
aktief, x=31 wist dus het hele scherm.

De tweede wijziging ondervangt dit bezwaar, maar geeft de
ASCII-waarde van de keywords. Niettemin is deze wijziging
bruikbaar om onlistbare programma's te ontcijferen.

Toepassing 4: voor telex-bezitters

Als printer gebruik ik een gewijzigde telex. Vanwege de
geluidsoverlast is deze 's avonds nauwelijks te gebruiken.
Inplaats van device 8 kies ik dan device 1 of 2 om op
te printen. Ik print dus niet op papier maar op tape. De
volgende dag gebruik ik dit programma om de inhoud van de
tape uit te printen. De tape dient dus als uitwendig

geheugen. Regel 30 moet dan veranderd worden:

```
30 OPENOUT ## 2,8,"75"      (75 = Baudrate telex)
```

Toepassing 5: Tape als uitwendige printbuffer of als uitwendig, permanent printergeheugen.

Om bij langzame (margrietwiel) printers niet voortdurend te hoeven wachten tot (grote hoeveelheden) tekst is geprint, kan het aanzienlijk sneller wegschrijven op tape worden gebruikt. Dit kan nog sneller (wanneer het geheugen van de NewBrain het toestaat) als een grote uitvoerbuffer voor device 1 wordt gebruikt, dus OPENOUT ~~##~~ 1,1,"*20" of OPENOUT ~~##~~ 1,1,"*40" i.p.v. OPENOUT ~~##~~ 1,1. Regel 20 moet dan wel worden aangepast. Zie verder toepassing 4. Als op de tape-uitvoer van een tekstverwerker alleen print-instructies staan, dan kan het programma gebruikt worden om snel een print van de tekst te krijgen.

Toepassing 6: Geen eigen printer

Als je geen printer bezit, maar je mag af en toe de printer van een kennis lenen of je kunt een printer huren (bijv. van de plaatselijke HCC) dan kan je alles wat je wilt printen op een tape verzamelen en op een avond of in het weekend printen.

Toepassing 7: "Printen" van bestanden op video-tape

Sluit de video-recorder-antenne-ingang aan op UHF van de NewBrain en de video-recorder-TV-uitgang op de TV-antenne-ingang (geeft bij Z/W TV vaak slecht beeld, gebruik dan KTV); wijzig de volgende regels:

```
30 CLOSE ## 2: OPEN ## 2,0,1 : REM of ## 2,0,1,"1"
50 GET ## 1,x: PUT ## 2,x: i=i+1: IF i = ..... GOSUB 100:
    REM voorwaarde voor i zelf invullen
55 GOTO 50
80 ON ERROR GOTO 0: GOSUB 100: PRINT enz.
100 LINPUT(" ZET VIDEORECORDER OP OPNAME")a$: FOR j =1 TO
    ..... : PUT ## 2,3:NEXT j: PUT ## 2,31: i=0
```

```
110 LINPUT("ZET VIDEORECORDER AF") a$: RETURN: REM
    stopwaarde voor j zelf bepalen
```

Het bestand kan later direkt met de videorecorder worden bekeken. Wanneer de nummering van de bestandrecords synchroon loopt met de nummering van de video-tape zijn de records zeer snel terug te vinden en is video-tape aanzienlijk goedkoper dan kartonnen kaartjes voor een kaartsysteem. Iets voor de "zwarte" boekhouding?

Toepassing 8: Veranderen van de in- en uitvoerbuffergrootte van device 1 en 2.

De grootte van de in- en uitvoerbuffer van device 1 en 2 worden bepaald door de parameter-string in de OPENIN- en OPENOUT-opdracht. Inlezen in een te grote buffer is wel mogelijk. Andersom niet. Het bestand op de tape moet dan anders geformatteerd worden. Bij dit programma hoeft slechts regel 20 te worden aangepast.

Bijv.

```
20 CLOSE ## 1: OPENIN ## 1,1,"*40": a$ = "*1:" + FILE$
```

om de in/uitvoer buffer te verkleinen van 10 kbytes naar 256 bytes.

Toepassing 9: Overbrengen van bestanden van tape naar disk ?

Misschien is het ook mogelijk na een kleine wijziging programmaas en data van tape naar disk over te brengen. Ik weet het niet, omdat ik alleen met tape werk.

Toepassing 10: Zelf bedenken

Verdere toepassingen laat ik aan jullie intelligentie over. Zelf wil ik het nog eens proberen om de 2 cassette-recorder-poorten met dit programma als "nodes" te gebruiken.

Bij mij is dit programma na kleine wijzigingen een zeer krachtig stuk gereedschap gebleken, wat vaak in de meest onverwachte situaties te gebruiken is. Succes ermee.

Wim Luijt

```

LISTING 1.
10 FORN=1 TO 255: CLOSE#N: NEXT N
20 CLOSE#1: OPEN#1, 1: A$=FILE$
30 CLOSE#2: OPEN#2, 2, A$
40 ON ERROR GOTO 60
50 GET#1, X: PUT#2, X: GOTO 50
60 IF ERRNO=133 RESUME 80
70 PRINT "ERROR"; ERRNO; "IN LINE"; ERRLIN:
  A$="PROBEER OP NIEUW": GOTO 90
80 ON ERROR GOTO 0: A$="BESTAND IS OVERGEBRACHT": CHR$(13)
  + "HET BESTAND BESTAAT UIT " + STR$(I) + " BYTES"
90 CLOSE#1: CLOSE#2: PRINT A$: END

```

```

LISTING 2.
10 FOR I=1 TO 255: CLOSE#N: NEXT N
15 RESERVE 10000: I=0
20 CLOSE#1: OPEN#1, 1: A$="*4": FILE$
40 ON ERROR GOTO 60
50 GET#1, X: I=I+1: POKETOP+I, X: GOTO 50
60 IF ERRNO=133 RESUME 80
70 PRINT "ERROR"; ERRNO; "IN LINE"; ERRLIN: CHR$(13):
  "PROBEER OP NIEUW"
80 ON ERROR GOTO 0: CLOSE#1.
  INPUT("ZET BANDRECORDER KLAAR OP OPNAME")B$:
  OPEN#2, 1, A$
90 FOR J=1 TO I: PUT#2, PEEK(TOP+J): NEXT J: CLOSE#2
100 PRINT "HET BESTAND IS OVERGEBRACHT":
  RESERVE 255+TOP: END

```

NASCHRIFT.

WADAT DIT ARTIKEL GESCHREVEN WAS, HEB IK HET PROGRAMMA
VERSNELD DOOR REGEL 50 TE VERVANGEN.

50 INPUT#1, X\$: PRINT#2, X\$: GOTO 50
DE GEBRUIKTE METHODE IS IN HET VOLGENDE KANT EN KLAAR
PROGRAMMA VERWERKT.

NA VERPLAATSING VAN DE REM IN REGEL 150 EN 160 NAAR
REGEL 140 LIST HET PROGRAMMA ZELFS PROGRAMMA S VANAF
BAND ZONDER DAT ZE GELAAD HOEVEN WORDEN.

VIND JE DEZE WIJZE VAN LISTEN NIET PLEZIERIG DAN KAN
JE DE REGELS 150, 160, EN 210 EN VERDER DELETEN.

```

10 REM TAPE DEVICE: COPIEREN VAN BAND NAAR BAND OF
  ANDERE DEVICES
20 A=10: PRINT "OUTPUT DEVICE", "NIETS= CONSOLE,
  2 OF 1 = TAPE, 8 = PRINTER": PUT(11, 28, 9, 9:
  INPUT(" ")A$: IF (A$ <> "") IF NUM(A$) THEN A=VAL(A$)
30 INPUT("BUFFER INPUT# ")A$: IF A$ <> "" THEN A$="*"+A$
40 IFA=10: INPUT("SCHERMLENGTE ")B$: IF B$ <> "" OPEN#0, 0, B$
50 IFA=8 THEN B$="75": REM BAUDRATE PRINTER
60 INPUT("AANTAL TE COPIEREN FILES ")C$: N=1:
  IF NUM(C$) THEN N=VAL(C$)
100 FOR J=1 TO N: CLOSE#1: OPEN#1, 1, A$
110 IFA=10 OR A=2: IFA$ <> "" THEN A$=A$+"*"+A$
120 IFA=10 OR A=2 THEN B$=A$+FILE$
130 CLOSE#2: OPEN#2, A, B$: PUT(23, 1: I=0: ON ERROR GOTO 170
140 INPUT#1, X$: PRINT#2, X$: PRINT X$: I=I+1: GOTO 140
150 REM GET#1, X: PUT#2, X: IF X < 128 PUT X: GOTO 150
160 REM GOTO 210
170 CLOSE#1: CLOSE#2: IF ERRNO <> 133 PRINT ERRNO; "IN"; ERRLIN:
  GOTO 200
180 PRINT "EINDE VAN DE FILE"; I; "REGELS OF BYTES
  GECOPIEERD"
190 NEXT J
200 PRINT J=1; "FILE(S) GECOPIEERD": END
210 X=X-128: IF X=0 THEN K=57222: GOTO 240
220 K=0: FOR L=57122 TO 57302: IF PEEK(L) > 128 THEN K=K+1:
  IF K=X THEN K=L+1: L=57302
230 NEXT L
240 IF PEEK(K) < 128 PUT PEEK(K): K=K+1: GOTO 240
250 PUT PEEK(K)-128: IF X > 44 GOTO 150
260 IF X=0 THEN K=569200: GOTO 300
270 K=0: FOR L=56920 TO 57119: IF PEEK(L) > 128 THEN K=K+1:
  IF K=X THEN K=L+1: L=57119
280 NEXT L
290 IF PEEK(K) < 128 PUT PEEK(K)+128: K=K+1: GOTO 290
300 PUT PEEK(K)
310 GOTO 150

```

Met behulp van dit kleine programma en je NewBrain kun je alles. Nou ja, bijna alles. Je kunt het programma o.a. gebruiken als MINI-TEKST VERWERKER. Gebruik dan onderstaand programma:

```
5 DATA 30,0,231,49,216,254,64,200,30,8,231,48,195,238,127
10 FOR I=1 TO 255:CLOSE#I:NEXT I:ON ERROR GOTO 25
15 IF TOP>32760 RESERVE 20:FOR I=32750TO 32764:READ MN:POKE I,MN:NEXT I
20 OPEN#0,0,"L65":OPEN#8,8:CALL 32750
25 FOR I=1 TO 255:CLOSE#I:NEXT I:END
```

MINI-TEKSTVERWERKER

Je hebt standaard een beeldscherm van 65 regels lang. De baudrate voor de printer staat op 9600 baud. Pas dit voor jezelf aan in regel 20. Na dit gedaan te hebben, ga als volgt te werk:

Typ de tekst in met behulp van je NewBrain. Het naar de printer sturen van de tekst kan op verschillende manieren. Je kunt na elke regel een NEW-LINE geven en de tekst dan regel voor regel naar de printer sturen. Je kunt ook eerst de hele tekst intypen en dan op het eind in 1 keer naar de printer sturen door middel van CONTROL+NEW-LINE of door CONTROL+C in te toetsen. Let er nu op dat je bij het invoeren van tekst NIET op de NEW-LINE toets drukt. Je stuurt dan namelijk alleen je laatste regel naar de printer. Wil je de tekst meerdere keren uitprinten, dan kan dit heel eenvoudig door het intoetsen van CONTROL+NEW-LINE of door CONTROL+C.

Wil je uit het programma, druk dan op "@" (=CHR\$(64)).

Door de full-screen-edit mogelijkheden van de NewBrain is dit een erg praktisch en snel programma.

COPIEER-PROGRAMMA

Wil je het programma gebruiken als copieer-programma, verander dan regel 5, regel 15 en regel 20 als volgt:

vb. van CASSETTE 1 naar CASSETTE 2:

```
5 DATA 30,1,231,49,216,30,2,231,48,195,238,127
15 IF TOP>32760 RESERVE 20:FOR I=32750TO 32761:READ MN:
```

(vervolg)

```
POKE I,MN: NEXT I
20 OPEN#1,1:A$=FILE$:OPENOUT#2,2,A$:CALL 32750
```

vb. van FLOPPY-DISK naar CASSETTE 1:

```
5 DATA 30,12,231,49,216,30,1,231,48,195,238,127
15 IF TOP>32760 RESERVE 20:FOR I=32750 TO 32761:READ MN:
POKE I,MN: NEXT I
20 INPUT("Welke file: ")A$:OPEN#12,12,A$:OPENOUT#1,1,A$:
CALL 32750
```

Let bij het copieeren van/naar cassette op het bufferformaat.

Dus vb. 20 OPEN#1,1,"*40":A\$=FILE\$:OPENOUT#2,2,"*40:"+A\$:CALL 32750

TERMINAL-PROGRAMMA

Gebruik je NewBrain als terminal door het veranderen van het programma als volgt:

```
5 DATA 30,5,231,49,216,30,9,231,48,195,238,127
15 IF TOP>32760 RESERVE 20:FOR I=32750 TO 32761:READ MN:
POKE I,MN: NEXT I
20 OPEN#5,5:OPEN#9,9:CALL 32750
```

Je kunt dit laatste programma ook weer veranderen, zodat de mogelijkheid bestaat, dat je "eruit springt" door het intoetsen van "@" (=chr\$(64)).

Regel 5 en regel 15 moet je dan weer als volgt veranderen:

```
5 DATA 30,5,231,49,216,254,64,200,30,9,231,48,195,238,127
15 IF TOP>32760 RESERVE 20:FOR I=32750TO 32764:READ MN:
POKE I,MN: NEXT I
```

Veel succes met bovenstaande programma's.

Tilburg, juni 1984

Door: Jos Hermans
Mozartlaan 523
5011 SM Tilburg

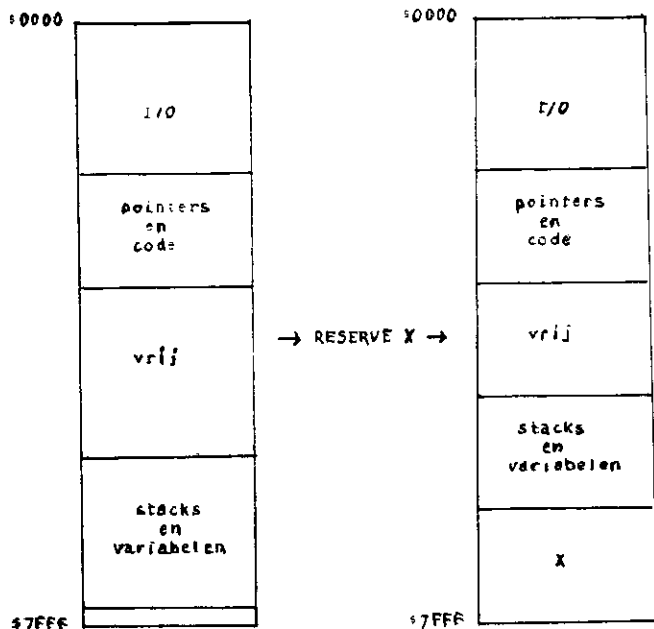
BASIC - MACHINETAAL INTERFACE

De standaard Newbrain heeft in zijn Basic wat mechanismes meegekregen voor het opzetten en executeren van machinetaal-routines, zoals CALL, RESERVE, PEEK en POKE.

Ook het doorgeven van Basic-parameters en gebruik van systeemroutines is vrij goed geregeld. Het enige primitieve gedeelte hierin is het geheugen-management. Basic kent alleen het RESERVE-statement, waarbij een aantal bytes van Basic gepikt kan worden voor het implementeren van machine-routines. Die bytes ben je dan echter voorgoed kwijt, of je moet het systeem weer uit en aan zetten om ze terug te krijgen. Bovendien kan die ruimte niet via een machine-routine aan worden gevraagd.

Voor de ontwikkeling van een eenvoudige Z80-assembler, geschreven in een combinatie van Basic en machinetaal, had ik echter een routine nodig waarbij ik naar hartelust bytes kon reserveren en weer vrijgeven, zonder dat Basic hier erg in had. Dit resulteerde in onderstaande routines die na uittesten nog bleken te werken ook. Enige uitleg lijkt mij hier wel op zijn plaats.

Basic gebruikt twee gedeeltes van het geheugen. De pointers en de programma-code worden onder het I/O-gedeelte opgeslagen. De variabelen en de stacks worden op de top van het RAM gezet. Deze twee gedeeltes groeien naar elkaar toe en daartussen zit de vrije RAM. Het RESERVE-commando haalt hier bytes van af door het stack/variabelen-gedeelte een stuk naar beneden te schuiven in RAM, waardoor de TOP lager- en het resterende gedeelte vrij- komt (zie figuur).



Het idee achter de gegeven routines is dat ze hetzelfde doen maar dit ook om kunnen keren.

