

signaler impuls

Jaargang 9 - Nummer 3 & 4

COLOFON - SINCLAIR IMPULS - JAARGANG 9 - DE NUMMERS 3 EN 4

"Sinclair Impuls"

HET blad voor en door de gebruikers
van ALLE Sinclaircomputers
- ZX80, ZX81, ZXSpectrum, QL en aanverwanten -
werd uitgegeven door de
"HCC Sinclair Gebruikers Groep" (SGG).

IMPULSREDACTIE

Ed Weijgers
H Marsmanlaan 29
2624 TJ Delft

Kees Versluis
Cypresstraat 25
2565 LS Den Haag
Tel 070-3463203

SGG-ADRES

Theo Molenaar,
secretaris

HCC Sinclair GG
Postbus 76
2260 AB Leidschendam

SGG-REKENING

George Burghgraef,
penningmeester

Postrekening 5374525
tnv HCC Sinclair GG
te Bunnik

DUCBANK (OPUS DISCOVERY)

via onze SGG-rekening
bankier Kees Versluis
Infotel 070-3463203

DD-BANK (DISCiPLE)

via onze SGG-rekening
bankier Rob Willig
Infotel 02159-43087

--
VAN DE REDACTIE
--

Lang gewacht en stil gezwezen,
niet (meer) verwacht: toch gekregen!

Hier is-ie dan toch, dat beloofde laatste dubbelnummer van onze IMPULS. Twee jaar te laat, dat wel, maar: belofte maakt schuld. We wilden u niet net zo laten zitten als een ander blad deed. Daarom heeft de redactie - tenminste wat daar nog van over is - alles op alles gezet en met een laatste nu-of-nooit krachtsinspanning dit slotstuk eruitgeperst. Nu kan onze penningmeester gerust zijn: hij behoeft geen girootjes te gaan zitten schrijven om alle abonnees een deel van hun betaalde geld te restitueren.

Veel van de kopij lag al tijden op de redactie, dus het is goed mogelijk dat er iets niet geheel meer klopt. Onze INFOTELEFOON, annex databank SINCLAIR BOX, bestaat bv helaas niet meer, aangezien Jack Raats door drukke werkzaamheden al zijn werk voor onze SGG moest beeindigen. Hij heeft lang veel voor ons gedaan, waarvoor wij hem hier hartelijk danken.

Het ligt niet in onze bedoeling om hier iedereen te bedanken en grafredes uit te spreken. Een uitzondering nog: de bijdrage van Rob Willig aan deze IMPULS is toch wel onmiskenbaar. Welbedankt.

Wat nog wel bestaat en voorlopig ook nog blijft bestaan zijn onze SGG-dagen. Die worden tegenwoordig gehouden in Bunnik, in het gebouw van de Gereformeerde Kerk. De data voor het volgend jaar:

5 februari, 9 april, 18 juni, 27 augustus en 1 oktober 1994.

Zoals gebruikelijk van 10 tot 16 uur. U kunt er als volgt komen. Op de autoweg tussen Utrecht en Arnhem neemt u de afslag Bunnik. Bij het tweede verkeerslicht na de "ahob" (spoor Utrecht-Arnhem) gaat u op de ventweg rijden. De eerstvolgende straat rechts is dan al de Cammingalaan, waaraan u het kerkgebouw zo ziet liggen. Voor mensen die met de trein komen moet dit ook voldoende zijn.

Er zijn daar dit jaar al twee bijeenkomsten geweest, maar zonder inbreng van SINCLAIR heb ik vernomen. De QUASAR heb ik ook al een tijd niet meer gehad. Wat ik nog wel regelmatig zie is het BULLETIN van de SinclairGebruikersGroep Groningen/Assen. Dat is dan het laatste tijdschrift! Daarin staat ook nog steeds vermeld Albert Hoekman, de Hennepe 351, 4003 BD Tiel, 03440-32182 voor reparaties. Altijd handig om te weten dat u nog ergens heen kunt.

Wat moeten we u deze laatste keer eigenlijk nog meer vertellen? Dat er nog steeds mensen zijn die lol in Sinclaircomputers hebben, maar dat hun aantal afneemt weet u wel, want daar behoren u en wij ook toe. We gaan door, zij het op een steeds lager pitje.

- rEd -

-- SIR CLIVE SINCLAIR --

In de automatiseringsspecial van het weekblad Intermediair (27e jaargang 10, 8 maart 1991) werd uitgebreid aandacht besteed aan de vele pioniers uit de geschiedenis van de computertechnologie, onder de kop 'Hall of Fame'.

Sir Clive Sinclair kwam, met kopstukken als Steve Jobs en Samuel Meijering, aan de orde in een artikel met een iets andere titel: 'Hall of Shame'.

In dat artikel, hieronder afgedrukt, wordt een tamelijk kritische blik geworpen op de verdiensten van Sinclair, die als een groot verliezer wordt afgeschilderd. Het is de vraag of de lezers van Impuls het met de beide auteurs ervan eens kunnen zijn. Het meest opmerkelijk is wellicht nog wel hun idee dat 'de ZX' - weten zij veel dat er zeker drie verschillende ZX'en bestaan - maar een uiterst simpele spelletjescomputer is, de Commodore 64 daarentegen een echte homecomputer, vergelijkbaar met de QL! Mij dunkt dat minstens een dezer schrijvers ooit zo'n C64 bezat.

Carlo Delhez

HALL OF SHAME

De Brit Clive Sinclair is uitvinder van de pocket-tv, van de C5 elektrische driewieler en de ZX80 spelletjescomputer. Hij werd geridderd door Margaret Thatcher en was tot voor een aantal jaren de bekendste Britse wetenschapper en uitvinder. Maar het liep slecht af met Sir of Uncle Clive, zoals het Britse publiek hem noemt.

Sinclair was rijk geworden met zijn postorderbedrijf. Een bedrijf runnen was niet zijn sterkste kant en in april 1986 was hij genoodzaakt zijn computerproducten aan Amstrad te verkopen. Daarmee verdween de naam Sinclair van het toneel. Hoewel men Sir Clive vaak toedicht grote verdiensten te hebben gehad voor het onderwijs en het uitbannen van de onwetendheid rond computers, is dat bezijden de waarheid, of men moet de bevordering van verslaving aan computerspelletjes als zodanig betitelen.

Sinclairs grootste prestatie was dat het hem is gelukt om een computer voor minder dan honderd pond sterling op de markt te brengen. Dat er daarbij bezuinigd moest worden op het toetsenbord, beeldscherm, geheugen en gegevensopslagsysteem, doet er even niet toe, het ging erom dat je iets in handen had dat op een computer leek. De opvolger van Sinclairs eerste en uiterst succesvolle computer, de ZX, was de Quantum Leap of QL. Deze keer was er gezorgd voor iets dat op een echt toetsenbord leek, en de gebruiker kreeg de beschikking over een ingebouwd opslagsysteem voor gegevens en programma's. Het was een poging van Sinclair om de uiterst simpele spelletjescomputer achter zich te laten en een echte home computer te bouwen in de traditie van de Commodore 64.

-- SIR CLIVE SINCLAIR --

De lancering van de QL was een grandioze mislukking. Veel kostbare tijd werd verdaan met het ontwikkelen van een plat beeldscherm op basis van Sinclairs kleine beeldbuis voor de pocket-tv. Nog meer vertraging liep men op door de keuze voor een nieuwe en inefficiënte zogenoemde 'microdrive' als data-opslagsysteem. De microdrive was een goedkope vervanger van de floppy disc en werkte op basis van een snelle, kleine cassetterecorder waarin een eindeloze band draait.

Sinclair had er een handje van producten teannonceren die nog niet af waren. Met de QL-lancering bereikte hij wat dat betreft zijn dieptepunt, de ontwikkelaars van het bedrijf hadden toen zelfs nog geen prototype gezien. Het duurde een jaar voordat de computer leverbaar was, en omdat Sinclair zijn productie veelal uitbesteedde en niet veel aan kwaliteitscontrole deed, kochten veel mensen een niet-werkende QL.

Na de verkoop van zijn computerbedrijf ging Sinclair door met Sinclair Research, een laboratorium waar hij stug doorwerkte aan nieuwe dromen. Een daarvan was de zogenoemde 'waferscale integration'. Tot op de dag van vandaag worden chips, de bouwstenen van een computer, uit een grote plak silicium gezaagd, elk apart getest, verpakt, van connectoren voorzien en weer met elkaar verbonden op een printplaat. Het zou logischer zijn om een complete computer op een grote plak silicium aan te brengen, ware het niet dat veel defecte chips op de plak dat moeilijk realiseerbaar maken.

Het is niettemin een verleidelijke technologie, zo verleidelijk dat grote bedrijven als Digital Equipment, Sperry (inmiddels opgegaan in Unisys) en Bull er begin jaren tachtig 240 miljoen dollar aan verspilden. De investering die ze samen in het bedrijf Trilogy staken, ging in rook op toen in juni 1984 wegens enorme technische problemen het onderzoek werd gestaakt.

Waar de groten faalden, wilde Sinclair wel een stap wagen. Dat gebeurde op instigatie van Ivor Catt, die in 1983 bij Sinclair in dienst was gekomen. Catt had twintig jaar ervaring in de halfgeleiderindustrie en stond bekend als een visionair dan wel volslagen idioot. Het project heeft nooit wat opgeleverd.

Sinclair is gelouterd na zijn ontnuchterende ervaringen. Voorlopig begint hij niet meer aan apparaten waar elektriciteit aan te pas komt. 'Het probleem met de C5 (een elektrische driewieler, die een enorme flop werd) was', zei hij vorig jaar in New Scientist, 'dat ik iets maakte dat ik zelf niet nodig had.' Dat zal hem niet meer overkomen. Momenteel werkt hij aan een opvouwbare stadsfiets, gemaakt van onder andere Kevlar (een supersterke vezel), een vervoermiddel waar hij om zit te springen, want hij fietst graag naar zijn Londense werkruimte.

OD

Een CAT-file bestaat uit records van 16 bytes: drie velden voor getallen van twee bytes (L voor H) en een naamveld van 10 bytes. De records voor de files worden geflankeerd door een beginrecord met de disknaam en een eindrecord met de eindmarkeerder (65535).

	beginrecord	filerecords	eindrecord
00	aantal der bytes	idem	kopie uit het
01	in laatste blok - 1		beginrecord
02	eerste	idem	aantal der in BASIC
03	bloknummer		bereikbare blokken
04	laatste	idem	de 'eindmarkeerder'
05	bloknummer		65535 (255 & 255)
06	naam	naam	kopie uit het
15	van de disk	van de file	beginrecord

Voor het eerste getal bezit de hoogste nibble geen betekenis. Volgens de handleiding worden die vier bits door het systeem gebruikt. Ik heb er echter nooit andere waarden dan nul in gezien.

Bij het beginrecord komt het eerste getal overeen met de blok-grootte minus 1. Het tweede moet 0 zijn (anders foutmelding) en het derde getal plus 1 is dus het totale aantal der CAT-blokken.

Bij een filerecord krijgt men de totale lengte van de file, dus inclusief een eventuele header van 7 bytes (bij een N-file), dmv

```
(eerste getal +1)-(tweede getal - derde getal)*(blokgrootte)
```

Een file, waarvan de eerste byte onder de 4 en het getal in het volgende paar bytes bovendien 7 onder de totale filelengte ligt, is zo goed als zeker een N-file en geen DATA-file (criterium).

Het eindrecord behoeft niet het laatste record van een CAT-file te zijn. Bij een lege disk bv is dit reeds het tweede record. Hiervan hebben slechts het tweede en derde getal echt betekenis. Het tweede getal is bij een 'echte' disk het product van de aantallen der tracks/side, blocks/track (sectors) en sides/disk verminderd met 1 (omdat het ankerblok in BASIC onbereikbaar is). Bij RAM-disk 5 en 6 is dit dus wel het aantal van alle blokken.