DELOBJ

Hiervoor wordt gebruik gemaakt van de subroutine DELOBJ. Deze vernietigt de OBJECT-code. Dit vanwege het volgende: Voor het vrijmaken en teruggeven van geheugen wordt het gehele STACK/VAR.-gedeelte met een Block-Move verschoven. Basic compileert de Source-Code (zoals ingetikt op je toetsenbord) naar een sneller executeerbare Object-Code. Deze code echter, bevat absolute pointers naar de Line-Number-Table (LNT) en de Symbol-Table (ST). Daar deze tabellen in het gebied zitten dat verschoven wordt, worden de pointers waardeloos en de Object-Code dus ook. Het vernietigen hiervan is geen bijzonderheid, daar de statements RESERVE, DELETE, CLEAR en MERGE dit ook doen (zie Techn. Manual: hfdstk 9.2, blz. 4.00). Het vernietigen van de code gebeurt door de Object-Code pointer (OTOP) gelijk te maken aan de Source-Code pointer (STOP) en door alle pointers uit de LNT naar de object-code nul te maken (betekent: "Regel op dit moment niet in gecompileerde vorm aanwezig". Zie Techn. Manual note 6, pag. 11).

RESERVE

Deze routine moet er voor zorgen dat geheugen wordt gereserveerd. Hij gaat er hierbij van uit dat het aantal te reserveren bytes staat aangegeven op de geheugenplaats: \$7FFE/\$7FFF in normale Z80 - low/high format.

Er wordt eerst gecontroleerd of het aantal aangevraagde bytes niet resulteert in een compleet overschrijven van het vrije gebied, inclusief de SOURCE-code. Anders resulteert dit in een foutmelding met nummer 10 (insufficient memory space).

Hierna wordt het STACK/VAR. block m.b.v. een LDIR-instructie naar beneden geschoven in het geheugen, waardoor het gevraagde aantal bytes vrij komt.

Deze actie heeft totaal geen zin als ook niet ALLE absolute pointers naar dit gebied aan worden gepast, alsook eventueel relevante pointers in het gebied zelf.

Allereerst zijn er de pointers die de gebieden bepalen binnen dit block en op een offset van IY staan. Dit zijn resp.; de USRPTR, FORPTR, GSBPTR, STBAS, ARBAS, SBAS, LBAS en TOP, die staan op de locaties: IY+\$12 en verder.

Deze pointers worden alle verlaagd met het aantal te reserveren bytes. Ook het register IX ondergaat deze behandeling, daar deze de functie heeft van program-counter en wijst naar de eerstvolgende te executeren regel in de LNT.

De pointers vanuit de gebieden voor strings, arrays, symbols en line-numbers zijn OF relatief t.o.v. de base-pointers (die reeds aan zijn gepast), OF wijzen naar de source-code (die niet is verschoven). Deze behoeven dus verder geen aanpassing voor de nieuwe situatie.

Zowel de FOR- als de GOSUB- stack bevatten 3-byte referenties naar de

LNT die door Basic t.z.t. vertaald worden naar een geheugenplaats IN de LNT. Ook deze behoeven dus geen aanpassing. Bij de USER-stack gebeurt ook nog het een en ander. Helaas is het zo dat dit niet al te best staat beschreven in de Techn. Manual. Vast staat dat deze in ieder geval wordt gebruikt door de Object-code, welke vernietigt wordt. Ik denk dat de stack dan voornamelijk gebruikt wordt bij het analyseren van ingewikkelde rekenkundige uitdrukkingen. Het is dan ook zaak de routine niet midden in zo'n uitdrukking aan te roepen daar dan de stack verziekt wordt. Ik kan me echter niet voorstellen dat iemand dit zou willen doen. Verder maken de user-defined functions op een onduidelijke manier gebruik van de stack. De routine dus ook hierin niet aanroepen. Het zou dus het beste zijn de stack maar gewoon te deleten, ware het niet dat ook de FORPTR nog enige mysterieuze connecties hiermee heeft. We laten de stack dus maar voor wat hij is en zullen voorzichtig moeten zijn bij het nesten van de routine in FOR-NEXT blocks. Alles gaat in ieder geval goed binnen 1 FOR-NEXT loop, de rest moet je zelf maar uittesten. Betreffende de pointers INPTR en SVINPT bestaat nog enige onzekerheid maar tot nu toe is alles goed gegaan. Als finishing touch wordt dan DELOBJ aangeroept en het werk is gedaan. Deze routine doet dus vermoedelijk hetzelfde werk als het RESERVE-commando met dit verschil, dat deze routine ook vanuit machinetaal aangeroept kan worden.

FREE

De routine FREE moet het aantal bytes dat op \$7FFE staat aangegeven aan het Basic-memory teruggeven. Deze routine is dus het spiegelbeeld van RESERVE. Aftrekken wordt optellen, LDIR wordt LDDR etc., etc. Een opmerking hierover: De routine introduceert nog een pointer MTOP. Deze moet staan op geheugenplaats \$7FFC/\$7FFD en geeft de top van de machine-routines aan. Dit om te voorkomen dat het STACK/VAR. block deze machine-routines gaat overschrijven. Er wordt dus gekeken of het aantal terug te geven bytes wel beschikbaar is tussen MTOP en TOP. Als dit niet zo is geeft ook deze routine foutmelding 10. Misschien slaat de foutmelding in dit geval niet helemaal op de situatie maar is toch maar gedaan uit oogpunt van consistentie.

Aldus hebben we nu ook voor machine-routines geheugen-management tot onze beschikking, zonder dat Basic en machinetaal elkaar bijten. De implementatie hiervan op machine met uitbreiding (96K of meer) is hier niet bekeken. Dit zal naar verwachting echter niet mogelijk zijn.

Vragen e/o opmerkingen hieromtrent kun je altijd bij mij kwijt.

Veel Succes,

T. Uijldert,
Zandwierde 79,
1353 CH Almere-Haven.

MACHINECODE - BASIC interface

```

; RESERVE routine.
;
; This routine reserves the number of bytes, given in location
; $7FFE/$7FFF so that it cannot be used by the BASIC-interpreter.
; Only registers IX and IY are preserved.
;
;
; ORG    $7F00      ; Routine start-address.
;
; BYTEN0 EQU    $7FFE      ; Number-pointer.
; MPTR   EQU    $7FFC      ; Machine-code pointer.
;
;
; RESERVE  LD     L, (IY+$12)      ; HL = USRSTP
;          LD     H, (IY+$13)      ;
;          PUSH  HL               ; save USRSTP
;          BC, (BYTEN0)           ; fetch no. of bytes.
;          LD     HL, BC           ; HL = New USRSTP
;          JR     C, NOMEM        ; Out of memory?
;          LD     D, H             ; DE = New USRSTP
;          LD     E, L             ;
;          LD     C, (IY+$0E)      ; BC = STOP
;          LD     B, (IY+$0F)      ;
;          SBC     HL, BC          ; HL = New USRSTP-STOP
;          JR     C, ?             ; < ?
;          POP    BC              ; BC = USRSTP
;          LD     DE, BC           ;
;          LD     L, (IY+$20)      ; HL = TOP
;          LD     H, (IY+$21)      ;
;          LD     HL, BC           ; HL = POP-USRSTP (#count)
;          LD     E, H             ; BC = count
;          LD     C, L             ;
;          POP    HL              ; HL = USRSTP
;          LD     DE, (BYTEN0)     ; block move
;          LD     IY, DE           ; DE = no. of bytes.
;          LD     BC, $0012        ; save IY
;          LD     IY, BC           ; BC = disp.
;          LD     B, $08           ; B = loop-count
;          LD     L, (IY+$00)      ; HL = PTR
;          LD     H, (IY+$01)      ;
;          SBC     HL, DE          ; PTR = PTR-requested bytes.
;          LD     (IY+$00), L      ; PTR = (HL)
;          LD     (IY+$01), H      ;
;          INC     IY              ; IY = next pointer.
;          INC     IY              ;
;          DJNZ   PTRLOOP         ; restore IY
;          POP    IY              ;
;          PUSH   IX               ;
;          POP    HL               ;
;          SBC     HL, DE          ; adjust IX to LNT.
;          PUSH   HL               ;
;          POP    IX               ;
;          CALL   DELOBJ           ; delete object code
;          RET                     ; end
;          POP    HL               ; restore stack
;          LD     A, $0A           ; error no. = 10
;          RET
;
; NOMEM

```



```

;
;
; DELOBJ - routine
;
; This routine deletes the object code built up by Newbrain-basic,
; through zeroing the LNT-pointers to the object-code. The LNT-
; pointer is reset to the STOP value.
; IX and IY are preserved.
;

```

```

DELOBJ  LD  L,(IY+#0E)  ; HL = STOP
        LD  H,(IY+#0F)  ;
        LD  (IY+#10),L  ; STOP = HL (=STOP)
        LD  (IY+#11),H  ;
        LD  B,(IY+#1E)  ; DE = USRPTR
        LD  D,(IY+#1F)  ;
        LD  C,(IY+#20)  ; PC = TOP
        LD  B,(IY+#21)  ;
        LD  A,#00       ; A = 0
        LD  H,B         ; copy BC to HL
        LD  L,C         ;
        SBC  HL,DE      ; HL = TOP-DE
        RET  Z          ; end, if TOP=DE
        INC  DE         ;
        INC  DE         ; DE = DE+4
        INC  DE         ;
        LD  (DE),A      ;
        JPH  DE         ; object code pointer + 0
        LD  (DE),A      ;
        INC  DE         ;
        JR   DELOOP     ; repeat
END.

```

```

;
;
; FREE - routine
;
; This routine returns a number of requested bytes in $7FFE/$7FFF,
; to BASIC. It will not go past the address of the MRPTR.
; Only registers IX and IY are preserved.
;

```

```

FREE    LD  BC,(BYTEND)  ; BC = no. of requested bytes.
        LD  L,(IY+#20)  ; HL = TOP
        LD  H,(IY+#21)  ;
        PUSH HL         ; save TOP
        ADD  HL,BC      ; HL = NewTOP
        JR   C,NOMEM    ; Out of memory?
        LD  D,R        ; DE = NewTOP
        LD  E,L        ;
        LD  HL,(MRPTR)  ; HL = MRTOP
        SBC  HL,DE      ; HL = MRTOP-NewTOP
        JP   C,NOMEM    ; \ ?
        POP  HL         ; HL = TOP
        PUSH HL         ;
        LD  C,(IY+#12)  ; BC = USRPTR
        LD  B,(IY+#13)  ;
        SBC  HL,BC      ; HL = TOP-USRPTR = count
        LD  B,H        ; BC = count
        LD  C,L        ;
        POP  HL         ; HL = TOP
        LDEF          ; Block move
        PUSH IY         ; save IY
        LD  BC,#0012    ; BC = disp
        ADD  IY,BC      ; (IY) = ptr. +
        LD  DE,(BYTEND) ; DE = no. of requested bytes.
        LD  B,#08      ; set loop-count
        LD  L,(IY+#00)  ; HL = ptr.
        LD  H,(IY+#01)  ;
        ADD  HL,DE      ; HL = HL+no. of bytes.
        LD  (IY+#00),L  ; ptr = (HL)
        LD  (IY+#01),H  ;
        INC  IY         ; adjust IY
        INC  IY         ;
        DJNZ PTRLOOP2  ; repeat until B=0
        POP  IY         ; restore IY
        ADD  IX,DE      ; adjust IX in LNT
        CALL DELOBJ     ; delete object code
        RET             ; end

```

TRICK LOADER

Het laden van machinetaalprogramma's op de NewBrain vergt nogal wat tijd. Behalve dat het programma van de band gelezen moet worden, duurt het bijna even lang om het vanuit een buffer op de juiste plaats in het geheugen te "poken".

De transmissiesnelheid van de band (of van de schijf) naar de NewBrain is in de hardware vastgelegd door de FSK-frekwenties. Daarom is de enige manier om e.e.a. sneller te laten verlopen het verkorten van de relocatieprocedure.

Zelf heb ik hier de volgende oplossing voor gevonden. In plaats van het programma eerst in een buffer in te lezen, krijgt de NewBrain een nieuwe buffer toegewezen en wel op de plaats waar we het programma uiteindelijk willen hebben. Het startadres voor deze buffer is te vinden in de stream table. Van de stream table is het beginadres weer te vinden op locatie 86 en 87. Hebben we de stream table gevonden, dan moeten we weten wat de cassette stream is. Om het niet te moeilijk te maken sluiten we eerst alle streams, behalve A0. Daarna openen we de cassette, waardoor deze automatisch op de tweede plaats in de tabel komt te staan. Elke stream table entry is 6 bytes lang. De laatste twee bytes geven het adres aan van de device own buffer. Door nu deze adressen te veranderen geven we het I/O System een ander bufferadres.

Aangezien echter de eerste 6 bytes van de buffer worden gebruikt door het I/O System, moeten we het startadres 6 plaatsen lager aangeven dan het beginadres van het machinetaalprogramma. Deze eerste 6 bytes bevatten echter wel relevante informatie en moeten dus verplaatst worden van de oude naar de nieuwe buffer.

Het zal duidelijk zijn dat het machinetaalprogramma in zijn geheel in de "buffer" moet passen; de buffer moet dus minstens zo groot zijn als het programma. Tevens moeten we ervoor zorgen, dat het programma in een gereserveerde ruimte komt.

Hebben we het een en ander zo uitgevoerd, dan blijkt er het nodige mis te gaan.

- 1) Na close* geeft de computer er de brui aan. Het Operating System probeert de stream op het nieuwe adres te sluiten, maar dat gaat fout. De oorspronkelijke waarde van het startadres moet weer worden teruggeschreven in de streamtable, voordat we de stream sluiten.
- 2) Als we het machinetaalprogramma willen runnen, dan blijkt dat het niet werkt. De NewBrain slaat informatie in een buffer stacksgewijs op; alles komt dus achterstevoren in het geheugen te staan. Om het programma nu toch goed in de "buffer" te krijgen,

moeten we het op de juiste wijze op de band zetten. Dat wil zeggen, dat we tijdens het wegschrijven de outbuffer achterstevoren moeten laden, dus te beginnen met het hoogste byte en te eindigen met het laagste.

- 3) Het wegschrijven naar de cassette gebeurt in blokken van 256 bytes. Voor een programma van 1024 bytes zouden we dus in principe 4 een buffer moeten openen van 4 blokken. De NewBrain schrijft zelf nog 6 additionele bytes weg, zodat we voor 1024 bytes 5 blokken moeten openen.

Houden we rekening met bovengenoemde eigenaardigheden, dan komen we tot een programma dat eruit ziet als onderstaande listing. Het is bedoeld voor de unexpanded NewBrain en een gewone audio cassette-recorder. Dit programma werkt nu drie maanden fool proof voor verscheidene machinetaalprogramma's, zoals HiSoft Pascal, HiSoft Assembler en Monitor.

Regels 75 - 240 bevatten het laadprogramma. Regels 300 - 360 bevatten het programma om de data op de juiste manier op de band te zetten, nadat ze eerst op de oude manier geladen zijn. HiSoft programma's mogen niet eerst geRUNd worden omdat deze direct na de start middels een relocatieprocedure de jump table veranderen.

Er zijn natuurlijk nog andere mogelijkheden om het laden van machinecode te bekorten, zoals een machinetaalprogramma dat de relocatie uitvoert, of een gewijzigde device driver. Bij de hier beschreven methode staat de machinecode direct op zijn plaats; een machinetaalprogramma, hoe snel ook, zal toch altijd alle bytes stuk voor stuk moeten verplaatsen.

J.P. Koning
Beatrixstraat 16
2161 RV Lisse
02521-12742

```

10 REM:      ** VOORBEELD LEECPROGRAMMA TRICKLOADER **
20 REM:
30 REM:      DIT PROGRAMMA LEEFT EEN FILE VAN 25 BLOKKEN VAN CASSETTE
40 REM:      EN PLAATST DIT DIRECT IN HET GEWENSTE TE BEGINNEN BIJ
50 REM:      ADRES 12000.
55 REM:
70 OPEN#1,1,"A00:VOORBEELD"      : REM 1 BLOK MEER OPENEN
80 LET SA=12000                    : REM LAADADRES
90 LET ST=PEEK(00)+256*PEEK(07)    : REM BEGIN STREAMTABLE
100 LET OB=PEEK(ST+10)+256*PEEK(ST+11) : REM BEGIN CASSETTEBUFFER
110 FOR J=0 TO 5                  : REM 6 BYTES BEVATTEN
120 LET V=PEEK(OB+J)              : REM INFORMATIE VOOR O.S.
130 POKE(SA+J),V                  : REM EN MOETEN MEE NAAR
140 NEXT J                        : REM NIEUWE BUFFER
150 LET SH=INT((SA-6)/256)         : REM POKE WAARDE HIGH ADRES
160 LET SL=(SA-6)-(256*SH)         : REM " " " LOW "
170 LET OH=INT(OB/256)            : REM IDEM VOOR HUIDIGE -
180 LET OL=OB-(OH*256)            : REM BUFFER I.V.M. TERUGKEER
190 POKE(ST+10),SL:POKE(ST+11),SH : REM NIEUW STARTADRES NAAR
195 REM                          : REM STREAMTABLE
200 GET#1,Z                       : REM START LEZEN TAPE
210 POKE(ST+10),OL:POKE(ST+11),OH : REM SCHRIJF JUISTE ADRES
220 CLOSE#1                      : REM SCHRIJF I.V.M. TERUGKEER
230 CALL SA                       : REM TERUG EN SLUIT # AF
240 END                          : REM RUN GELADEN PROGRAMMA
?
```

```

250 REM      ** VOORBEELD WEGSCHRIJFPROGRAMMA TRICKLOADER **
260 REM
270 REM      DIT PROGRAMMA SCHRIJFT EEN FILE WEG VAN 25 BLOKKEN
280 REM      TE BEGINNEN OP ADRES 12000
290 REM
300 OPENOUT#1,1,"*26:VOORBEELD"
310 FOR I=6400 TO 0 STEP -1       : REM 25 BLOKKEN, HOOG NAAR LAAG
320 LET Z=PEEK(12000+I)          : REM PROGRAMMA STAAT OP 12000
330 PUT#1,Z                      : REM SCHRIJF OP DE BAND
340 NEXT I
350 CLOSE#1                      : REM EN SLUIT AF
360 END
```

LUIS in de NewBrain.