MOVE " CAT ";D TO ... en MOVE ... TO " CAT ";D slaan altijd het beginrecord over. Gebruik om dit toch te lezen of te beschrijven OPEN #S;" CAT ";D:POINT #S;1: MOVE #S TO ... of MOVE ... TO #S.

OD

CLEAR CAT IPV FORMAT

OD

In het julinumnummer van "QUASAR" stond het artikel "Fast Format". Hierin werd opgemerkt dat het opnieuw FORMATTen van een disk die al geFORMAT is veel meer tijd kost dan eigenlijk nodig is. Het is nl overbodig om zo'n disk helemaal van nieuwe tracks te voorzien wanneer die er nog zijn (mits ze natuurlijk in orde zijn). Alleen de CATALOGUS legen volstaat dan. Dit nu kan bij de OD ook heel eenvoudig worden gedaan, zelfs in BASIC en zonder ERASEn.

HET BASICPROGRAMMA "clearcat" LINE 1 - EdW -

```

1>CLEAR #: DIM N$(10):
  INPUT ,," CLEAR CAT IN DRIVE ";D," NEW DISKNAME",N$:
  IF D<1 OR D>4 THEN RUN
2 OPEN #3;" CAT ";D: POKE 23774,1:POINT #3;1:
  OPEN #4;"CODE ":POINT #4;10114:
  LET N=CODE INKEY$#4*CODE INKEY$#4: LET Z=CODE INKEY$#4/16
3 RANDOMIZE N*(INT Z-INT (Z/2)*2+1)-1:POINT #4;23670:
  LET H$=INKEY$#3+INKEY$#3+INKEY$#4+INKEY$#4+" COPY COPY ":
  POINT #3;7: LPRINT N$;H$;N$;: CLOSE #3: RUN

```

TOELICHTING

Eerst vraagt regel 1 om het drivenummer en de (nieuwe) disknaam. Dan OPENT regel 2 het CAT-kanaal, wijzigt de recordlengte daarin van 16 in 1 en richt de POINTER op het begin van de CAT-file. Na OPENING staat er altijd een kopie van het ankerblok in IC0. Hierin bevinden zich op de adressen 10114 t/m 10116 de aantallen van tracks/side, blocks/track en sides/disk (zie IMPULS 91-09). Deze worden gelezen en hun product, verminderd met 1, wordt ontbonden in SEED. Dit is nl het aantal der vanuit BASIC bereikbare blokken, dat in het CAT-record met de eindmarkeerder moet staan. Vervolgens wordt de blokgrootte-1 uit het oorspronkelijke begin-record, aangevuld met het aantal uit SEED en een eindmarkeerder (255 255), in een hulpstring gezet. Tot slot wordt de POINTER op het eerste naamveld gericht en worden de naam, de hulpstring en weer de naam in de CAT-file geschreven. De CATALOGUS is nu leeg.

OPMERKINGEN

Gebruik dit programma nooit wanneer u geen IC6116 in uw OD hebt, en ook niet nadat u ZX-IF1-ruimte gecreeerd hebt door USR 4007.

Informatie over de opbouw der CAT-records staat in IMPULS 63-39.

We hadden ook het oude eindrecord naar het tweede record kunnen kopiëren. Maar als er veel files zijn duurt het opzoeken daarvan lang en bovendien wilden we meteen de disknaam kunnen wijzigen.

E H F Weijgers - H Marsmanlaan 29 - 2624 TJ Delft

SP

PRINTEN EN POKEN MET DE MULTIPRINT

SP

Het MultiPrint-interface (MP) is een interface voor het maken van schermafdrucken en het onderzoeken van het SP RAM-geheugen. De fabrikant is de onder Spectrumgebruikers vanwege de MultiFace (MF) bekende firma Romantic Robot.

VROEGER. Ooit begon ik met een Seikosha GP-50S. Maar ook bij mij moest het groter, sneller en mooier. Ik werk vrij veel met TW2 en op den duur voldeed mijn eigen "TW GP" niet meer. Ik zette mijn zinnen op het ZX LPrint interface met een STAR NL-10.

AANKOOP. Op naar de HCC Dagen, voor printer plus interface. Ik kwam thuis met inderdaad een STAR printer, maar niet met een ZX LPrint. Bij de kraam van Elra wist men mij te overtuigen van een kastje dat meer kon en minder kostte. De MultiPrint. In een vrij kleine doos zat een kastje met kabel en een handleiding. Ik zal over beide iets vertellen.

HANDLEIDING. Het papier in het doosje verdient nauwelijks de naam handleiding. Acht kantjes A5, in het Engels. De reclame op de voorkant en de garantiebepalingen op de achterkant vullen samen een kantje, dus netto kreeg ik zeven kantjes uitleg. Uiteraard wordt verteld wat u kunt instellen, hoe u dat doet en waarom dat moet of kan. Speciale aandacht is er voor Tasword. Er staan echter ook dingen niet in de handleiding. Hoe past u Omnicalc en Beta BASIC aan bijvoorbeeld. Wat kunt u verwachten met Art Studio en zijn er moeilijkheden met Dumpy? Het aanpassen van beveiligde software kan eigenlijk alleen door een kopie te maken op een apart bandje met een LOADprogrammaatje ervoor. Dan moet u zo'n programma wel kunnen kopiëren. Ook staat er nergens dat de printer na de SP aangezet moet worden. Anders loopt u kans op een hangup. Of dat een exemplarische fout van version 87/5 is weet ik niet, dat het vervelend is wel. Ook aan de MF-achtige toolkit wordt aandacht besteed. Er schijnt zelfs een speciale (dis)assembler voor de MP te bestaan. Ik ben die echter nog nooit tegengekomen. De MF zelf ken ik niet. Ik moet voor wat betreft de mogelijkheden van dat kastje dus op de MP-handleiding vertrouwen. U kunt met de MP geen programma-backups (of illegale duplicaten) maken. Met de MP kunt u alleen het beeldscherm opslaan op band onder de naam "Screen". U kunt het plaatje echter ook op een dotmatrix printer afdrucken. Dat is de specialiteit van de MP. U kunt kiezen uit:

CPT Afdruk van ASCII-schermen. Nuttig als u een spelletje wilt spelen waarvan u de handleiding kwijt bent. U kunt dan de instructies zondermeer dumpen. Zelfs al maakt het spel gebruik van een nieuwe, onleesbare tekenset.