Bij het testen van een programma kwam ik een "luis" op het spoor, die ik nog nergens gedocumenteerd heb gezien. Voor de volledigheid: mijn NewBrain is van model A en werkt volgens OS 1.91. Het gaat om de index van een variabele in een array. Wanneer die index 0 is tengevolge van het aftrekken van twee variabelen, die in het programma reeds gelijk aan 0 gemaakt zijn, dan geeft de computer ERROR 6. Voorbeeld: Zet de computer aan en tik in: ?a(j-1) NEWLINE. Zoals men verwachten mag, is het resultaat 0. Maar....: zet de computer (af en weer) aan en tik eens in: i=0:j=0:?a(j-1) NEWLINE. Dit levert (ten onrechte !) ERROR 6.

Het is echter niet noodzakelijk uw NewBrain onmiddellijk het raam uit te gooien, want: i=0:j=0:?a(-i+j) NEWLINE levert het correcte resultaat 0 op. Ook: i=0:j=0:?a(j-1+0) NEWLINE geeft 0 als resultaat. Mogelijk zijn er nog meer manieren om deze luis te omzeilen. Het is maar een weet. Met computergroeten,

A. Krauzen.

Om de file-namen (= FILE\$, de titels van de programma's of data) van de band te lezen, zonder dat het aanwezige programma of variabelen worden beïnvloed gebruik ik MERGE "22222". Wanneer het gewenste blok is gevonden druk ik op * om de band te stoppen. Is het programma of data-blok groter dan default (*4) dan gebruik ik MERGE"*40:22222" anders stopt de recorder na het inlezen van de FILE\$ met ERROR 131. De file-naam verschijnt dan wel op het scherm. VERIFY "22222" werkt ook. Het enige waar je voor op moet passen is om nooit een programma of data onder de naam "22222" te save of weg te schrijven.

Wim Luijt

NewBugs

1 In een BASIC programma kwamen de volgende regels voor:

```
10 OPEN #5,5: DIM a$(100): a$(68)= "O.K."
1720 GET #5,a: IF(a>50)+(a<+9) GOTO 1720
1730 IF a=49 CLEAR a$()
1740 GOTO 2000
2000 PRINT a$(68)
```

CLEAR a\$() was opgenomen om voldoende geheugenruimte vrij te maken voor de voortzetting van het programma. Wanneer tijdens de uitvoering van regel 1720 op toets 2 (ASCII 50) werd gedrukt gaf dit bij de uitvoering van regel 2000 ERROR 7!!! Onderzoek leerde dat a\$(100) niet meer bestond !! Tussenvoeging van 1725 PRINT a toonde aan, dat na 1720 a=50, en tussenvoeging van 1715 PRINT a\$(68){gaf OK} toonde aan dat a\$() nog niet was gecleared. Verandering in 1730 IF a=49 THEN CLEAR a\$() gaf hetzelfde resultaat.

Na verandering in

```
1730 IF a=49 STOP: CLEAR a$()
```

bleef a\$() wel in takt. In de praktijk voldoet dit echter niet, omdat de totale uitvoering van het programma tot meer dan 4 h kan oplopen en daarom meestal 's nachts gedraaid wordt.

De volgende oplossing voldoet wel maar kost één regel meer:

```
1730 IF a=50 GOTO 1750
1740 CLEAR a$()
1750 GOTO 2000
```

Volgens het handboek (pg.47) mag een IF statement worden gevolgd door ieder keyword. Hier blijkt dus dat dit niet zo is.

2 Formatter

Het keyword STR\$() dient om een numerieke variabele in een string variabele om te zetten. Om de string een bepaalde vorm te geven kan d.m.v de formatter aangegeven worden hoe het getal in de string moet worden gezet, zie pg 112 van de handleiding. De gegeven informatie vind ik te beknopt om er prettig mee te kunnen werken.

Hoewel ERROR 75 suggereert dat de waarden tussen de [] moeten liggen tussen 0 en 255 blijkt dat het totaal niet groter mag zijn dan 10, dus

$n \leq 10$ en $n+m \leq 10$.

Bovendien mogen voor n en m geen variabelen worden gebruikt. Vandaar ook vermoedelijk de ingewikkelde subroutine 310 in de Basicode subroutines.

a=123456.12345: i=10 : PRINT STR\$(a[i]) levert ERROR 75.

Het gebruik wordt vriendelijker als je rekening houdt met het volgende:

- de eerste positie van de string geeft aan of het getal positief of negatief is, met resp. " " en "-";
- de laatste positie is altijd " " (spatie);
- deze twee posities, de decimale punt positie en de 4 posities voor de exponent moeten bij de waarde(n) in de formatter worden opgeteld om de totale string lengte te verkrijgen (maximaal 17);
- is het getal te groot om in de string te passen dan gebeurt het volgende:
 - voor de decimale punt te lang, wordt afgedrukt als *****
 - na de decimale punt te lang, wordt afgekapt (geen afronding!); of als

- om grote getallen weer te geven kan de volgende constructie gebruikt worden:

```
a= INT(10000000000*a+0.50/10000000000: PRINT STR$(a)
```

deze geeft vaak vorspelbaarder resultaten dan het gebruik van de formatter.

```
Probeer ook eens : a=.12345678955555: PRINT STR$(a[1])
```

en vul achtereenvolgens in 8F, 9F, 10F en 11F.

Begrijp jij het? Ik niet.

Wel toegestaan, maar niet in de handleiding vermeld, is nE als formatter.

3 Basicode-2

Aan het basicode leesprogramma moet de volgende regel worden toegevoegd:

```
305 IF SR<0 THEN SR$ = "-" +SR$
```

anders gaat het min teken van een negatief getal verloren! Willen de ontwerpers hiermee suggereren, dat de NewBrain geen negatieve punten kent?

4 Array element Ø

Probeer het volgende (OPTION BASE = Ø):

```
A=Ø:B=Ø:PRINT A(A-B)
```

Resultaat: ERROR 6

```
A=Ø:B=Ø:C=A-B:PRINT A(C)
```

Resultaat: ERROR 6

Bovengenoemde konstruktie kwam in een Hobby-skoop programma voor dus op andere computers kan het kennelijk wel.

Verder geprobeerd:

```
A=Ø:B=Ø:C=A-B:PRINT C* 1E99
```

Resultaat: Ø ; dus C is wel degelijk Ø

Oplossing:

```
A=Ø:B=Ø:PRINT A(ABS(A-B))
```

5 LIST

Een opdracht LIST kan vervangen worden door de "TOKEN" vorm in te typen, Graphics/A,afgekort als G/A G/A90 heeft hetzelfde resultaat als LIST 90.

LIST # 8,90 list regel 90 op de printer (wanneer de printer stream geopend is). G/A = 8,90 geeft echter ERROR 35. LIST65000-65100 gaat, G/A 65000-65100 gaat niet.

6 Grootte van Basicode programma's.

Begin juni zond Hobbyskoop het programma "Kleinste kwadraten"uit. Dit programma is ca 25 kbyte lang.

Met mijn NewBrain kon ik het maar tot ongeveer regel 19870 laden. Met een van REMs gekuiste versie kwam ik iets verder, maar had ik nog niet het gehele programma. Toch kan dit als volgt:

- Volg de werkwijze uit de NewBrain On-Line. Dit levert een onvolledig programma op, programma 1.
- Begin opnieuw volgens de handleiding tot punt 5 (pg. 52 van On-Line).

→ Spoel het bandje met de radio-opname tot ongeveer halverwege het geratel. ←

- Wacht op de instructie "Start cassette en geef NewLine" en voer die uit.
- Volg verder de handleiding. Dit geeft weer een onvolledig programma, programma 2.
- Merge programma 1 en programma 2.
- Controleer of de laatste regel is overgekomen (Copyright NOS) en zo niet maak programma 2 op-nieuw maar start later.

Later heb ik het programma aangepast aan de NewBrain waardoor het sneller is geworden en "slechts" 11 kbyte lang is. Voor belangstellenden heb ik een copy beschikbaar.

Technische aspecten van de Expansion Interface. Deel 2

In deel 1 zijn reeds een aantal basispunten behandeld van de expansion interface. In dit deel wil ik het verhaal afronden met een beknopte beschouwing van de systeemvariabelen, video onder het paged operating system en een waarde oordeel.

Systeemvariabelen en -tabellen.

Hieronder worden enkele van de belangrijkste systeemvariabelen zoals die voor het paged operating system bekend zijn in het system page description toegelicht. De cijfers zijn in hex.

algemene variabelen:

33	bit 7 "1" betekent dat het paged operating system in gebruik is.
5A/5B	video adres van cursor positie, voor paged geldt het adres als de video pages in het Z80 geheugen zitten (zie bij video)
5C/5D	video adres van "raam" van zichtbare schermdevice.
77/78	adres van kopie van karaktergenerator, voor paged system als deze kopie in slot 1 zit.

uitsluitend paged system variabelen:

8B/9A	pagenummers van achtereenvolgens slots 0 t/m 7
9F/A0	pagenummer van het laatste video RAM page
A1/A2	pagenummer van eerste page
A8/A9	eind van het gebruikte deel van de system page +1. Vanaf dit adres kunt u eigen routines plaatsen die andere pages in bepaalde slots gebruiken.
AE/AF	pagenummer van (ROM)page waarin zich de kopie van de karaktergenerator bevindt.

Enkele paged system werkgebieden zijn er ook. Ze nemen relatief veel ruimte in beslag en zijn bedoeld voor de administratie van MMS, video, ROM routines enzovoort. Zie het system page description.

Video.

Is het bij de standaard NewBrain zo dat het scherm zich ergens binnen het Z80 bereik van het geheugen bevindt, met linksbovenhoek adressen als een veelvoud van 64 bij een 40 koloms scherm en 128 bij een 80 koloms scherm (het scherm heeft een anderhalf keer zoveel geheugen nodig als bruikbaar is), bij het paged operating system is er het verschil dat het zichtbare video RAM altijd buiten het Z80 geheugen zit als er door een programma niet mee gemanipuleerd wordt. Voor normaal gebruik levert dit geen problemen op maar voor programma's die rechtstreeks in het schermgeheugen poken is het een ramp. Spelletjes e.d. moeten dan ook bij voorkeur van de editor commando's gebruik maken, die altijd geldig blijven.

Het is mogelijk om het video RAM in het Z80 geheugen te stoppen tegelijk met het draaien van het betreffende programma, maar er zal straks blijken dat dat niet zonder meer aan te raden is.

Het video RAM kan maximaal 32 K RAM omvatten. Het is dus niet mogelijk om bij onbepaald geheugen ook onbepaald lange tekst- of grafische schermen te maken. Dit komt doordat de video hardware slechts de RAM kan lezen die binnen de NewBrain zelf zit. Er zijn dus ook vaste pagenummers gereserveerd voor video of normaal gebruik, immers, een pagenummer stelt een fysisch stukje RAM voor.

Als de videopages in zijn geheel in het Z80 bereik komen, en wel in de slots 4 t/m 7 voor de achtereenvolgende videopages, dan gelden daarvoor alle waarden die voor videostreams verkregen kunnen worden, zoals bijv. ook de pen(7) en pen(8) functies voor grafische schermen, of bijv. het adres van de cursor zoals gegeven op locatie 5A/5B. Zo beschouwd is het dus t.o.v. niet paged system niet moeilijker om bepaalde plaatsen in het video RAM te bereiken. De moeilijkheid is alleen dat de videopages eerst ingebracht moeten worden. Dit kan bijv. m.b.v. de GETVIDEO routine (aanroep 84H). De oorspronkelijke pages gaan dan uiteraard verloren, dus het is nodig dat de inhoud van S4 t/m S7 bewaard worden, totdat de gewenste manipulaties op het video geheugen gedaan zijn, en de oorspronkelijke slots weer gelezen kunnen worden.

Vaak zal het zo zijn dat men een machinetaalroutine heeft die men in BASIC heeft ingeladen boven TOP, dat men van tevoren geRESERVED heeft. Die routine zal dan in vele gevallen in slot 4 zitten. Dat is dus heel vervelend, want om er videomanipulaties mee uit te laten voeren moet de routine in slot 0, 1, 2 of 3 zitten. Slot 1, 2 en 3 kunnen door BASIC opgevuld zijn. Dus de systempage blijft over. Dit kan het beste opgelost worden door aan het begin van het programma een stuk te plaatsen dat het programma kopieert naar page 0, vanaf ESYS PAGE. Meestal is de plaats van een (relocceerbaar) programma niet bekend als deze vanaf een TOP ingeladen is. Voor het kopiëren is het handig om met het volgende achter de plaats van het programma te komen:

```
CALL GETPC
GETPC POP DE ;terugkeeradres GETPC in DE
```

Een LDIR instructie kan dan het karwei afmaken. Hierna is het niet moeilijk meer om zelf te doen met de video wat je wilt. Een opmerking is nog op zijn plaats. Het pagenummer van de eerste videopage is gegeven in A1/A2, maar de volgende pages hebben achtereenvolgens pagenummers die 1 lager zijn.

We kunnen na lezing van het bovenstaande gerust concluderen dat het allemaal wat lastig gaat, dus alleen toepassen als het niet anders kan, bijv. bij het manipuleren van grote blokken data in het scherm.

Expansion hardware.

Hierover enkele dingetjes.

De centronics uitgang is tevens te gebruiken als user port (poort 3). Bij centronics is handshake mogelijk door een interrupthandling, maar die hoeft niet gebruikt te worden.

De user databus is niets anders dan een bidirectionele databus buffer die een momentopname neemt van de op de connector aanwezige buitenwereld bij een read actie op poort 23 (dec.) en bij een write actie eventjes de byte laat zien met een strobe puls, zodat deze byte indien gewenst naar een 8 bit latch gestuurd kan worden om bewaard te blijven tot de volgende write actie.

Tot slot de opmerking dat een DAC 0808 en een ADC 0809 in de expansion interface thuis hoort.

Tot slot.

De expansion interface is voor eindgebruikers die veel geheugen nodig hebben en/of de diverse extra faciliteiten een welkome aanwinst. Jammer is het dat het systeem als geheel trager wordt als gevolg van de intensieve page manipulaties, terwijl de NewBrain in beeldschermoutput toch al niet zo snel is. Prettig is dat bij CP/M de boot slechts een keer plaats hoeft te vinden en dat CTRL/C overbodig is geworden bij verwisselen van schijven.

Voor programma ontwikkelaars is de expansion interface een last zodra wat specifieke programmatuur gemaakt moet worden, vanwege het complexe systeem.

Rob Maris

Met dank aan Maban Nederland, Hilversum voor het tijdelijk beschikbaar stellen van een expansion interface.

UNPAGE

Onderstaande routines veranderen een paged (expanded) systeem in een unexpanded NewBrain, zodat je er bijvoorbeeld programma's uit de software bibliotheek mee kunt draaien. De eerste is voor een systeem met en de tweede voor een systeem zonder disk

```
1 PUT01:0"UNPAGE 17/8/1984 Tadecom International":RESERVE TOP=18672-65576*(TOP
28672):FORA=TOP+1028798:READ:FOR EAX:NEE:FOR LTOP:DATA205,172,0,14,2,42,149,0,6
,32,237,105,42,151,0,6,84,237,105,46,100,0,120,237,105,36,112,3,188,237,105,35,0
,160,17,0,128,1,0,32,243,237,176,14,2,42,176,0,68,237,105,42,176,0,68,237,105,4
2,180,0,68,237,105,42,147,0,6,0,237,105,35,149,224,17,0,0,1,116,0,243,237,176,33
,254,127,34,0,0,62,24,50,12,0,62,0,50,54,0,50,173,0,49,104,2,33,0,1,54,237,35,54
,105,35,54,201,33,23,160,229,46,104,1,2,98,195,0,1
```

```
1 PUT31:0"UNPAGE 28/8/1984 Tadecom International":0"Versie zonder disk, call 40
960 voor":0"return naar expanded systeem":RESERVE TOP=28672-65576*(TOP<28672):FO
Ra=TOP+1028746:READ:FOR EAX:NEE:FOR LTOP:CALL TOP:DATA205,172,0,14,2,42,180,0,68,237,105
,42,147,0,6,0,237,105,35,149,224,17,0,0,1,116,0,243,237,176,33,254,127,34,6,0,62
,24,50,12,0,62,0,50,54,0,50,173,0,49,104,2,33,0,1,54,237,35,54,105,35,54,201,33,
38,160,229,46,0,1,255,255,195,0,1
```

Een directory leesroutine in machinetaal.