CPN Afdruk als van een ZX-printer.

CPL Afdruk als een ZX-printer, maar dan groter.

CPS Een mooie, grote afdruk, met kleuren als grijstinten.

Een genot om te zien. Kost alleen veel tijd en inktlint.

Hoewel ik inmiddels met de DD werk, gebruik ik mijn MP toch nog zo af en toe voor de schermafdrucken. Met de DD worden die ofwel

SP

PRINTEN EN POKEN MET DE MULTIPRINT

SP

(te) groot of (te) klein, maar het MP-formaat vind ik precies goed. Bovendien hoef ik met de MP mijn afdrucker tenminste niet te kantelen. Ik snap de schermen met de DD, zet ze op band en gebruik dan mijn reserve rubber SP om ze te dumpen.

TOEGANG. U kunt de instellingen wijzigen door of REM MP <ENTER> te typen en vervolgens op een letter te drukken, of door REM MP, inst te tikken. Voor <inst> typt u een afkorting, zoals die staat aangegeven in de handleiding. De Toolkit kunt u oproepen dmv de rode Bevriesknop. Vanuit de Toolkit kunt u de instellingen wijzigen. Andersom gaat helaas niet. Vanuit BASIC is de Toolkit dus ook niet bereikbaar. Jammer. Ik vraag me af waarom Romantic Robot niet koos voor bv "MPC" voor "configure" instellingen en "MPU" om de Toolkit te kunnen gebruiken. Bij mij wilde alles nog wel eens vastlopen, als ik een instelling voor de tweede keer wijzigde. Vanuit BASIC gaf dat steeds een "Out of Memory" melding, zelfs als het geheugen leeg was. Ik heb dat echter nooit onoverkomelijk gevonden.

PROGRAMMEREN. U kunt Printer Driver Strings maken, om stuurcodes naar uw printer te zenden. Die worden dan in het interface zelf opgeslagen, zodat ze een eventuele reset overleven. Waar ze precies terechtkomen en hoe, wordt niet verteld. Een lijst gedefinieerde PDStrings opvragen kan niet en al met al vind ik dat niet handig. U kunt beter het volgende doen: Toets REM MP, zet de Tokens Off, verlaat de instellingen, type uw opdracht en zet de Tokens weer On. Of laat TW2 het vuile werk opknappen.

KASTJE. Tot zover de handleiding. Die zegt hoe het moet, maar de mogelijkheden zitten in het kastje. Dat moet u achter op de doorvoerpoort schuiven. Van achter komt er een ca 1.40 meter lange lintkabel uit. Die moet naar de printer. Er is ook een doorvoerconnector. Die is naar mijn smaak alleen te smal. Het achterop schuiven van mijn losse joystickinterface was een hele klus. Ook in verband met de bovenop gemonteerde Magic Button, een drukknop, is een goede ondergrond geen overbodige luxe.

BUITENBOORD. Het interface heeft een eigen ROM en RAM aan boord. Daarin zitten de Toolkit en de instellingen. De 8K RAM kunt u gebruiken voor de opslag van eigen MC, data of plaatjes. Alleen de manier hoe. Dat staat niet duidelijk genoeg in de handleiding en als je dan ook nog, zoals ik, geen held bent in Z80 assembly, kun je dat verder wel vergeten. Slechts met deskundige hulp tijdens een gebruikersdag in Houten ben ik eruit gekomen.

COLLECTIE. Zoals hierboven viel te lezen, heeft het interface z'n eigen RAM en ROM geheugen. Dat kan dus botsen met andere interfaces, die ook hun eigen geheugenchips hebben. Voor de combinatie DD met MF had wijlen de DD Nieuwsbrief een oplossing. Ik weet echter niet of die ook voor de MP geldt.

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087

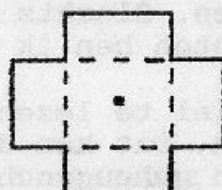
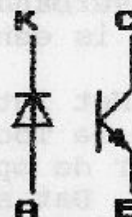
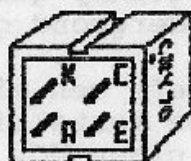
Eerst even een refresh van uw memory. In de vorige Impuls stond de beschrijving van een plotter en de software. Toen vertelde ik u wat zoal de mogelijkheden zijn. Nu is het werken met een plotter wel aardig, maar dingen verbeteren aan mechaniek en software vind ik leuker. Hierdoor werkt de plotter nu met direct-drive op de papieras en is al de zesde pennenhouder gemonteerd (de vijfde bleek toch te zwaar te zijn). Toen dit alles af was ben ik weer een nieuw ideetje gaan uitwerken: een SCANNER oftewel inlezer.



De belangrijkste dingen hiervoor zijn: positieherkenning en een leeskop, beter gezegd reflectietastkop. Voor het eerste kon ik mijn plotter gebruiken. Verder kocht ik twee reflectietasters in een elektronikawinkel, waaronder een CNY 70. Door experimenteren kwam ik erachter dat die voor dit doel het meest geschikt is.

Hiernaast een tekening van een CNY 70.

Ze kosten ongeveer f 6.-. (Straks volgt meer over het plaatje).