Als je in BASIC werkt op de NewBrain, kun je het door Grundy geleverde programma 'CAT' mergen, zodat je de directory van de schijf kunt lezen.

Dit heeft echter een paar nadelen:

1. Als je je eigen programma saved, moet je eerst 'CAT' deleten.
2. Als je een programma load dan ben je 'CAT' kwijt.

Om van deze problemen verlost te zijn, heb ik een machinetaal routine geschreven.

Deze routine opent device 15 via stream 15 met poort 0, leest de directory uit en stuurt dit naar stream 0. Daarna wordt stream 15 weer geclosed. De uitvoer op het beeldscherm is identiek aan de uitvoer die het 'DIR' kommando levert onder CP/M. (Dit betekent dat voor deze uitvoer dus een d0 karakter scherm geopend moet zijn!)

Het enige wat je nu hoeft te doen om deze routine te gebruiken, is dit BASIC programma laden vanaf de schijf. Daarna geef je het kommando RUN, waarna de instructies in het geheugen worden gepoked te beginnen op adres TOP+1 en het BASIC geheugen wordt schoongemaakt (via de NEW aan het eind van het programma). Via CALL TOP+1 kun je de routine aanroepen.

PROGRAMMA "BDIR"

```
100 DATA 22,0,30,15,62,15,231,50,62,53,231,48,62,13,231,1125
110 DATA 48,33,256,136,231,49,119,231,49,35,54,1,33,256,130,1661
120 DATA 6,6,30,15,126,231,48,35,16,250,33,256,139,6,14,1211
130 DATA 54,32,35,16,251,33,256,140,30,15,231,49,183,254,13,1592
140 DATA 40,13,254,46,32,5,33,256,149,24,240,119,35,24,236,1506
150 DATA 33,256,140,126,183,254,32,32,3,231,52,201,33,256,137,1969
160 DATA 53,32,13,54,4,43,30,0,62,13,231,48,126,231,48,988
170 DATA 35,35,6,15,30,0,126,231,48,35,16,250,30,15,62,934
180 DATA 52,231,48,62,13,231,48,24,167,51,44,42,46,42,13,1114
190 DATA 0,0,38,0,0,0,0,0,0,0,0,0,0,0,0,58
200 RESERVE 160
210 FOR I=TOP+1 TO TOP+150 STEP 15
220 CS=0
230 FOR J=0 TO 14
240 READ D:CS=CS+D
250 IF D<>256 POKE I+J,D:GOTO 280
260 READ D:CS=CS+D
270 A2=INT((TOP+D)/256):A1=TOP+D-A2*256:POKE I+J,A1:J=J+1:POKE I+J,A2
280 NEXT J
290 READ D
300 IF D<>CS?"CHECKSUM ERROR":END
310 NEXT I
320 NEW
```

H. Seinhorst
Hendrikstraat 19
9724 NA Groningen

MAX database

Met een database programma kunt U gegevens op een floppy disk opslaan: adressen, recepten, woordenlijsten, statistieken enz. Een gegevensbestand bestaat uit een aantal velden, in een adressen bestand bijvoorbeeld Naam, Adres, Woonplaats, Telefoon, Beroep, Bijzonderheden.

MAX is geschikt voor de opslag van zowel strings (alfabetische gegevens zoals naam, adres en dergelijke) als getallen.

String bestanden worden op alfabetische volgorde bewaard - U kunt het sorteerveld uiteraard zelf kiezen.

Numerieke gegevens worden op grootte gesorteerd opgeslagen.

MAX is geschreven door Rob van Albada voor de 96k NewBrain met CP/M en wordt gebruikt voor de NBGG ledenadministratie. Dit doet MAX voor U:

-Maakt bestanden tot 8 Megabyte maximaal;

-Sorteert sneller dan elk ander NewBrain BASIC programma:

	MAX	GEMINI (Cassette)	dBaseII
275 records sorteren	15	80	3 min.

taal	BASIC	BASIC	Machinetaal
prijs fl.	100	200	1500
theor.capaciteit	8Mb	24k	8Mb
pract.capaciteit	200k	24k	???

(De praktische capaciteit, d.w.z. de maximale omvang van een bestand, is kleiner dan de theoretische waarde van 8 Megabyte omdat bij zeer grote bestanden de sorteertijd te lang wordt.)

-Kan selectief printen, bijvoorbeeld uitsluitend alle Jansens die NIET in Elst wonen, of alle vrouwen die langer zijn dan 175 cm.

-Kan labels printen d.m.v. speciaal overlayprogramma MAXLAB. Een naam ingevoerd als: Heslinga,Dr. Sicco wordt afgedrukt als Dr. Sicco Heslinga. De gewone printoverlay gebruikt een eenvoudig afbreeksysteem.

-MAX is aangepast aan de STAR printer (EPSON codes) maar kan door de gebruiker gemakkelijk aan andere printers aangepast worden. Desgewenst kunt U adressen krijgen van andere MAX bezitters met dezelfde printer. D.m.v. het hulpprogramma MAX-PATCH kunt U zelfs alle non standard characters van Uw printer gebruiken (bijv. de internationale karaktersets).

-Maximaal 80 velden mogelijk.

-MAX houdt automatisch eenvoudige statistiek bij van de numerieke velden: som, gemiddelde, deviantie, standaard afwijking.

-Opzoeken van gegevens: (a) op sorteerveld d.m.v. Binary Search; (b) op nummer; (c) op willekeurig veld (sequentieel).

-Te koop bij de schrijver op de NewBrain dag in De Bron (26 mei).

PROTO

61

Deze programmers-tool is bruikbaar voor de functie's:

RENUMBER	HERNUMMEREN
FIND	ZOEK
DELETE	VERWIJDER
CHANGE	VERANDER
COMPACT	VERWIJDER SPATIE'S

INLEZEN

=====

Proto 1 bestaat zowel uit machinetaal als basic. Het programma wordt bij het te bewerken programma geladen met behulp van de instructie MERGE.

Om Proto 1 te initialiseren moet men daarna goto 60000 ingeven. Als PROTO 1 als eerste geladen wordt, kan men volstaan door alleen RUN in te geven. Het te bewerken programma moet daarna met MERGE geladen worden.

Als het commando NEW gegeven wordt zal alleen het machinetaal gedeelte voor het hernummeren in de top van het geheugen aanwezig blijven.

HERNUMMEREN

=====

Het hernummerings programma kan men op 2 manieren starten:

1. CALL 31500
2. CALL 31500,p,q,r,s

De eerste manier geeft een hernummering vanaf 100 met stapgrootte 10.

Bij de tweede geeft men 4 variabelen mee (nooit meer of minder !!!) waarbij

p= vanaf regel nr.
q= tot regel nr.
r= naar regel nr.
s= stap grootte

ZOEKEN

=====

GOTO 65000

U kunt na het gebied waarin U zoekt naar bepaalde gegevens, opgeven. Op de vraag "verander in *" geeft U NEWLINE.

VERWIJDEREN

=====

GOTO 65000

Nadat het gebied is opgegeven geeft U op "zoekgegeven" hetgeen U verwijderen wilt in. Op "veranderen in *" moet U het sterretje verwijderen door SHIFT met cursor rechts, daarna NEWLINE.

VERANDEREN

=====

GOTO 65000

In het gegeven gebied kan een zoekgegeven veranderd worden in hetgeen U wenst. Denk er om dat spaties ook gezien worden.

COMPACT

=====

GOTO 65000

Dit is een toepassing van het verander commando waarbij op het zoekgegeven een spatie wordt ingegeven. Bij "veranderen in *" moet U het sterretje weer verwijderen met SHIFT cursor rechts.

=====

DE COMPAS PASCAL COMPILER

Wie op zoek gaat naar een Pascal compiler voor een CP/M computer merkt al gauw dat er een verrassend groot aanbod is. De beschikbare compilers verschillen aanmerkelijk in prijs en in uitvoering.

Enkele namen: JRT Pascal, Turbo-Pascal, Microsoft Pascal, Pascal-Z, Pascal MT+, Pro Pascal, COMPAS Pascal.

JRT Pascal is velen onzer bekend uit de CP/M Public Domain Software bibliotheek - de versie 2.1 is Public Domain; de versies 3 en 4 zijn voor resp. ca. f. 150.00 en f. 200.00 in de handel.

Pascal MT+ is vanaf ca. f. 1500.00 verkrijgbaar; Pascal-Z is eveneens zeer prijzig.

Evenals de prijzen, lopen ook de mogelijkheden zeer uiteen.

JRT Pascal heeft geen eigen editor, zodat men een tekstverwerker (bijv. Wordstar) nodig heeft om het programma mee te schrijven.

Daarmee maakt men een programmatext (source file) met de extensie '.PAS'. Bijvoorbeeld: 'SORT.PAS'. Vervolgens maakt de compiler daar een '.INT' file van (bijv.: 'SORT.INT'), en die INT-file wordt tenslotte uitgevoerd door het execute programma.

Zit er een foutje in het programma, dan moet men eerst weer Wordstar starten om de verbetering aan te kunnen brengen. Nogal omslachtig en tijdrovend dus.

Verder is JRT-Pascal geen volledige Pascal-compiler: een aantal standaard Pascal woorden kent hij niet. De meeste wiskundige functies zijn uitgevoerd als 'external' functies, waardoor het programmeren ook al niet eenvoudiger wordt.

Overigens heeft JRT Pascal wel een aantal interessante extra's, zoals dynamische strings en random access files (geen standaard Pascal).

De duurdere compilers hebben echter meer mogelijkheden, zoals een ingebouwde editor en een complete implementatie van de taal.

Een hiervan is de Deense COMPAS Pascal van Poly-Data, geschreven door de ontwerper van de taal COMAL.

Het is een zeer uitgebreide compiler, met de volgende eigenschappen:

- ingebouwde editor, met Wordstar commando's (je zou er een brief of een korte scriptie mee kunnen schrijven);
- de compiler maakt COM files aan, die direct uitvoerbaar zijn;
- uitgebreide en handige foutmelding door de compiler;
- complete implementatie van Pascal met de o.m. volgende extra's:
- dynamische strings;
- random access files;
- gestructureerde constanten;
- overlay procedures en functies;
- vrije volgorde van de secties van het declaratiegedeelte;
- control codes in string constanten;
- type conversies;
- 'chaining' van programma's met gemeenschappelijke variabelen;
- include files;
- CP/M BDOS en BIOS calls via standaard functies of procedures;
- logische operaties op integers;
- bit/byte manipulatie;

- niet-decimale integer constanten;
- directe toegang tot CPU geheugen en ports;
- variabelen met absolute adres (VAR a: str12 AT \$4000)
- in-line machinetaal opdrachten.

Daarnaast zijn er nog wat andere procedures en functies en opdrachten. CASE..OF is uitgebreid met OTHERWISE.

Om het scherm te editen zijn er zeven standaardstrings zoals CLRHOME en DELLIN. Write(CLRHOME) maakt het scherm schoon en zet de cursor linksboven (als in NewBrain BASIC PUT 31). Write(dellin) wist de regel waar de cursor staat (NewBrain BASIC PUT 30).

De procedure GOTOXY(x,y) zet de cursor op de positie x,y.

Het werken met COMPAS Pascal was na mijn geklungel met JRT een verademing. Als je COMPAS geladen hebt, kom je in het hoofdmenu, die je herkent aan de COMPAS-prompt >>. Het hoofdmenu heeft als opties Load d.w.z. laden van een source file, Save (opslaan van een source file), Dir (directory opvragen), Edit (schrijven en wijzigen van een source file), Zap (wissen van een file), Compile (compileren), Program (compileren en maken van een COM-file), Object (aankijken van een OBJ file) Name (Naam van een file wijzigen) en Help (zet een help text op het scherm).

Als het om een nieuw programma gaat, type je

```
>>e(newline).
```

Hierna kom je in de editor en kun je gaan schrijven.

Je verlaat de editor weer met ^KD (net als in Wordstar).

Terug in het hoofdmenu compileer je het programma met

```
>>c (newline)
```

of met

```
>>r (newline)
```

Met r wordt het programma na compilatie ook meteen uitgevoerd.

Vind COMPAS tijdens het compileren een fout, dan verschijnt er een foutmelding, bijv. Undeclared Identifier.

Druk je hierna op Newline, dan ben je meteen terug in de editor, het programma staat op het scherm en de cursor staat bij de fout, zodat je meteen kunt gaan verbeteren. Dit is een van de prettigste eigenschappen van COMPAS-Pascal. Je tikt even de vergeten puntkomma in, CONTROL/KD, >>c (newline) en klaar is Kees.

Zeker erg handig als je nog niet zo bedreven bent in Pascal en er dus veel fouten uit je programma's gehaald moeten worden. Het compileren gaat trouwens erg snel, ik schat zeker 20 x sneller dan bij JRT Pascal.

BDOS CALLS

Bij de meeste compilers is het wel mogelijk om direct de BDOS (de kern van het CP/M operating system) aan te roepen, maar dat gaat an vaak erg omslachtig, omdat eerst alle Z-80 registers als variabelen gedeclareerd moeten worden, waarna er dan ook nog een stuk assembler geschreven moet worden.

In COMPAS Pascal is dat allemaal niet nodig, omdat deze compiler een opdracht bdos(f,p) kent.

De parameter f is de gewenste BDOS-functie. De parameter p is

afhankelijk van de gekozen functie. Vaak is geen tweede parameter nodig.

Een paar voorbeelden:

```
VAR toets: CHAR;
BEGIN
  toets := bdos(1)
END;
```

Deze procedure leest een karakter van het toetsenbord zonder dat eerst Newline moet worden ingedrukt. Hiermee vervang je de BASIC opdracht `CLOSE#5: OPEN#5,5: GET#5,a`.

```
VAR DMA_adres: INTEGER;
BEGIN
  DMA_adres := $FCB0;
  bdos(26,DMA_adres)
END;
```

Dit is een procedure waarmee het adres van de DMA buffer gewijzigd kan worden (de buffer waarin CP/M gegevens kan neerzetten, zoals records of bestandsnamen).

Een paar andere BDOS calls zijn:

```
bdos(0)      System reset (CONTROL/C)
bdos(2,<integer 0..255>) Stuur karakter naar console.
bdos(13)     Reset disk system
bdos(17,<adres van FCB>) Zoek eerste file in directory
                  (FCB: File Control Block, het adres 5CH)
bdos(18,$005C) Zoek volgende file in directory
bdos(35,$005C) Berekent de grootte van een bestand en
                  zet het resultaat in het File Control
                  Block.
bdos(37,-1)  Reset disk (na verwisselen van disks).
```

Voor veel voorkomende system calls heeft COMPAS Pascal standaard-opdrachten, zoals delete en rename, waarmee bestanden resp. gewist of herbenaemd kunnen worden.

DE NOTATIE

Bij de bovenstaande bdos-calls 18 en 35 zie je als tweede parameter een string die begint met een \$ teken. Dat is een hexadecimal getal (je mag overigens ook de decimale waarde invullen. In COMPAS Pascal mag je Hex- en decimale waarden vrijelijk doorelkaar gebruiken. Bijv. totaal := 43 + \$03FF mag!

Control codes (de ASCII karakters 0 t/m 31):
 Die kun je op maar liefst drie manieren aangeven.
 Ten eerste d.m.v. de standaard Pascal notatie `c := chr(nn)`;
 Ten tweede door het prefix @: `c := @08`;

Ten derde door het prefix ^: `c := ^H`; (d.w.z. CONTROL/H).
 Je mag ze op de volgende manier aaneelkaar plakken:

```
write(1st,@27'@'@27'CH');
```

Met deze opdracht stuur ik de karakters ESC,@,ESC,C en H naar mijn printer, inhoudende de opdrachten 'initialiseer' (ESC @) en 'stel het aantal regels per vel in op 72' (ESC C gevolgd door chr(72) d.i. de H)

TOEGANG TOT HET GEHEUGEN

Pascal kent geen PEEK en POKE opdrachten, maar met de COMPAS Pascal compiler kan het hele geheugen bekeken en veranderd worden d.m.v. het mem array.

```
iobyte := mem[3];
```

Kent de inhoud van geheugenplaats 3 toe aan de variabele iobyte. (In BASIC: `io = peek(3)`).

```
mem[i] := mem[i + 1];
```

Vult geheugenplaats i met de waarde die op adres i+1 staat.

Ook is mogelijk:

```
VAR string_met_vaste_plaats: STRING[255] AT $F000;
```

```
writeln(string_met_vaste_plaats);
string_met_vaste_plaats[64] := 'A';
```

Deze string staat opgeslagen op het adres \$F000
 De tweede regel maakt van het 64e karakter van deze string een A.

BOOLEAN bewerkingen op getallen (BIT ~ manipulatie).

De woorden AND, OR en NOT worden in BASIC gebruikt om voorwaarden te testen:

```
100 IF a=3 AND vr=TRUE THEN k = k + 1
110 IF a=4 OR vr=FALSE THEN k = 0
120 IF NOT(a>2) THEN k = -k
```

In standaard Pascal worden deze woorden net zo gebruikt, alleen is de zinsbouw enigszins verschillend:

```
IF a=3 AND vr=TRUE THEN k := k + 1;    enz.
```

Maar in NewBrain BASIC mag ook dit:

```
200 a = a AND 127
Hiermee worden waarden boven de 127 afgetoet, bijv. 210 wordt 82, omdat het 8e bit met waarde 128 op 0 wordt gezet. (Zie NewBrain tips in on_line 1).
```

In standaard Pascal is echter het gebruik van AND, OR en NOT beperkt tot het bewerken van BOOLEAN variabelen, die alleen de waarden TRUE en FALSE kunnen aannemen. Toepassing op getallen leidt tot een foutmelding.

Omdat het echter heel nuttig is om in bepaalde gevallen elk afzonderlijk BIT-je te kunnen wijzigen, en dat kan d.m.v. deze logische bewerkingen, is deze mogelijkheid in COMPAS Pascal ingevoerd. Dit is dan wel beperkt tot INTEGER variabelen (gehele getallen).

Een voorbeeld:

```
c := chr(ord(c) AND 95);
```

95 is gelijk aan $64 + 16 + 8 + 4 + 2 + 1$.

Het 6e bit, met waarde 32, komt niet in dit rijtje voor en is 0.

Is c nu een kleine letter, bijvoorbeeld 'e', dan is de ASCII waarde, d.w.z. ord(c) in Pascal, hoger dan 95, namelijk 101.

101 is gelijk aan $64 + 32 + 4 + 1$.

De AND opdracht laat alleen de gemeenschappelijke bits van beide getallen over:

```
c := chr(64 + 4 + 1);
```

Omdat $64 + 4 + 1 = 69$ wordt $c = \text{chr}(69)$ en dat is de hoofdletter E.

Onze opdracht maakt dus van kleine letters hoofdletters.

Je kunt BIT manipulatie ook gebruiken om heel veel gegevens in weinig ruimte op te slaan. Als je bijvoorbeeld een vragenlijst hebt met 400 vragen die alleen met JA of NEE beantwoord kunnen worden is het mogelijk elk antwoord in slechts 1 bitje op te slaan. Dat is acht maal zo efficiënt als een byte per antwoord, en alle 400 antwoorden kun je kwijt in een string van 50 karakters.

(Vraag: zijn hier praktische bezwaren tegen? Hoe kan daaraan tegemoet worden gekomen?)

AFWIJINGEN VAN STANDAARD PASCAL.

Dynamische variabelen en pointers gebruiken de new, mark en release procedures i.p.v. new en dispose. Hiermee bereikt men o.m. uitwisselbaarheid met andere compilers zoals UCSD Pascal.

Get en put zijn niet geïmplementeerd. In plaats hiervan gebruikt COMPAS uitgebreide read en write procedures (handiger, sneller en

minder geheugenruimte vragend).

GOTO statements mogen het block niet verlaten.

Procedures en functies mogen niet als parameters gebruikt worden.

De PACK en UNPACK procedures zijn niet geïmplementeerd omdat arrays en strings in COMPAS Pascal altijd PACKED worden opgeslagen (d.w.z. zodanig dat het een minimum aan geheugenruimte kost).

COMPILER OPTIES

In Pascal wordt commentaar (de REM statements van BASIC) tussen accolades gezet: { dit is commentaar }. COMPAS-Pascal biedt bovendien de mogelijkheid om na de een open-accolade zogenaamde compiler opties te zetten.

Enkele voorbeelden:

Na {A+} schrijft de compiler absolute code voor procedures en functies, die sneller is dan relatieve code, maar geen mogelijkheid biedt voor recursie. Normale stand \$A-.

Met {C+} kan het programma onderbroken worden door CONTROL/S of CONTROL/C, met {C-} reageert het programma daar niet op.

Na {I-} worden de run time checks op input en output routines niet uitgevoerd. Standaardinstelling is \$I+.

Na {S+} wordt array indexing geoptimaliseerd met betrekking tot snelheid i.p.v. plaatsruimte. Standaard is andersom: \$S-.

Na {V-} kunnen string variabelen van willekeurige lengte dienst doen als variabele parameters, ook als hun lengte niet overeenkomt met de formele parameter.

GROTE PROGRAMMA'S

Als een programma te lang dreigt te worden voor het beschikbare geheugen, dan kan men gebruikmaken van 'Program chaining' (het programma opdracht geven een ander programma uit te voeren), INCLUDE files, d.w.z. stukje programma die alleen tijdens het compileren in het geheugen komen te staan, OVERLAY PROCEDURES and FUNCTIONS, die alleen worden ingelezen nadat ze zijn opgeroepen en die na uitvoering uit het geheugen worden gegooid, en EXTERNAL procedures en functies in een andere taal dan Pascal, zoals machinetaal.

MACHINETAAL

Het is ook mogelijk om stukjes machinetaal direct in een Pascal programma op te nemen. Het is verleidelijk om hiervan een voorbeeld op te nemen, maar dan zou dit stukje te lang worden. Ik volsta ermee te vertellen dat na het woord CODE de Z-80 commando's als integer opgegeven moeten worden, eventueel gevolgd door parameters of adressen, en gescheiden door komma's.

DOCUMENTATIE

COMPAS Pascal wordt geleverd met een uitvoerige Engelstalige handleiding, bestaande uit een Operating Manual van 25 pagina's en een Programming Manual van 138 pagina's. Dat is nodig ook, want bij een compiler met zoveel mogelijkheden is een duidelijke handleiding bepaald onmisbaar. Uiteraard dient men daarnaast over goede Pascal leerboeken beschikken. De BDOS calls zijn te vinden in diverse CP/M handleidingen, o.m. in *Soul of CP/M* van Mitchell Waite en Robert Lafore, Indiana 1983. Maar op dit punt had de COMPAS handleiding wat mij betreft wel wat uitvoeriger mogen zijn. Met name onbreken voorbeelden van het meegeven van parameters aan CP/M en van het terughalen van strings uit de DMA buffer. Overigens wordt de compiler geleverd met een aantal programma's die zowel als source code als als COM file op de schijf staan. Met name de listing van het INSTALL programma is zeer interessant.

CONCLUSIE

COMPAS Pascal is een compiler die zeer veel mogelijkheden biedt, meer dan de meeste concurrenten. Ik heb er nu enige tijd mee gewerkt en totnutoe deed hij het voorbeeldi. Crashes kwamen niet voor.

Gezien de prijs van vergelijkbare Pascal compilers is COMPAS niet duur: f. 1200,00 ex BTW. Toch geen bedrag om luchthartig uit te geven. Maar voor wie veel zelf programmeert en op BASIC of goedkopere Pascals is uitgekeken zal COMPAS Pascal wellicht een uitkomst zijn.

De Nederlandse importeur Atebo heeft in een brief korting aangeboden voor leden van de NewBrain gebruikersgroep. De hoogte van de korting is afhankelijk van het aantal bestellingen. Om te beginnen krijgt elk NBGG lid 10 pct. korting. Blijken er 10 of meer compilers besteld te worden dan stijgt het kortingspercentage tot 25 pct. Degenen die slechts 10 pct korting gehad hebben krijgen het teveel betaalde teruggestort. Komen er erg veel bestellingen dan kan het kortingspercentage nog verder toenemen en krijgen de vroege kopers weer een deel teruggestort. Bij je bestelling uiteraard wel je lidmaatschap van de NBGG vermelden, alsmede je lidnummer.

Rob van Albada

Importeur van COMPAS Pascal is Atebo in Aalten, tel. 05437-5341.

COMPAS PASCAL

Fast.Compact. Interactive.

For any Z-80 Micro with CP/M.

COMPAS PASCAL nauwgezet de PASCAL definities welke zijn beschreven in het manual "Pascal user manual and report" door Jensen en Wirth. Daarnaast zijn enkele uitbreidingen toegevoegd zodat COMPAS de meest complete implementatie van PASCAL is welke ooit voor microcomputers werd gemaakt.

Naast de zeer uitgebreide COMPAS PASCAL compiler is deze tevens voorzien van een full-screen editor met mogelijkheden tot zoeken, vervangen, blokopertes en vele andere. De besturing van de editor is vrijwel identiek aan de Word-star editor.

Bijgeand de statements en andere zaken van de compiler welke beschikbaar is voor alle computer-systemen met CP/M 2.2 en Z-80 processor en voor systemen met het MS-DOS of PC-DOS operating systeem.

Vraag een demonstratiediskette voor uw NEW BRAIN computer bij ATEBO. Ook voor andere computerzaken kunt u bij ons terecht.

ATEBO
Adviesburo voor automatisering
Hofstraac 14
Postbus 91
7120 AB Aalten
tel. 05437-5341

Statements

BEGIN END	IF THEN ELSE	FOR TO/DOWNTOW DO
WHILE DO	REPEAT UNTIL	CASE OF OTHERWISE
WITH DO	GOTO	CODE

Standard data types

INTEGER	16 bit integers, from -32768 to 32767.
REAL	11 significant digits, from 1E-38 to 1E+38.
BOOLEAN	Logical (true/false)
CHAR	Characters.
BYTE	8 bit integers, from 0 to 255.
Scalar	User defined scalars.
Subrange	With optional range checking
STRING	Dynamic strings.
ARRAY OF	Arrays of any type with multiple dimensions
RECORD END	Records with variants
SET OF	Sets of all simple types.
FILE OF	Random access data files.
TEXT	Textfiles and logical devices.
FILE	Untyped files (direct sector I/O)
Pointer	Dynamic variables.

Declarations

LABEL	CONST	TYPE	VAR	PROCEDURE	FUNCTION
-------	-------	------	-----	-----------	----------

Operators

+	-	*	/	=	<>	>=	<=	>	<
DIV	MOD	AND	OR	EXOR	SHL	SHR	NOT	IN	

Standard procedures

READ	READLN	WRITE	WRITELN	ASSIGN
RESET	REWRITE	CLOSE	ERASE	RENAME
SEEK	BLOCKREAD	BLOCKWRITE	EXECUTE	CHAIN
DELETE	INSERT	STR	VAL	NEW
MARK	RELEASE	RANDOMIZE	GOTOXY	CLREOS
CLREOL	MOVE	FILL	BDOS	BIOS

Standard functions

EOF	EOLN	LENGTH	POSITION	ABS
SQR	SQRT	SIN	COS	ARCTAN
LN	EXP	INT	FRAC	SUCC
PRED	ODD	TRUNC	ROUND	ORD
CHR	LEN	POS	COPY	CONCAT
PTR	LO	HI	SWAP	RANDOM
PWRTRN	KEYPRESS	MEMAVAIL	IORES	ADDR
SIZE	BDOS	BIOS	BDOSB	BIOSB

Toelichting bij gebruikersvriendelijke basicode

Omdat ik in de loop der tijd een aantal BASICODE programma's had opgenomen, en deze achtereen wilde vertalen en opslaan, zat ik te broeden op een praktisch programma. Nu moest ik eerst maar laden, wegschrijven, herladen, subroutines toevoegen en weer wegschrijven met een korrekte titel. Voorlopig heb ik een redelijke oplossing gevonden.

De subroutines zijn door mij ingekort tot zo weinig mogelijk regels, waarbij de REM's zijn weggelaten. Andere regels zijn samengetrokken. Verder heb ik het ingeven van de titel van het programma naar achteren verschoven. Nadat het te vertalen BASICODE programma al dan niet juist is ingelezen, volgt een korte listing (±5 regels) op het scherm, voldoende om de titel er uit te kunnen halen. Hierna wordt de naam van het weg te schrijven programma gevraagd. (kortere programma's kunnen ook met b.v. "20:prog.naam" worden weggeschreven, om de blokken te vergroten en inleesfouten te beperken)

Nadat de titel is ingevoerd, wordt gevraagd, of de listing tijdens het wegschrijven getoond moet worden. Ja, Nee of via Printer. Is het de p van printer, dan wordt nog aangegeven dat de regellengte zoveel tekens lang is als het getal bij PEEK(617). D.m.v. een POKE 617,... instructie tussen te voegen is deze regellengte in te korten of te verlengen.

Bij het wegschrijven, worden eerst de subroutines weggeschreven, als onderdeel van het nieuwe vertaalde programma. Dit gebeurt met LIST 12,20-360. Dan volgt de rest van het vertaalde programma, met evt. tegelijkertijd de listing op scherm of printer. Hierna kan het volgende programma geladen worden met RUN.

toelichting bij MOGELIJKHEDEN BIJ INPUT

Het programma zelf zegt niets. Het is alleen illustratief. De onderstreepte INPUT-statements geven de verschillend gebruikte prompt-statements. Ook hier is d.m.v. chr\$() zelf de plek aan te wijzen waar de input op het scherm geprint wordt, soms geïnverteerd. Probeer de uitwerking maar eens uit, door bij NUMER eerst een woord in te geven, of bij de input een " of een ,. Bij het een volgt een herhaling eronder, bij de ander springt de vraag naar de aangeduide plek terug. De invoer van de tekens in regel 95 kan b.v. met ESCAPE en ONIL/V voor teken 22. Vergelijk de uitwerking op het scherm met de listing. Veel succes ermee.

Jan Tolkamp,
Blauwe Hof 56-02,
6602 XA Wychen,
tel 08994-11786

P.S. Wie kan mij wat wegwijs maken met
de WERKING van de machinetaal, zoals
b.v. bij het BASICODE-leesprogramma?

mogelijkheden bij INPUT

```
20 PUT23,68,31,14:?"DIT KAN MET INPUT":PUT15:?  
30 ?"WAT IS JE VOORNAAM":;INPUT VS  
40 INPUT(VS$;" WIE ? ")NS  
50 PUT31:B=4:?"STEL ";VS;" ";NS; " HEEFT";B;"BUREN"  
60 FOR I=1 TO B  
70 INPUT(CHR$(22)+CHR$(1)+CHR$(1*2)+CHR$(14)+STR$(I 1)+CHR$(15)+?"E BUURMAN VAN  
  "+VS$;" HEET: ")BS(I)  
80 NEXT I  
85 INPUT(".... WELK NUMMER HEEFT "+BS(2)+?" . ")X  
90 INPUT(CHR$(22)+CHR$(1)+CHR$(21)+CHR$(14)+?" WELK NUMMER HEEFT "+BS(2)+?" +CHR$(  
  (15)+?" ")X:REM deze regel is listbaar  
95 INPUT("..... WELK NUMMER HEEFT "+BS(2)+?" ■ ")X:END:REM deze regel is NIET norm  
aal listbaar
```

```
1 REM  GEbruikersvriendelijke BASICODE SUB+LEESROUTINE  
2 REM  (DATA REGELS 10000 T/M 10130 MOETEN VAN DE VORIGE NOG OVERGENOMEN WORDEN)
```

SUBROUTINES BASICODE-2

```
10 GOTO 1000  
20 FOR I=1 TO 255:CLOSE I:NEXT I:BS$="9600":OPEN E,8,BS:OPEN E,0:OPEN E,5:OPEN E,6:RANDOMIZE:  
  GOTO 1010  
100 PUT31,12:RET  
110 PUT22,H0,VE:RET  
120 PUT21:GET H0,VE:RET  
200 PUT6:GET E6,INS:PUT7:RET  
210 PUT6:GET E5,INS:PUT7:RET  
250 PUT 23,49,0,0,0,0,0,0,0,0,0,0,0,23,48:PUT E8,7:RET  
260 RV=RND:RET  
270 FR=FREE:RET  
300 SR$=STR$(SR):SR$=MID$(SR$,2,LEN(SR$)-2):RETURN  
310 SR=INT(SR*(10*CN)+.5)*10^-CN:GOSUB 300:IF CN=0 GOTO 320  
316 IF INSTR(SR$,".")=0 THEN SR$=SR$+"."  
318 SR$=SR$+"000000000":SR$=LEFT$(SR$,LEN(SR$)-((LEN(SR$)-INSTR(SR$,".))-CN))  
320 SR$=RIGHT$(SR$," "+SR$,CT):RET  
350 ?E8,SR$:RET  
360 ?E8:RET
```

```
1000 CT=52:EN=11776:T=32*1024:DV=PEEK(19)  
1040 FOR I=1 TO 255:CLOSE I:NEXT I:OPEN E,8:OPEN E,0:OPEN E,5  
1050 PUT31:?:?:?:?"Lees basicode NewBrain release 1.1":?  
1060 RESERVE 1:RESERVE TOP+2*15:RESERVE TOP-EN:REM geeft na BREAK/RUN voldoende reserve  
1070 RESTORE:FOR I=EN TO EN+223:READ A:POKE I,A:NEXT I  
1110 POKE EN+163,CT  
1120 C=PEEK(EN+3)+PEEK(EN+4)*256-1  
1130 POKE C+10,2: POKE C+11,3  
1140 ?:"Start cassette met BASICODE-prog. (N/L)":;GET A  
1160 ON ERROR GOTO 1180: ? : CALL EN  
1170 ?:"FOUT ?????":END  
1180 A=ERRNO: ON ERROR GOTO 0:PUT31  
1190 ON A GOTO 1400,1200,1210,1220,1230  
1200 ?:"Checksum klopt": GOTO 1240  
1210 ?:"Checksum klopt niet": GOTO 1240  
1220 ?:"Inlezen afgebroken": GOTO 1240  
1230 ?:"Time out error": GOTO 1240  
1240 REM Schrijf Basicode programma naar default uitvoer  
1250 ? : b=c  
1260 b=b+1:?:PUT28:a=PEEK(b): IF a=02 AND b=0 THEN GOTO 1260  
1270 IF b=2 THEN ?:"Geen start tekst gevonden": b=c  
1280 FOR I=12030 TO 12250:PUTPEEK(I):NEXT I:PUT30:REM eerste progr.regels zichtbaar maken  
1290 ??:LINPUT("Naam weg te schrijven programma: ")A$: IF DV=12 THEN IF A$="" THEN 1290  
1300 PUT31:?"moet listing zichtbaar zijn? (j/n/p)":;GET E5,nl$:nl$:IF nl$="p" THEN NL=8:?"zon  
  odig printer op":PEEK(617):"tekens zetten":(N/L):GET E5,pr$  
1310 IF DV=1 OR DV=2 THEN ?:"Start cassette en (N/L)":;GET A  
1320 CLOSE I2: OPENOUT I2,dv,a$  
1325 LIST I2,10-360:REM als eerste de subroutines wegschrijven  
1330 IF nl$="n" THEN 1370  
1340 b=b+1: IF b=2 THEN ?:"Geen einde tekst": GOTO 1360  
1350 a=PEEK(b): IF a=03 THEN PUT E12,a: PUT E,x,a: GOTO 1340  
1360 GOTO 1390  
1370 b=b+1: IF b=2 THEN ?:"Geen einde tekst": GOTO 1390  
1380 a=PEEK(b): IF a=03 THEN PUT E12,a: GOTO 1370  
1390 PUT E12,4,13:CLOSE I2  
1400 ??:a$:?" is weggeschreven":RESERVE 2*15:TOP:END
```