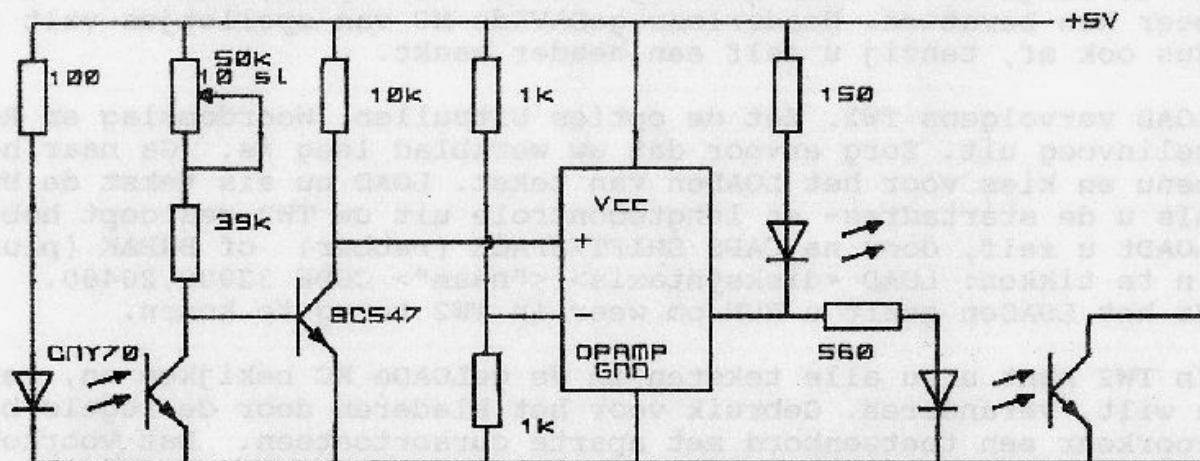
**Plaatje van
latoenkoper.
Gat in het
midden +/-
0.75 mm**

Na het maken van de benodigde versterker/comparator-schakeling, kwam ik erachter dat de tastkop te veel zag. Ook licht van buiten beïnvloedde het resultaat. De plotter heeft een stapgrootte van 0.2 mm. De tastkop mag de plek die hij zag, na een stap niet

HET INLEZEN VAN TEKENINGEN OP DE SPECTRUM

meer zien. Door nu een latoenkoperen plaatje, aan de binnenkant gezwart en met een gat in het midden, op de kop te monteren, is de scheiding maximaal en de valse lichtinvloed minimaal. Na de nodige tests bleek een gaatje van 0.75 mm voldoende te zijn. Het liefst had ik het nog kleiner gewild, maar de boortjes daarvoor ontbraken in mijn prive-collectie. Theoretisch is een gaatje van 0.2 mm nodig, maar dan komt er zo weinig licht terug dat de kop helemaal geen verschillen meer ziet.

Na deze veranderingen overtroffen de resultaten mijn verwachtingen. Zelfs zwart-wit foto's nam hij goed over. Alleen kleurenfoto's niet, maar kopieën daarvan weer wel. De ingelezen SCREEN\$ LOAD ik in "Studio Artist" om ze nog wat bij te werken.



De versterker/comparator-schakeling is hierboven afgebeeld. Zoals de naam al aangeeft gaat het signaal eerst naar een versterkertje. Daar is tevens de signaalspanning te regelen. De Opamp, die als comparator geschakeld is, vergelijkt dit signaal met de spanning op een weerstandsdeler en stuurt een 'hoog' of 'laag' naar de Opto-coupler. De uitgang gaat naar de Z80A PIO. Zo'n PIO kost ca f 10.-, de componenten in de schakeling ongeveer f 15.-. Alle andere illustraties in dit verhaal zijn ingelezen door de scanner, ietwat bijgewerkt en geplot door de plotter. De kat is van een zwartwitfoto overgenomen, net als het gezicht hieronder.

Mocht u iets dergelijks willen bouwen, een printerkop zou ook als een positieherkenner kunnen dienen.



SCUMARI Research and Development

Rick Schuitemaker
Nauerna 84
1161 DT Zwanenburg

SP

MC BEWERKEN MET TASWORD TWEE

SP

Dat tekstbewerking een van de meest gebruikte zakelijke toepassingen voor een computer is zal duidelijk zijn. Dat Tasword Twee (TW2) een veelgebruikte tekstbewerker is bij de ZX Spectrum weet ook elke Spectrist. Deze tekst is bijvoorbeeld gemaakt met TW2.

Dat het met TW2 mogelijk is een tekst in te typen of te LOADen, te wijzigen en tenslotte af te drukken en/of te bewaren, is bekend. Waarschijnlijk minder, of helemaal niet bekend is, dat u met TW2 ook software kunt vertalen. Mits het althans machinecode (MC) betreft. Het gaat als volgt.

Kies de MC die u wilt bewerken. Controleer met een koppenlezer of de lengte niet meer bedraagt dan 20.480 bytes, omdat TW2 niet meer kan bevatten. Headerless geSAVEde MC van spelletjes valt zo dus ook af, tenzij u zelf een header maakt.

LOAD vervolgens TW2. Zet de opties Uitvullen, Woordomslog en Regel invoeg uit. Zorg ervoor dat uw werkblad leeg is. Ga naar het menu en kies voor het LOADen van tekst. LOAD nu als tekst de MC. Als u de startadres- en lengtecontrole uit uw TW2 gesloopt hebt, LOADt u zelf, door na CAPS SHIFT/SPACE (rubber) of BREAK (plus) in te tikken: LOAD <disksyntaxis> <"naam"> CODE 32000,20480. Na het LOADen geeft u RUN om weer in TW2 terug te komen.

In TW2 kunt u nu alle teksten in de geLOADE MC bekijken en, als u wilt, veranderen. Gebruik voor het bladeren door de regels bij voorkeur een toetsenbord met aparte cursortoetsen. Dat voorkomt dat u op ongelegen momenten per ongeluk een getal intypt: niet eerst op CAPS SHIFT drukken kan de beste overkomen. En als dat gebeurt bent u uw werk "kwijt". In 99 van de 100 gevallen is dan nl de MC onbruikbaar, en moet u weer van voren af aan beginnen. Het zal u ook duidelijk zijn waarom alle TW2-functies die spaties toevoegen of andere, in dit geval nare, dingen met de tekst kunnen uithalen, uitgeschakeld moeten zijn.

Zodra u klaar bent gaat u weer naar het menu. U SAVET de bewerkte MC nu HANDMATIG weg met CODE 32000,lengte. Laat dit niet aan TW2 over, want dan kan de bewerkte MC langer dan eerst worden. Zet de bewerking op een apart bandje. Type nu PRINT USR 0 in of druk op de resetknop. LOAD vervolgens de bewerkte CODE op het startadres van het origineel en start het programma, eventueel met uw vinger op de resetknop. Als alles werkt: gefeliciteerd! Zoniet, dan zult u helemaal opnieuw moeten beginnen. Helaas ... Op deze manier heb ik bijvoorbeeld in de hulppagina het woord "lijn" vervangen door "regel" en de vloek "uitlijnen" door "uitvullen" (zonder breukstreepje).