FORTH ,een programma omgeving

Ik wil proberen om aan u wat over de taal FORTH te vertellen iedereen heeft het tegenwoordig over ADA & PASCAL maar een taal die minstens zo mooi is (zo niet krachtiger) is de taal FORTH. Eigenlijk is zoals door de titel al wordt aangegeven FORTH geen taal maar een programmeer omgeving het is namelijk ;

- 1.. Hogere Programmertaal
- 2.. In-line assembler (machinetaal)
- 3.. Operating systeem (hier is dit CP/M)
- 4.. Woordenboek (systeem woorden)

FORTH is geschreven door Charles H. Moore. hij gebruikte het om telescopen te besturen en om op het systeem de financiële administratie bij te houden van het observatorium. Het eerste systeem waar FORTH op liep was een IBM 1130. Charles H. Moore vond dit een krachtige taal en noemde dit een vierde generatie programmeer taal en wilde het FOURTH noemen (vierde dus) maar het operating systeem van de IBM accepteerde maar vijf letters en dus moest hij de naam in FORTH veranderen. Met deze naam gaat deze taal de wereld rond en vindt steeds meer aanhangers. Ook is het mogelijk dat een applicatie (bij FORTH spreekt men niet van een programma maar van een applicatie) welke op een systeem geschreven is met een Z-80 zonder problemen op een FORTH met een 6502 draait. Er is wel een adder onder het gras en wel dat er geen directe machinetaal in voor komt. Men kan FORTH uitstekend gebruiken voor besturings doeleinden ook voor het schrijven van spelletjes is FORTH zeer geschikt omdat het maar 10% langzamer dan machinetaal is maar veel dichter bij de mens staat en dus voor deze veel eenvoudiger te begrijpen is. In de meeste video-games machines welke in de video-hallen staan wordt FORTH gebruikt om de gehele machine te besturen inclusief de geld-automaat.

Een eerste introductie welke ik U van FORTH wil geven is een voorbeeld hoe men een berekening maakt in FORTH;

FORTH is een STACK georiënteerde taal een stack is een soort kladblaadje voor getallen. Een paar voorbeelden Om iets uit te rekenen moet men eerst de getallen op de stack zetten. Dit doen we door het getal gewoon in te tikken en dan een spatie of een 'new-line' te geven FORTH meldt zich terug met Ok. Nadat we de 'new-line' hebben gegeven. Stel we voeren het volgende in;

12 34 56 78 90'new-line' (let op de spaties tussen de cijfers)

Als men nu een . (punt) in tikt krijgt men 168 op het scherm gevolgd door een Ok. Dit is de berekening van $78 + 90 = 168$ maar dan op de volgende manier 78 en 90 staan op de stack door nu de plus te geven worden deze twee bij elkaar opgeteld en staat er 168 op de stack (78 en 90 zijn verdwenen) de punt wil zeggen druk af op het beeldscherm wat boven op de stack staat. Dat was in dit geval 168. Ook kunnen we andere berekeningen uitvoeren als men maar rekening houdt dat men met integers werkt (gehele getallen tussen 0 en 32768 negatief is ook mogelijk) vooral als men gaat delen kan men nog wel eens vreemde uitkomsten tegen komen. Want bij / (delen) wordt alleen het quotient achter gelaten eventuele rest is verdwenen maar FORTH zou FORTH niet zijn als daar geen oplossing voor was die is er in de vorm van /MOD deze laat op de stack een remainder (rest) en het quotient achter in deze volgorde dus het quotient ligt bovenop. Op deze manier kan er altijd achter komen of de berekening volledig is. Verdere berekenings operatoren zijn;

- +.. Optellen.
- .. Aftrekken.
- *.. Vermenigvuldigen
- /.. Delen.
- /MOD.. Delen met een rest.
- MOD.. Laat alleen de rest achter.

Buiten deze operatoren zijn er nog andere stack operatoren namelijk de stack manipulators deze verrichten handelingen met waarden die op de stack staan. Op de stack staat nog steeds de waarden;

12 34 56 (78 90 en 168 zijn in handelingen + en . opgegaan)

stel dat we 56 willen delen door 34 (als we nu zouden delen delen we 34 door 56). Hier is de instructie SWAP voor, deze wisselt de twee bovenste stack elementen, van plaats. We typen dus SWAP'new-line' en FORTH meldt zich met Ok nu is de stack als volgt;

12 56 34

Als men nu /MOD in typt wordt 56 door 34 gedeelt en er blijft een rest plus een quotient achter welke op de stack staan;

12 22 1 (1 = uitkomst en 22 is de rest)

Er zijn nog meer stack manipulators maar om die allemaal te behandelen gaat een beetje te ver voor deze introductie. Laten we een aplikatie gaan maken. Om een nieuw FORTH woord aan het woordenboek toe te voegen is hier een vaste procedure voor door het nieuwe woord met een : (dubbele punt) te beginnen en te eindigen met een ; (punt komma) wat ook nog zeer belangrijk is dat als scheidings teken gewoon de spatie gebruikt wordt (net zoals bij de nederlandse taal). FORTH heeft een grote overeenkomst met de taal (engels of nederlands) omdat onze spreektaal ook zich uitbreidt met woorden (neem maar het woord microprocessor dat bestond niet voor 1970).

Dus we beginnen met een : (dubbele punt) en eindigen met een ; (punt komma) Stel we willen een asteriks (*) op het beeldscherm krijgen als we het woord 'STER' gebruiken in een woord. We zullen dus eerst het woord 'STER' moeten maken. Dat doen we door het woord 'EMIT' wat wil zeggen druk in ascii af het bovenste stack element. Het woord 'STER' maken we als volgt;

```
: STER 42 EMIT ;'new-line' Ok
```

De Ok achter de 'new-line' wil zeggen dat FORTH dit nu als een geldig woord ziet en IEDERE keer als we nu 'STER' intypen komt er een asteriks (*) op het scherm. Dus als volgt;

```
STER'new-line' * Ok
```

Let er op dat we tussen 'STER' en 'new-line' geen spatie hoeft (mag wel). En dat het onderstreepte door ons is ingevuld en de spatie gevold door de asteriks spatie en Ok door FORTH wordt verzorgd. Wat is hier nu de zin van zult U denken wel stel dat U een aantal sterren bijv. 10 stuks op het scherm wilt hebben. Nu zou U een nieuw woord kunnen maken waar achter elkaar 10 maal het woord 'STER' in zou voorkomen maar er is een betere oplossing voor dit probleem namelijk met een lus instructie welke vergelijkbaar is met de FOR NEXT loop in BASIC

We wilden 10 sterren op het scherm laten we het woord 'STERREN' (meervoud van STER) gebruiken;

```
: STERREN 10 0 DO STER LOOP ;'new-line' Ok
```

Hoe werkt dit? Wel heel eenvoudig de 10 en de 0 worden op een tweede stack de RETURN stack gezet en er wordt gewoon het woord of woorden welke achter DO staan uitgevoerd daarna komt men bij LOOP welke gewoon de uitvoer van de aplikatie bij DO weer laat beginnen DO verhoogd de 0 met 1 en wordt dus 1 enz. Tot 10 en hierna wordt de aplikatie STERREN verlaten omdat er niets meer uit te voeren is. Typt U maar

```
STERREN'new-line' ***** Ok
```

Ook kunt U deze aplikatie vergeten, uit het woordenboek halen door de instructie FORGET als U nu FORGET STER intypt dan meldt FORTH zich weer met Ok en bent U deze aplikatie kwijt dus heeft U weer een schone lei om zo te spreken.

Ik hoop dat ik U iets van FORTH heb kunnen laten zien. Op de aanstaande gebruikersdag de 22 sept 1984 zal ik met een werkend FORTH systeem aanwezig zijn. Heeft U dus vragen of wilt U meer zien van de ongeveer 200 instructies dan kan U dus bij mij terecht.

Jan J. Vooijs,
p/a v. Aalstlaan 544
2722 SC ZOETERMEER.

Lourens,	1000 BR AMSTERDAM
Bakker, Klaas f.	1011 AZ AMSTERDAM
Bleeker, W.K.	1012 LI AMSTERDAM
Letn, P.J.A. van	1013 TL AMSTERDAM
Foeken, Paul	1015 EA AMSTERDAM
Boerboom, Simon K.	1015 HR AMSTERDAM
Kalwij, Jan	1016 VS AMSTERDAM
Cocu, Paul J.	1024 NL AMSTERDAM
Techninc Europe	1025 RX AMSTERDAM
Dudshoorn, W.	1034 LL AMSTERDAM
Technosure	1034 LL AMSTERDAM
Heeren, Frans	1053 NX AMSTERDAM
Beukelaar, M.J.	1054 LS AMSTERDAM
Ruth, Tj. van	1056 AS AMSTERDAM
Degens, B.	1056 XB AMSTERDAM
Scholengemeenschap, Osdorpe	1068 MR AMSTERDAM
Beima, Bas	1072 EL AMSTERDAM
Scull, John	1073 HG AMSTERDAM
Albada, Rob var.	1073 JX AMSTERDAM
HCC NewBrain G6, Secr.	1073 JX AMSTERDAM
Oss, P. van	1078 GB AMSTERDAM
Werker, F.	1078 GL AMSTERDAM
Ridder, O.W. de	1078 VV AMSTERDAM
Remeus, M.	1079 SR AMSTERDAM
Blanken, R.R.	1082 LE AMSTERDAM
Stevens, H.C.C. (Menno)	1092 JV AMSTERDAM
Vishvas, S.V.	1093 TP AMSTERDAM
Roskam Abbing, Paul	1094 J6 AMSTERDAM
Bekers, Jan	1094 JH AMSTERDAM
Krol, Jodi	1095 AB AMSTERDAM
Pelikaan, Kees	1098 CK AMSTERDAM
Hoek, W. A. van	1098 KL AMSTERDAM
Piets, Theo	1102 CA AMSTERDAM
Lohstro, C.L.	1104 AK AMSTERDAM
Bruin, J.W.	1106 GE AMSTERDAM 20
Elfferich, A.	1121 CP LANDSMEER
Woude, Hidde v.d.	1121 CP LANDSMEER
Rijnbeek, Els	1141 JA MONNICKENDAM
Rijnbeek, Bob	1141 RM MONNICKENDAM
Bouma, D.	1151 ER BROEK IN WATERLAND
Corba, D.	1181 BZ AMSTELVEEN
Nieuwenhuis, W.H.M.	1182 HM AMSTELVEEN
Murachawed, M.T.	1183 AC AMSTELVEEN
Huwalda, P.	1185 TG AMSTELVEEN
Gerszt, S.	1186 EJ AMSTELVEEN
Meyer, Pieter	1211 EJ HILVERSUM
Hekman, A.	1211 KX HILVERSUM
Wielwaker, C.	1214 CP HILVERSUM
Valster, Cornelis Andries	1215 RLHILVERSUM
Hageman, J.	1216 AA HILVERSUM
Rhemmen, B. van	1217 TE HILVERSUM
Rutten, G.W.H.	1221 EE HILVERSUM
Houten, W. van	1222 RD HILVERSUM
Morgen, A. von	1231 AP LOOSDRECHT

Harmelen, M.W.J. van
 Laere, G.A. van
 Hilhorst, C.
 Sniijders, C.
 Hiddink, H.
 Zomer, G.
 Uylkert, Tom
 Dijk, R.E. van
 Steunebrink, Klaas
 Boterenbrood, P.
 Leeuw, A. de
 Ballast, M.H.J.
 Dorenbos, Karne H.
 Ditzhuyzen, G.L.G. van
 Vredenburg, M.S.
 Persbureau Koolhoven
 Oosterink, M.
 Ditzel, J.C.L.
 Osborn, Paul
 Raij, Wie van
 Eggermont, F.J.
 Vries, F.R. de
 Stolp, J.J.
 Dennessman, G.
 Duivenbode, K. van
 Zanen, J. van
 Berkel, H. van
 Duyvenbode, C. van
 Verdon, H.
 Kok, A.S.P.
 Wouterson, R.G.
 Assema, A. van
 Nieuwenhuizen, R.
 Floeg, J. van der
 Kanon, J.H.
 School Den Helder, Alg. Techn
 Koolstra, R.H.
 Lageman, A.J.J.
 Paassen, N.C.J. van
 Kalis, D.
 Leeuw, L. de
 Borst, A.L.M.
 Knippels, H.P.A.M.
 Hof, R.
 Bakker, G.J.C.
 Reinking, P.H.W.
 Bree, L.H.F.C. van
 Akerboom, P.
 Heyden, E.v.d.
 Abbas, H.H.
 Zeegers, W.
 Koning, E.
 Technisch Burp DELTA-T
 Hoogland, C.
 Koning, J.P.

1241 LV KORTENHOEF
 1241 XE KORTENHOEF
 1243 KC 's-GRAVELAND
 1273 AP HUIZEN
 1273 XJ HUIZEN
 1315 CL ALMERE
 1353 CH ALMERE
 1354 GP ALMERE
 1381 AJ WEESP
 1381 EE WEESP
 1382 JX WEESP
 1399 GK MUIDERBERG
 1401 BM BUSSUM
 1402 CG BUSSUM
 1402 BA BUSSUM
 1412 AN NAARDEN
 1412 GK NAARDEN
 1422 EJ UITHOORN
 1422 JS UITHOORN
 1442 NG PURMEREND
 1444 VN PURMEREND
 1484 PC GRAFT
 1504 AT ZAANDAM
 1505 WP ZAANDAM
 1509 BS AMSTERDAM
 1509 ZAANDAM
 1541 GV KODG a/d ZAAW
 1562 TP KROMMENIE
 1628 HK HOORN NH
 1688 CW NIBBIXMOUD
 1703 KB HEERHUGOWAARD
 1711 RT HENSBOEK
 1779 EL DEN DEVER
 1782 AT DEN HELDER
 1782 GD DEN HELDER
 1782 ND DEN HELDER
 1784 AG DEN HELDER
 1823 EV ALKMAAR
 1829 HJ OUDORP NH
 1834 AR ST.PANCRAS
 1852 KV HEIJLOO
 1861 VB BERGEN N-H
 1871 EV SCHOORL
 1906 AX LIMMEN
 1921 BW AKERSLOOT
 1962 SG HEEMSKERK
 1931963 AV HEEMSKERK
 2025 ZD HAARLEM
 2032 ZN HAARLEM
 2034 BW HAARLEM
 2034 SP HAARLEM
 2102 BE HEENSTEDE
 2105 SK HEENSTEDE
 2105 XV HEENSTEDE
 2161 RV LISSE

Jona, R.A.F.
 Iwants, J.M.S.
 De Spinbaan
 Dijk, Pieter W.M. van
 Holst, F.B.v.d.
 Pellislaan, H.
 Bos, W.A.M.v.d.
 Robus, H.W.
 Legen, M.
 Ruizenga, T.
 Jans, H.A. van
 Kouwenhoven, B.J.M.
 Rijckevorsel, M. van
 Peeter, C.A.v.d.
 Thiele, T.
 Pelt, J.J.M. van
 Radijk, A.
 Lange, E.L. de
 Attema, S.
 Ouboen, Charles L.
 Idelenburg, J.
 Wilms, G.M.T.
 Leenes, R.E.
 Toeker, J.H.
 Nierop, B. van
 Hendriks, C.A.
 Loon, R.D. van
 Lagendijk, C.L.W.
 Kops, D.J.
 Floop, Arend
 Vreugd, R. de
 Pol, P. van der
 Heumen, T.L. van
 Haags Montessorilyceum
 Meij, v.d.L.
 Reuvers, D.C.
 Bartelds, E.
 Meyer, M.C.
 Davis, P.
 Lievaart, Hans
 Regt, A.C.J. de
 Valkenburg, Menno
 Voors, J.J.
 Meyer, F.V.H.
 Hoog, C. de
 Hermans, P.P.M.T.
 Glass, A.
 Iwaan, A.G.
 Das, P.
 Erkelens, G.
 Neut, P.v.d.
 Pillay, M.
 Caron,
 Verbeek, M.B.
 Bothof, B.