NB: Ik heb met opzet betrekkelijk vage aanwijzingen gegeven voor het werken met TW2, omdat vrijwel iedereen zijn "eigen" versie heeft ontworpen, met eigen menu's en keuzes enz.

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087

DD

HiSOFT-C OP DE DISCIPLE

DD

Wie vaak computerlectuur leest weet dat de meeste programmatuur tegenwoordig in C wordt geschreven. Nu bestaat er voor de Spectrum al jaren een C-compiler, die uitgebracht wordt door HiSoft.

Deze compiler is redelijk K&R-compatibel; alleen floating point ontbreekt, maar voor veel toepassingen is dit niet zo'n gemis. Wel jammer is dat hij uitsluitend geschreven werd voor CR en MD.

Gelukkig schreef Enrico Maria Giordano uit Rome een aantal functies, die werken met de Disciple mogelijk maken. Voor vragen en updates kunt u terecht in message-area 12 van onze Sinclair Box.

Hier volgt de include-file voor de Disciple.

Voeg het onderstaande toe aan STDIO.H of noem deze file DISCD.H.

```
#define BSZ 5000
#define EOF -1
#define NULL 0

typedef struct
{
    char    DSTR1;
    char    FSTR1;
    char    SSTR1;
    char    LSTR1;
    char    NSTR1;
    char    NSTR2[10];
    char    HD00;
    unsigned HD0B;
    unsigned HD0D;
    unsigned HD0F;
    unsigned HD11;
} FILE;

FILE UFIA;
unsigned _RPT;
char    _MODE;

/* end of DISCD.H */
```

Op het volgende tweetal bladzijden vindt u nog een include-file, DISCD.LIB, met de functies die u in staat stellen om de Disciple te gebruiken in HiSoft-C. U kunt het beste de volgende functies anders benoemen: fopen, putc, getc en fclose. Kies hier bv voor: dfopen, dputc, dgetc en dfclose. U kunt dan nog steeds de standaardroutines gebruiken. SAVE de volgende routines onder de naam DISCD.LIB en include die altijd voor STDIO.LIB.

DD

HiSOFT-C OP DE DISCIPLE

DD

```

/* start of DISCD.LIB */
FILE *fopen(name, mode)
char *name, *mode;
{
    static char chr;
    char b[BSZ];
    int i, j;
    if (!MODE) return NULL;
    UFIA.DSTR1 = 1;
    UFIA.FSTR1 = 0;
    UFIA.SSTR1 = 0;
    UFIA.LSTR1 = 'd';
    UFIA.NSTR1 = 4;
    move(UFIA.NSTR2, "
", 10);
    for (i = 0; i < 10 && *name; UFIA.NSTR2[i++] = *name++);
    UFIA.HD00 = 3;
    UFIA.HD0B = 0;
    UFIA.HD0D = 0;
    UFIA.HD0F = 0;
    UFIA.HD11 = 0;
    _RPT = 0;
    switch (*mode)
    {
        case 'r':
            inline(221, 229, 221, 33, &UFIA, 207, 59, 56, 15);
            inline(221, 34, &RPT, 17, &UFIA.HD00, 6, 9, 207, 60, 18);
            inline(19, 16, 250, 221, 225);
            break;
        case 'w':
            inline(221, 229, 221, 33, &UFIA, 207, 65, 221, 33, &UFIA);
            inline(207, 53, 56, 4, 221, 34, &RPT, 221, 225);
            break;
        case 'a':
            static char chr;
            putc(c, fp);
            int c;
            FILE *fp;
            {
                static char chr;
                inline(221, 229, 221, 33, &UFIA, 207, 59, 56, 15);
                inline(221, 34, &RPT, 17, &UFIA.HD00, 6, 9, 207, 60, 18);
                inline(19, 16, 250, 221, 225);
                if (!RPT) return NULL;
                for (i = 0; i < BSZ && UFIA.HDOB; i++)
                {
                    inline(221, 229, 221, 42, &RPT, 207, 60, 50, &chr);
                    inline(221, 225);
                    b[i] = chr;
                    UFIA.HDOB--;
                }
                if (UFIA.HDOB) return NULL;
                inline(221, 229, 221, 33, &UFIA, 207, 65, 221, 33, &UFIA);
                inline(207, 53, 221, 34, &RPT, 221, 225);
                for (j = 0; j < i; j++)
                {
                    chr = b[j];
                    inline(221, 229, 221, 42, &RPT, 58, &chr, 207, 54);
                    inline(221, 225);
                    UFIA.HDOB++;
                }
                _MODE = 'w';
                return &UFIA
            }
        if (!RPT) return NULL;
        _MODE = *mode;
        return &UFIA;
    }
}

```


DD

HiSOFT-C OP DE DISCIPLE

DD

```

        if (_MODE != 'w') return EOF;
        chr = c;
        inline(221, 229, 221, 42, &_RPT, 58, &chr, 207, 54);
        inline(221, 225);
        UFIA.HDOB++;
        return chr;
    }

    static char chr;
    FILE *fp;
    {
        if (_MODE != 'r') return EOF;
        if (UFIA.HDOB == 0) return EOF;
        inline(221, 229, 221, 42, &_RPT, 207, 60, 50, &chr);
        inline(221, 225);
        UFIA.HDOB--;
        return chr;
    }

    fclose(fp)
    FILE *fp;
    {
        if (!_MODE) return NULL;
        if (_MODE == 'w')
        {
            inline(221, 229, 221, 42, &_RPT, 207, 56, 221, 33, &UFIA);
            inline(207, 59, 62, 3, 207, 54, 1, &UFIA.HDOB, 10, 207, 54);
            inline(3, 10, 207, 54, 207, 62, 221, 225);
        }
        _MODE = 0;
        return &UFIA;
    }

    dir(name)
    char *name;
    {
        int i;
        if (_MODE) return NULL;
        UFIA.DSTR1 = 1;
        UFIA.SSTR1 = 2;
        UFIA.LSTR1 = 'd';
        UFIA.HDOO = 2;
        move(UFIA.NSTR2, " ", 10);
        for (i = 0; i < 10 && *name; UFIA.NSTR2[i++] = *name++);
        _RPT = 0;
        inline(221, 229, 221, 33, &UFIA, 207, 51, 207, 67, 56, 4);
        inline(221, 34, &_RPT, 221, 225);
        if (!_RPT) return NULL;
        return &UFIA;
    }

    erase(name)
    char *name;
    {
        int i;
        if (_MODE) return NULL;
        UFIA.DSTR1 = 1;
        move(UFIA.NSTR2, " ", 10);
        for (i = 0; i < 10 && *name; UFIA.NSTR2[i++] = *name++);
        _RPT = 0;
        inline(221, 229, 221, 33, &UFIA, 207, 65, 56, 4);
        inline(221, 34, &_RPT, 221, 225);
        if (!_RPT) return NULL;
        return &UFIA;
    }
}

/* END of DISC.LIB */

```

SP

ONTHOUD UW PINCODE MET EEN ZELFGEKOZEN WOORD

SP

Eindelijk heb ik dan ook eens een pincode aangevraagd. Ik heb er welliswaar nooit behoefte aan gehad, maar ... je weet maar niet!

Zo werd ik dus ook geconfronteerd met het welbekende feit dat je zo'n pincode natuurlijk moet kennen om hem te kunnen gebruiken, dus onthouden, zonder hem op te schrijven, uit veiligheidsoverwegingen. Nu is een getal van maar vier cijfers niet zo moeilijk te onthouden - wie kent zijn eigen telefoonnummer niet, wat toch doorgaans meer cijfers telt - maar toch ... letters, een woord zou gemakkelijker zijn. En, laten we nou maar eerlijk wezen, ik kreeg meteen een idee voor een leuk programmeerkaartje ervoor.

Ik had destijds, bij de introductie van pincodes, wel eens iets gehoord of gelezen over hulpmiddeltjes om pincodes te onthouden, maar hoe die werken weet ik niet. Om kort te gaan, op de andere bladzijde van dit artikelletje ziet u het resultaat, waar ik zelf wel tevreden mee was, maar ik ben niet objectief in dezen.

—

TOELICHTING OP HET PROGRAMMA

In regel 1 wordt een string gEDIMd met 7 rijen van 10 posities. Deze wordt gevuld met drie rijen letters (het gehele alfabet) en vier rijen cijfers (telkens alle tien).

Regel 2 drukt dit diagram af met een kop erboven en een tekst.

De regels 4 t/m 9 bevatten een FOR-NEXT-LUS om de rijen 4 t/m 7 achtereenvolgens zo te kunnen verschuiven dat de pincodecijfers erin precies onder de letters van een zelfgekozen woord komen.

Regel 5 wacht totdat het toetsenbord vrij is.

Regel 6 drukt de onderhavige rij BRIGHT af en BEEPT.

Regel 7 wacht totdat u op een toets drukt.

Regel 8 zet, mits u geen ENTER gaf, het laatste cijfer uit de onderhavige rij voorop en laat terugspringen naar regel 6.

Regel 11 geeft u de mogelijkheid om te herhalen, bv om een fout te herstellen of voor een volgende pincode.

Regel 33 drukt het diagram af dat op het scherm staat. Hier kunt u nog printeropdrachten toevoegen, bv om "condensed" te drukken.

—

Merk op dat de vier cijferrijen gewoon op volgorde gerangschikt zijn: "at random" is echt niet nodig en zou dus overbodig zijn. Verder zijn er ook in de computer op geen enkel moment gegevens over uw pincode aanwezig. Sterker nog: u kunt dit instelprogramma in alle openbaarheid gebruiken, iedereen mag meekijken! Tenslotte: de eerste vier letters van een langer woord kan ook.

—

SP

ONTHOUD UW PINCODE MET EEN ZELFGEKOZEN WOORD

SP

HET BASICPROGRAMMA "pincode" LINE 99 - EdW -

```

1 DIM P$(7,10):
  LET P$(1)="ABCDEFGH IJ":
  LET P$(2)="KLMNOPQRST":
  LET P$(3)="-UVW--XYZ-":
  LET P$(4)="6789012345":
  LET P$(5)="6789012345":
  LET P$(6)="2345678901":
  LET P$(7)="1234567890"

2 PRINT "ONTHOUD UW PINCODE DMV EEN WOORD"
  P$(1)      ;" Is CODE uw codewoord,";
  P$(2)      ;" dan vindt u hiernaast";
  P$(3)      ;" uw pincode, 8055, zo:";
  P$(4)      ;" 1e rij onder de C: 8,";
  P$(5)      ;" 2e rij onder de O: 0,";
  P$(6)      ;" 3e rij onder de D: 5,";
  P$(7)      ;" 4e rij onder de E: 5."

3 PRINT AT 12,0;
  "Bedenk nu zelf een      "; P$(1);
  "geschikt codewoord.     "; P$(2);
  "Schuif per rij het      "; P$(3);
  "pincodecijfer onder",
  "de juiste codeletter.",
  "Rij goed: geef ENTER,",
  "schuif: andere toets.",

4 FOR r=4 TO 7
5   IF LEN INKEY$ THEN
6     GO TO 5
7   PRINT AT r+11,22; BRIGHT 1;P$(r): BEEP .1,9
8   LET i=CODE INKEY$:
9   IF NOT i THEN
10    GO TO 7
11  IF i<>13 THEN
12    LET P$(r)=P$(r,10)+P$(r):
13    GO TO 6
14  NEXT r

11 INPUT "0 herhaal""1 druk af"i:
  IF NOT i THEN
    GO TO 3

33 FOR r=1 TO 7:
  LPRINT P$(r): NEXT r:
  GO TO 10

```

```

99>CLOSE #3: OPEN #3;"t";1: RUN : REM
  De opdrachten voor RUN om de printerpoort aan te
  zetten kunt u desgewenst veranderen of weglaten.