2181 XE HILLEGOM
 2201 BN NOORDWIJK Z.H.
 2240 AH WASSENAR
 1241 SB WASSENAR
 2242 XP WASSENAR
 2266 ES LEIDSCHENDAM
 2273 SH VOORBURG
 2274 LH VOORBURG
 2282 HL RIJSWIJK ZH
 2285 HK RIJSWIJK ZH
 2295 LH KWINTSHEUL
 2295 LX KWINTSHEUL
 2311 GL LEIDEN
 2312 GP LEIDEN
 2316 AC LEIDEN
 2324 EA LEIDEN
 2341 HN OEGSTGEEST
 2352 JB LEIDERDORP
 2391 VH HAZERSWOUDE DORP
 2396 CX KOUDEKERK a/d RIJN
 2401 FA ALPHEN A/D RIJN
 2402 ALPHEN A/D RIJN
 2403 JK ALPHEN A/D RIJN
 2403 PD ALPHEN A/D RIJN
 2411 BS BODEGRAVEN
 2481 BA WOUBRUGGE
 2555 KH 's-GRAVENHAGE
 2563 BZ 's-GRAVENHAGE
 2574 SD 's-GRAVENHAGE
 2582 LN 's-GRAVENHAGE
 2582 XK 's-GRAVENHAGE
 2586 TB 's-GRAVENHAGE
 2587 ED DEN HAAG
 2596 AK 's-GRAVENHAGE
 2613 BM DELFT
 2624 LM DELFT
 2624 GH DELFT
 2624 LC DELFT
 2624 LG DELFT
 2631 DX 's-GRAVENZANDE
 2700 AE ZOETERMEER
 2713 CE ZOETERMEER
 2722 SC ZOETERMEER
 2726 ZOETERMEER
 2741 XR WADDINXVEEN
 2771 CC BOSKOOP
 2806 DA GOUDA
 2811 AN REEUNIK
 2905 AT CAPELLE A/D IJSSEL
 2951 BW ALBLASSERDAM
 2982 VN RIDDERKERK
 3008 AE ROTTERDAM
 3061 AR ROTTERDAM
 3061 GN ROTTERDAM
 3076 DL ROTTERDAM

Hello, Giovanni
 Boender, M.
 Vrijland, L.
 Bont, N.J.J.M. de
 Schuddebeurs, P.
 Grootveld, W.J.
 Hart, H. 't
 Jongma, S.
 Vermeul, M.S.
 Menink, P.
 Kerklaan, J.
 Bakelaar, B.
 Jongma, S.
 Berg, A.v.d.
 Veen, L. van
 Pinto, M.
 Naburgh, E.P.
 Spall, R.M.P.
 Zwanenburg, P.
 Scholten, J.C.J.M.
 Truijens, Kees L.
 Brinkers, G.
 Hussen, A. van
 Rijke, D. de
 Moor, C.H. de
 HCC NewBrain GG, Voorz.
 Vries, Jan de
 Kreuger, T.
 Bode, A.W.D.
 Born, Y.T.v.d.
 Voorn, A.G.
 Engelsman, John den
 Jansen, C.D.
 Koffeman, I.
 Regteren, J. van
 Jansen, J.
 Born, M.
 Berkhof, C.
 Bruggen, J.v.d.
 Verbraeken, R.
 Eekelen, J.J. van
 Wijnoltz, Fred
 Ouden, J. den
 Erve, H.J.v.d.
 Olphen, A.F. van
 Kamperman, Th.M.
 Unlandt, A.
 Hendriks, P.J.M.
 Sluijs, Jan
 Hoogenhuyzen, P.
 Alba Heijdenrijk, E.
 Roos, E.
 Boesburg, E.J. van
 Hoegaerden, V. van
 Weststrate, P.W.

31031 BIGOLINO Tv. (ITALIE)
 3116 AG SCHIEDAM
 3116 AJ SCHIEDAM
 3116 BK SCHIEDAM
 3133 KR VLAARDINGEN
 3134 EB VLAARDINGEN
 3145 TL MAASSLUIS
 3171 NB POORTUGAAL
 3174 EN POORTUGAAL
 3203 BG SPIJKENISSE
 3206 PR SPIJKENISSE
 3257 LJ OOLTGENSEPLAAT
 322 BP OUD BEVERLAND
 3295 LE 's-GRAVENDEEL
 3295 VC 's-GRAVENDEEL
 3314 HH DORDRECHT
 3319 VL DORDRECHT
 3431 SK NIEUWEGEIN
 3435 BH NIEUWEGEIN
 3511 VO UTRECHT
 3512 NP UTRECHT
 3514 EJ UTRECHT
 3562 GH UTRECHT
 3571 SL UTRECHT
 3572 RH UTRECHT
 3581 SP UTRECHT
 3581 SX UTRECHT
 3603 GA MAARSSEEN
 3605 HJ MAARSSEEN
 3607 DE MAARSSENBROEK
 3641 AK MIJDRECHT
 3703 GB ZEIST
 3705 HW ZEIST
 3712 AR HUIS TER HEIDE UT
 3721 BB BILTHOVEN
 3732 ES DE BILT
 3742 EN BAARN
 3766 EG SOEST
 3811 AM AMERSFOORT
 3811 LR AMERSFOORT
 3815 VS AMERSFOORT
 3828 VC MOOGLAND
 3844 MK HARDERWIJK
 3925 RX SCHERPENZEEL GLD
 3962 TB WIJK BIJ DUURSTED
 3972 V8 DRIEBERGEN RIJSENB.
 3972 XB DRIEBERGEN RIJSENBURG
 4051 EX OCHTEN
 4121 BC VIANEN ZH
 4142 AG LEERDAM
 4201 BG GORINCHEM
 4206 AG GORINCHEM
 4254 STEENWIJK
 4285 CB WOUDRICHEM
 4333 EN MIDDELBURG

Mouwrik, Aart S. van
 Veltema, J.
 Stroo, Reeko
 Luijk, J. van
 Koppelaar, R.V.
 Kampen, I. van
 Lindeboom, H.
 Roo, F. de
 Milliano, Etienne de
 Schuurs, J.
 Admiraal, J.H.M.
 Velde, N.D.v.d.
 Wielen, E.G.M. van der
 Luijt, W.M.
 Eekelen, J. van
 Kanter, A.A.M. de
 Dekker, L.
 Elias, G.
 Dekker, H.G.
 Aarts, M.J.
 Tilburg, H.W. van
 Wiggerde, K.
 Kleyn Hesselink, Jan G.C.
 Agtmaal, E.A.J. van
 Hermans, Jos
 Bont, R.J.M.M. de
 Weijters, A.J.M.M.
 Haans, A.C.M.
 Brock, J.
 Wijk, H. van
 Blom, P.
 Velden, M.F.J.A.v.d.
 Alisic, Slobodan
 Vercammen, L.J.
 Mimpes, T.J.C.M.
 Wiersma, G.J.
 Linden, Hans (J.G.J.) v.d.
 Vossenbergh, M.W.F.M.v.d.
 Berkemaans, R.
 Haaren, R. van
 Maasland Coll. tav F.vd Velde
 Velden, F.W.J. v.d.
 Tonies, Ton
 Kruysbergen, J.W.J. van
 Beerensteyn, A.L.
 Neys, Ch.
 Haanstra, H.B.
 Jansen, P.M.
 Hurk, Theo v.d.
 Welten, H.Th.
 Lemmen, G.
 Pronk, C.M.
 Ham, W.A.
 Russel, A.J.M.
 Heijnen, M.

4337 AD MIDDELBURG
 4384 BP VLISSINGEN
 4384 LC VLISSINGEN
 4386 CC VLISSINGEN
 4388 HW OOST SOUBURG
 4414 RS WAARDE
 4431 CK 's-GRAVENPOLDER
 4431 DO 's-GRAVENPOLDER
 4453 BA 's-HEERENHOEK
 4463 CJ GOES
 4463 VZ GOES
 4475 AG WILHELMINADORP
 4561 AC HULS
 463 VM GOES
 4726 AE HEERLE
 4816 CM BREDA
 4818 BC BREDA
 4841 LB PRINSENBEEK
 4847 SR TETERINGEN
 4901 AV OOSTERHOUT NB
 4902 AE OOSTERHOUT (N-B)
 4905 AX OOSTERHOUT
 4907 HC OOSTERHOUT
 4907 ZP OOSTERHOUT NB
 5011 SM TILBURG
 5012 BT TILBURG
 5014 BL TILBURG
 5032 BB TILBURG
 5037 DG TILBURG
 5042 BH TILBURG
 5044 ND TILBURG
 5051 BT GOIRLE
 5109 TZ 's-GRAVENMOER
 5144 CX MAALWIJK
 5151 GJ DRUNEN
 5161 GH SPRANGCAPELLE
 5171 DG KAATSHOEVEL
 5212 FH 's-HERTOGENBOSCH
 5271 NK ST. MICHIËLSGESTEL
 5301 AD ZALTBOMMEL
 5343 XK OSS
 5343 XK OSS
 5368 AJ HAREN NB
 5434 SZ GUYK
 5463 NE VEGHEL N.B.
 5473 GZ HEESWIJK DINTHER
 5550 AB VALKENSWAARD
 5641 MK EINDHOVEN
 5703 XR HELMOND
 5708 JD HELMOND
 5712 BR SOMEREN
 5737 EG LIESHOUT
 5752 BE DEURNE
 5802 AD VENRAY
 5804 AA VENRAY

Kurstjens, Jan
 Steinmann, H.
 Sonnemans, A.J.F.
 Schoeber, M.
 Muller, E.G.
 Arieaans, W.H.
 Janssen, N.
 Dassen, F.J.A.M.
 Neertens, A.
 Prins, A.
 Drongelen, R. van
 Hauptmeijer, W.M.
 Koutmeyer
 Balemans, H.C.G.C.
 hristiaans, M.
 Prins, J.H.
 Ploemen, M.M.F.
 Elema, P.
 Kok, A.K.G. de
 Naelten, Mark A.M.G. van
 Spek, W.
 Wit, J.M.
 Pynappel, Leon
 Hoppembrouwers, G.A.J.
 Verhoeven, N.G.
 Reus, B.M. de
 Hoffman, D.J.
 San Salvatorrnavo
 Gendts, P.M.A.G. van
 Tolkaap, G.J.A.
 Kramer, P.
 Feddes, R.A.
 Mavo, Alfred Nobel
 Dragt, A.
 Aalderink, R.H.
 Roelofs, R.A.
 Dieckmann, P.A.
 Kuiken, J.C.J.
 Gemert, M. van
 Koetsier, A.
 Berge, W.F. ten
 Feenstra, M.
 Hendriks, H.
 Horstink, F.H.
 Thijssen, G.
 Schuurman, H.J.G.
 Rienks, R.
 Steverink, M.
 Scholengemeenschap, Chr
 Kreuzen, A.
 Scholengemeenschap Mar:
 Aloysiusmavo
 Timmerman, W.
 Welan, B.G.
 Alma, P.H.

5861 BA WANSUM
 5914 VR VENLO
 5922 XJ VENLO
 5953 LS KEUVER
 6031 BV NEDERWEERT
 6121 JE BORN
 6121 KP BORN
 6171 KK STEIN (L.)
 6212 CH MAASTRICHT
 6214 AR MAASTRICHT
 6227 TD MAASTRICHT
 6228 BT MAASTRICHT
 6228 BT MAASTRICHT
 6267 BG CADIER EN KEER
 6301 EX VALKENBURG (L.)
 6305 AZ SCHIJN OP GEUL
 6336 XN HULSBURG
 6372 MX SCHWABERG
 6433 AJ HOENSBROEK
 6524 LN NIJMEGEN
 6531 CK NIJMEGEN
 6532 XL NIJMEGEN
 6537 TB NIJMEGEN
 6542 RM NIJMEGEN
 6545 NK NIJMEGEN
 6551 AP MEURT
 6561 EX GROESBEEK
 6591 GN DENEKAMP
 6601 SC WIJCHEN
 6602 XA MYCHEN
 6661 BN ELST (bld.)
 6706 CC WAGENINGEN
 6706 GD WAGENINGEN
 6711 AU EDE (GLD)
 6718 WP EDE (GLD)
 6813 JL ARNHEM
 6842 AP ARNHEM
 6845 BH DOORNWERTH
 6865 EB DOORNWERTH
 6874 BR WOLFHEZE
 6931 CV WESTERVOORT
 6941 XK DIDAM
 6942 WD DIDAM
 6991 TR RHEDEN
 7003 DT DOETINCHEM
 7004 CX DOETINCHEM
 7051 ZS VARSSEVELD
 7061 BH TERBORG
 7122 AM AALTEN
 7122 KH AALTEN
 7141 EA GROENLO
 7141 KE GROENLO
 7151 MK EIBERGEN
 7152 GP EIBERGEN
 7161 PH NEEDE

Scholengeme., Isendoorn
 Wijnen, F.M.
 Commandeur, N.J.M.
 Manrooy, W. van
 Kroops, J.B.
 Zantinge, R.
 Maris, Rob J.
 A.V.U.
 Doesschate, J.F. ten
 Nijhuis, S.A.
 Rothengatter, R.A.H.P.
 Wit, Z. de
 Knipstra, Svtse
 Nijke, R.A.F.
 Ruiter, Bert de
 Ruijter, S. de
 Lubberhuizen, J.W.
 Elektr. broodhandel Debe W
 Morssinkhof, G.H.J.
 Bolt, J.H.
 Wisselink, M.
 Wit, F. de
 Computerwinkel Dost Ned.
 Weide, H.v.d.
 Riswick, M.
 Land, K.
 Spiels, H.
 Kapper, B.J.
 Jong, W.O. de
 Heuw, H.G. ten
 Smit, T.A.
 Wisselink, M.A.
 Duthler, J.W.
 Scholengemeenschap Zuid
 Braspenning, N.P.A.
 Dijkstra, Jan Pieter
 Tax, Henny
 Essen, M. van
 Galama, Ysbrant Haye
 Meenderink, J.
 Hoek, R.G.
 Woerthuis, J.
 Koekoek, W.
 Laak, J.J.C. ter
 Mulder, B.
 Oude Nijhuis, J.G.M.
 Johannink, C.F.
 Meijer, R.
 Scholengeme. Erasmus, Rijks
 Hek, L.
 Walraven, C.D.
 Kok, A.A.
 Melchior Winhof MAVO
 Meyering, Tom
 Dijkstra, J.F.

7200 AH ZUTPHEN
 7201 HX ZUTPHEN
 7206 BZ ZUTPHEN
 7221 AC STEENDEREN
 7301 BG APELDOORN
 7311 AA APELDOORN
 7311 BB APELDOORN
 7311 LK APELDOORN
 7313 AP APELDOORN
 7314 AN APELDOORN
 7314 GW APELDOORN
 7316 DS APELDOORN
 7412 XM DEVENTER
 7431 CG DIEPENVEEN
 7441 HS NIJVERDAL
 7441 HS NIJVERDAL
 7451 NL HOLTEN
 7460 AA RIJSEN
 7491 GN DELDEN
 7511 DP ENSCHEDE
 7511 JK ENSCHEDE
 7511 LK ENSCHEDE
 7514 BC ENSCHEDE
 7514 CR ENSCHEDE
 7514 DW ENSCHEDE
 7522 BT ENSCHEDE
 7522 HL ENSCHEDE
 7522 KM ENSCHEDE
 7522 LD ENSCHEDE
 7524 AD ENSCHEDE
 7524 CW ENSCHEDE
 7531 HP ENSCHEDE
 7532 RK ENSCHEDE
 7541 VG ENSCHEDE
 7542 EJ ENSCHEDE
 7544 VH ENSCHEDE
 7545 NC ENSCHEDE
 7546 AD ENSCHEDE
 7551 HH HENGELO (OV)
 7552 GT HENGELO (O.)
 7558 BC HENGELO (O)
 7558 NK HENGELO (O)
 7558 RJ HENGELO (O.)
 7576 TJ OLDENZAAL
 7577 GJ OLDENZAAL
 7581 EZ LOSSER
 7591 TH DENEKAMP
 7602 HA ALMELO
 7602 HT ALMELO
 7622 CL BORNE
 7626 LW HEKIME
 7631 AT OOTMARSUM
 7631 GR OOTMARSUM
 7612 HS ENNEN
 7661 AS OOSTERHESSELEN

Vermet, Rinus
 Geerts, K.
 Gelden, F. G. van
 Scholten, H.
 MCC NewBrain 66, Penn.
 Hovius, Bert
 Ramaker, W. B. J.
 Bakker, D.
 Hellsloot, H. D. P.
 Verhoef, J.
 Sanders, A. F. M.
 Hoeve, Jan Willem
 Lapros, L. B. E.
 Gelden, R. E. van
 Berg, J. v. d.
 Bakker, T.
 Koekoek, B.
 Beylevelt, G.
 Mentjes, I.
 Oldert, J. H.
 Popkema, G.
 Mortelboer, C.
 Lemeer, P. M. J.
 Alting, M.
 Wilkens, J.
 Jansens, W. G. F.
 Jong, L. de
 Kappert, M.
 School, Gem. LEAD
 Johanneschool
 Visman, P.
 Sædes, I. J.
 Bakker, Bart R.
 Ensing, H.
 Snablië, R.
 Schutgens, N.
 Boonstra, G.
 Brons, J. H.
 Bora, P.
 Bruins, Mancel
 Kierenga, W.
 Altenburg, B. S. F.
 Blik, H. J.
 Bijlevelt, Th. J.
 Hoogendijk, J.
 Verhage, Mw. H. B.
 Seinhorst, B. H.
 Kiers, Jan
 Emerencia, S. F.
 Wessels, F.
 Kokhuis, A.
 Hovius, J.
 Drijver, Evert
 Bender, W.
 Cold Air

7944 VM MEPPEL
 7951 EC STAPHORST
 8011 MC ZWOLLE
 8011 MS ZWOLLE
 8011 XM ZWOLLE
 8011 XM ZWOLLE
 8014 ZR ZWOLLE
 8032 BB ZWOLLE
 8032 NN ZWOLLE
 8070 AB MUNSPEET
 8081 JT ELBURG
 8096 AS ULDEBROEK
 8096 ZG ULDEBROEK
 8101 GB RAALTE
 8172 AM VAASSEN
 8251 HR DRONTEN
 8266 GA KAMPEN
 8303 BW EMMELOORD
 8321 AS URK
 8332 BB STEENWIJK
 8426 CX APPELSCHA
 8442 BJ HEERENVEEN
 8536 TD OOSTERZEE
 8608 VV SNEEK
 8608 VV SNEEK
 8608 XJ SNEEK
 8861 JE HARLINGEN
 8861 JE HARLINGEN
 8861 JE HARLINGEN
 907 EL RIDDERKERK
 9247 BG URETERP
 9257 RS NOORDBERGUM
 9257 NF NOORDBERGUM
 9321 AS PEIZE
 9351 DB LEEK
 9355 TK MIDWOLDE (LEEK)
 9405 GK ASSEN
 9462 SE GASSELTE
 9477 RH ZUIDLAREN
 9602 AJ HOOGEZAND
 9602 NP HOOGEZAND
 9711 KS GRONINGEN
 9719 RT GRONINGEN
 9721 WT GRONINGEN
 9722 GK GRONINGEN
 9724 AM GRONINGEN
 9724 NA GRONINGEN
 9727 CK GRONINGEN
 9737 NL GRONINGEN
 9741 AE GRONINGEN
 9741 CE GRONINGEN
 9741 SE GRONINGEN
 9742 LH GRONINGEN
 9801 AC ZUIDHORN
 B-2100 DELURNE (BELGIE)

Vreven, René
 Huet, Laurent
 Velde, Alfons van de
 Houwer, Mia de
 Houwer, Chris de
 Vinck, R.
 Broogmans, Guillaume J. J.
 Hennan, Frans
 Cumps, René
 Jas, D. H.
 Vermeulen, Jef
 Acker, G. A. van
 Commissaris, P.
 Crookes, Philip
 Frenkel, E. J.
 Dusooswa, D. C. J.
 Vos, D. E.
 Groenlund, Bo
 Slegtenhorst, Astrid
 Bartels, G.

B-2400 MUL (BELGIE) (C)
 B-2600 BERCHEN (BELGIE) (C)
 B-2600 BERCHEN (BELGIE) (C)
 B-2610 WILRIJK (BELGIE) (C)
 B-2610 WILRIJK (BELGIE) (C)
 B-2730 ZWIJNDRECHT (BELGIE) (C)
 B-3080 BERTEM (BELGIE) (C)
 B-3110 ROTSELAAR, BELGIE (C)
 B-3128 BAAL (BELGIE) (C)
 B-3761 LANAKEN (BELGIE) (C)
 B-8200 BRUGGE (BELGIE) (C)
 B-8400 OOSTENDE (BELGIE) (C)
 B-9720 DE PINTE (BELGIE) (C)
 BROMLEY BR1 3PE, KENT, ENGLAND (C)
 CH-3032 HINTERKAPPELEN (ZWITS) (C)
 Co. Kerry, S.W. IRELAND (L)
 DK V-1622 KØBENHAVN (DENMARK) (C)
 DK-V-3070 SNEKKERSTEN (DENEM.) (C)
 LEIDERDORP
 A-00 00 SANAA NORTH YEMEN

Ledenlijst N866: sleutel op naam.

A.V. 7311 LK :Aald 6718 WP :Aart 4901 AV :Abbe 2034 BW :Acke B-8400 :Adri 4463 VZ :Agta 4907 ZP :
 Aker 2025 ZD :Alba 4201 BG :Alba 1073 JX :Alis 5109 TZ :Alma 7161 PH :Alvy 7141 KE :Alte 9711 KS :
 Alti 8608 VV :Aria 6121 JE :Asse 1711 RT :Atte 2391 WH :Bake 3257 LJ :Bakk 9257 WP :Bakk 8032 BB :
 Bakk 1921 BW :Bakk 1011 AZ :Bakk 8251 HR :Bale 8267 EG :Ball 1399 GK :Bart 2624 GG :Bart X-00 DD :
 Beer 5463 NE :Beim 1072 EL :Bend 9801 AC :Berg 3295 LE :Berg 8172 AM :Berg 6931 CV :Berk 1541 GV :
 Berk 5271 NK :Berk 3766 EG :Beuk 1054 LS :Bevl 8303 BW :Bijl 9721 WT :Bian 1082 LE :Blee 1012 CT :
 Blik 9718 RT :Blom 5044 ND :Bode 3605 HJ :Boen 3116 AG :Boer 1015 HR :Bolt 7511 DP :Bom, 2273 SH :
 Bont 3116 BK :Bont 5012 GT :Bom 9405 GK :Born 9477 RH :Born 3742 EN :Born 3607 DE :Bors 1861 VB :
 Bote 1381 EE :Both 3076 DL :Bouw 1151 ER :Bras 7542 EJ :Bree 1963 AV :Brin 3514 EJ :Broc 5037 DG :
 Bron 9462 SB :Brug 3811 AM :Bruit 1106 SE :Bruit 9602 AD :Caro 3061 AR :Chri 6301 EX :Citr 2396 CK :
 Cocu 1024 ML :Cold B-2100 :Comm 7206 BZ :Comm B-9720 :Comp 7514 BC :Corb 1181 BZ :Croo BROMLEY :
 Comp B-3128 :Das, 2905 AT :Dass 5171 KX :Davi 2624 L6 :De S 2240 AA :Dege 1056 XB :Dekk 4847 SR :
 Dekk 4818 BC :Denn 1505 WP :Diec 6842 AP :Dijk 2241 SB :Dijk 1354 GP :Dijk 7861 AS :Dijk 7544 VH :
 Ditz 1422 EJ :Ditz 1402 CG :Does 4254 ST :Does 7313 AP :Dore 1401 EM :Drag 6711 AD :Drij 9742 LH :
 Dron 6227 TD :Droo B-3060 :Duik 2403 PD :Duiv 1509 BS :Duso Co. Ker :Duth 7532 RK :Duyv 1562 TP :
 Eeke 3815 VS :Eeke 4726 AE :Egge 1444 VN :Elek 7460 AA :Elen 6372 MX :Elff 1121 CP :Elia 4841 LB :
 Eeer 9737 ML :Enge 3703 GB :Ensi 9321 AS :Erke 2951 BW :Erve 3925 RX :Esse 7546 AD :Fedd 6706 CC :
 Feen 6941 KX :Foek 1015 EA :Fren CH-3032 :Gala 7551 HH :Geer 7951 EC :Geld 8011 HC :Geld 8101 GB :
 Geae 6865 EB :Gend 6601 SC :Gers 1186 EJ :Glas 2806 DA :Groe DK-V-30 :Groo 3134 EB :HCC 8011 XM :
 HCC 1073 JX :HCC 3581 SP :Haag 2596 AK :Haan 5032 BB :Haan 5550 AB :Haar 5301 AD :Hage 1216 AA :
 Ham, 5752 BE :Ham 1241 LV :Hart 3145 TL :Haup 6228 BT :Heer 1053 NX :Heij 5804 AA :Hek, 7622 CL :
 Hek 1211 KX :Hels 8032 MN :Hend 2481 BA :Hend 6942 WD :Hend 4051 EX :Henn B-3110 :Herm 5011 SM :
 Herm 2771 CC :Heuw 258 ED :Heuw 7524 AD :Heyd 2032 ZN :Hidd 1273 JZ :Hilh 1243 KC :Hoeg 4285 CB :
 Hoek 7558 BC :Hoek 1098 KL :Hoev 8096 AS :Hof, 1906 AX :Hoff 6561 EX :Hols 2242 XP :Hoog 2741 XR :
 Hoog 9722 GK :Hoog 4142 AG :Hoog 2105 XV :Hopp 6542 RM :Hors 6991 TR :Houp 6228 BT :Hout 1222 RD :
 Houw B-2610 :Houw B-2610 :Hovi 8011 XM :Hovi 9741 GE :Huet B-2600 :Huiz 2285 HK :Huma 1185 TE :
 Hurk 5703 XR :Huss 3562 GH :Idel 2401 PA :Jans 3705 BW :Jans 3732 ES :Jans 5641 WK :Jans 8608 XJ :
 Jans 6121 KP :Jas, B-3761 :Joha 2181 XE :Joha 907 EL :Joha 7591 TR :Jong 8861 JE :Jong 7522 LD :
 Jong 3171 NB :Jong 3262 BP :Kadi 2341 HN :Kali 1834 AR :Kalm 1016 VS :Kamp 4414 RS :Kamp 3972 VB :
 Kano 1782 GD :Kant 4816 CM :Kapp 7522 KM :Kapp 8861 JE :Kerk 3206 PR :Kier 9727 CK :Kley 4907 HC :
 Knij 7412 XM :Knip 1871 EV :Kobu 2274 CM :Koek 7558 RJ :Koek 8266 GA :Koet 6874 BR :Koff 3712 AR :
 Kok, 7631 AT :Kok, 6435 AJ :Kok, 1688 CW :Kokh 9741 CE :Koni 2102 BE :Koni 2161 RV :Kool 1784 AG :
 Kopp 4388 HM :Kouw 2295 LX :Kram 6661 BN :Kreu 3603 SA :Kreu 7122 KH :Krol 1095 AB :Krop 7301 BG :
 Krui 5434 SZ :Kuik 6865 BH :Kurs 5861 BA :Laak 7576 TJ :Laer 1241 XE :Lage 1823 EV :Lage 2563 GZ :
 Land 7522 BT :Lang 2353 JB :Lap 8096 Z6 :Leen 2403 JK :Leeu 1382 JX :Leeu 1852 KV :Lege 2282 HL :
 Leme 8536 TD :Lemm 5712 BR :Leth 1013 TC :Liev 2691 DX :Lind 4431 CX :Lind 5171 DG :Lohs 1104 AK :
 Loon 2555 KM :Lubb 7451 ML :Luij 4386 CC :Luij 463 VM :Maas 5343 XE :Mari 7311 BB :Mayo 6706 GO :
 Meen 7552 GT :Meer 6212 CM :Meij 2613 BM :Meij 7602 HA :Melc 7631 GR :Mell 31031 B :Meye 2726 ZD :
 Meye 2624 LC :Meye 1211 EJ :Meye 7812 HS :Mill BROMLEY :Mill 4453 BA :Mimp 5151 GJ :Moor 3572 RH :
 Morg 1231 AP :Mors 7491 GW :Mouw 4337 AD :Muld 7577 GJ :Mull 6031 BV :Nabu 3319 VL :Nael 6524 LM :
 Nent 8321 AS :Neut 2982 VN :Neys 5473 GZ :Nier 2411 BS :Nieu 1182 HM :Nieu 1779 EL :Nijh 7314 AN :
 Nura 1183 AC :Ders 2295 LH :Olde 8332 BB :Olph 3962 TB :Oost 1412 GK :Osbo 1422 JS :Oss, 1078 GB :
 Oude 7581 EZ :Oude 3844 NK :Ouds 1034 LL :Paas 1829 HJ :Pell 2266 ES :Pell 1098 CK :Pelt 2324 EA :
 Pers 1412 AN :Piet 1102 CA :Pill 3008 AE :Pint 3314 HH :Ploe 1782 AT :Ploe 6336 XN :Pion 2582 LN :
 Poi, 2586 TB :Popk 8426 CX :Prin 6214 AR :Prin 6305 AZ :Pron 5737 EG :Pyna 6537 TB :Raij 1442 NG :
 Rama 8014 ZR :Regt 2700 AE :Regt 3721 BB :Rein 1962 SG :Reke 1094 JH :Reue 1079 GR :Repe 2312 GP :
 Reus 6551 AP :Reuv 2624 CM :Rhee 1217 TE :Ridd 1078 W :Rien 7051 ZS :Rijc 2311 GL :Rijk 3571 SL :
 Rijn 1141 RM :Rijn 1141 JA :Rism 7514 DW :Roel 6813 JL :Roo, 4431 DD :Roos 4206 AG :Roos 2574 SD :
 Rosk 109 J6 :Roth 7314 GW :Ruij 7441 HS :Ruit 7441 HS :Russ 5802 AD :Ruth 1056 AS :Rutt 1221 EE :
 San 6591 GN :Sand 8081 JT :Scho 5953 LS :Scho 7602 HT :Scho 7200 RH :Scho 7141 EA :Scho 7541 VG :
 Scho 7122 AW :Scho 1068 MR :Scho 8011 MS :Scho 3511 VD :Scho 1782 ND :Scho 8861 JE :Scho 3133 KP :
 Schu 9355 TK :Scho 7004 CX :Scho 4463 CJ :Scul 1073 HG :Sein 9724 NA :Sleg LEIDERD :Slui 4121 BC :

Smed 9257 RS :Smit 7524 CW :Snao 9351 DB :Snij 1273 AP :Sonn 5922 XJ :Spal 3431 SK :Spek 6531 CK :
 Spie 7522 HL :Steij 5914 VR :Steu 1381 AJ :Stev 1092 JV :Stev 7061 BH :Stol 1504 AT :Stro 4384 LC :
 Tax, 7545 MC :Tech 1025 RX :Tech 2105 SK :Tech 1034 LL :Thie 2316 AC :Thij 7003 DT :Tilb 4902 AE :
 Timm 7151 MK :Toik 6602 XA :Toni 5368 AJ :Trui 3512 NP :Unia 3972 XB :Uyld 1353 CH :Valk 2713 CE :
 Vals 1215 RL :Veen 3295 VC :Veld B-2600 :Veld 4475 AB :Veld 5343 XK :Veld 5051 BT :Velt 4384 BP :
 Verb 3061 GN :Verb 1628 HK :Verb 3811 LR :Verc 5144 CK :Verh 9724 AM :Verh 8070 AB :Verh 6545 NK :
 Verm 7944 VM :Vera 3174 EN :Vera B-8200 :Vinc B-2730 :Vish 1093 TP :Visa 9247 BG :Voor 3641 AK :
 Vooy 2722 SC :Vos, DK V-16 :Voss 5212 PH :Vree 1402 BA :Vreu 2582 XK :Vrev B-2400 :Vrie 1484 PC :
 Vrie 3581 SX :Vrij 3116 AJ :Wair 7626 LW :Wann 7221 AC :Weid 7514 CR :Weij 5014 BL :Wela 7152 GP :
 Welt 5708 JD :Weni 3203 BG :Werk 1078 GL :Wess 9741 AE :West 4333 EM :Wiel 1214 CP :Wiel 4561 AC :
 Wier 9602 NP :Wier 5161 GH :Wijg 4905 AX :Wijk 5042 BH :Wijn 7201 HX :Wijn 3828 VC :Wilk 7431 CG :
 Wilk 8608 VV :Wila 2402 AL :Wiss 7511 JK :Wiss 7531 HF :Wit, 7511 LK :Wit, 6532 XL :Wit, 7316 DS :
 Woer 7558 NK :Wort 8442 BJ :Woud 1121 CP :Wout 1703 KB :Zane 1509 ZA :Zant 7311 AA :Zeeg 2034 SP :
 Zome 1315 CL :Zuur 1000 BP :Zwaai 2811 AN :Zwan 3435 BH :Zwar 2201 BN :