```

SP

ONE-LINERTJES

SP

De Sinclair ZX81 kende een heel nare beperking in de BASIC. Per regel mocht slechts een opdracht worden neergezet. Geen enkele andere computer kent zover ik weet een dergelijke beperking. Omdat de ZX81 niet zoveel geheugen heeft en elke regel al twee bytes kwijt is voor het regelnummer, twee voor de lengte en een voor de afsluitende Newline (CR of Enter), valt het schrijven van wat grotere programma's niet mee.

Gelukkig kent de Spectrum die beperking niet. Door een heleboel opdrachten op een regel te zetten, kan soms een heel programma in een regel worden geperst. Wijlen de Sinclair Gids heeft zelfs ooit een wedstrijd uitgeschreven wie de mooiste "one-liner" kon programmeren.

Met wat fantasie heb ik er zelf een paar gemaakt. Kijkt u maar.

Programma 1 (snelfORMAT voor DD).
FORMAT altijd de schijf in drive 1.

```
1 FORMAT d1:  
  PRINT #1;"Wissel disk & druk op een toets":  
  PAUSE 0:  
  GOTO 1
```

Programma 2 (color madness).
Gebruik voor eigen risico!

```
1 FOR t=0 TO 7:  
  BORDER T: BORDER T: PAPER T: CLS : NEXT T:  
  GOTO 1
```

Programma 3 (ROM checksum).
In verband met verschillende versies wordt de checksum alleen berekend en getoond.

```
1 LET cs=0:  
  FOR t=0 TO 16363: LET cs=cs+PEEK t: NEXT t:  
  PRINT "Spectrum ROM checksum: ";cs
```

Programma 4 (RAM spy).
Toont adres, waarde en de ASCII-CODE, mits groter dan 31.

```
1 INPUT "Van: ";s:"Tot: ";e:  
  FOR t=s TO e:  
    PRINT t;TAB 9;PEEK t,CHR$ PEEK t AND PEEK t>31:  
  NEXT t
```

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087

SP

RS232-INTERFACES VOOR DE ZX SPECTRUM

SP

De standaard ZX Spectrum wordt vaak een "kale Spectrum" genoemd. Daarmee bedoelt men: zonder allerlei interfaces, die de computer extra mogelijkheden geven. Wie een RS232-interface wil aanschaffen heeft een vrij ruime keuze, want in de afgelopen jaren zijn er nogal wat op de markt verschenen. Het was Sinclair zelf die het eerste interface ontwierp: de RS232-poort op ZX Interface 1.

Sommige RS232-interfaces hebben slechts een doel: het zijn ofwel printer-interfaces (alleen voor zenden), danwel modem-interfaces (zonder handshake-lijnen). Een RS232-interface dat kan zenden en ontvangen en handshake-lijnen kan gebruiken is natuurlijk handiger; alle gebruiksmogelijkheden zijn in de hardware voorbereid.

De RS232-poort op ZX IF1 heeft precies wat het nodig heeft: twee data-lijnen, twee handshake-lijnen, GND en +9 Volt. De namen van de signalen, de "werkrichting" en de connectorpinnen hebben echter voor veel verwarring gezorgd. Toch heeft uw HCC-Sinclair GG de IF1-pinbezetting overgenomen en ondersteund, vooral omdat IF1 alom gebruikt werd. We geven u de meest begrijpelijke pin-namen:

2	RX	<----	input : Received Data
3	TX	---->	output: Transmitted Data
4	CTS/DSR	<----	input : Data Set Ready
5	RTS/DTR	---->	output: Request To Send
7	GND		
9	+9 volt	---->	output

Elke pin kan slechts in twee toestanden verkeren. Meestal zeggen we: "hoog" of "laag". Daarover straks meer. Wat doet elke pin?

- RX (ook: RD of RXD) ontvangt de data;
- TX (ook: TD of TXD) verstuurt de data;
- pas als CTS/DSR "hoog" is mag de Spectrum data gaan versturen;
- zodra de Spectrum RTS/DTR "hoog" maakt kan hij data ontvangen;
- GND moet altijd doorverbonden worden;
- de +9 volt kan evengoed +5 of +12 volt zijn; het gebruik kan verschillen per toegepaste kabel.

RS232-interfaces op de Spectrum hebben (als het goed is) altijd twee in- en drie output-pinnen. De uitgaande lijnen worden door de Spectrum zelf "hoog" of "laag" gemaakt. Wat de inkomende lijnen betreft moet de Spectrum maar afwachten wat de "andere kant" op de lijn zet (een printer, een modem of een tweede computer).

Bij RS232-kabels zorgen de handshake-lijnen (pinnen 4 en 5) voor de meeste problemen. Let er dan op dat de "andere kant" vrijwel altijd meerdere in- en output-lijnen heeft. Vooral de input-lijnen soms allen om een aansluiting. Pin 9 van de Spectrum komt dan vaak uitstekend van pas, want daarmee kunnen diverse input-lijnen van de "andere kant" "hoog" gemaakt worden. Bijvoorbeeld CD aan PC-zijde (pin 8) moet dikwijls "hoog" zijn. Proberen dus.

SP

RS232-INTERFACES VOOR DE ZX SPECTRUM

SP

De HCC-Sinclair GG ondersteunt acht verschillende seriele interfaces. Ze zijn verschillend, maar in welk opzicht verschillen ze nu eigenlijk van elkaar? We gaan dan uit van drie criteria:

- gebruikte poortadressen
- uitgangsspanning
- hard- of softwarematige seriele omzetting

POORTADRESSEN

Vrijwel alle toegevoegde hardware aan de Spectrum wordt bestuurd via poortadressen. Een bekend voorbeeld is de joystick-standaard van de firma Kempston: het gebruikt poort 31 voor input. Met de BASIC-functie IN 31 of de mc-instructie IN A,31 kan de stand van de joystick worden ingelezen. Is eenmaal zo'n Kempston-joystick-interface aan de Spectrum gekoppeld, dan kan geen andere randapparatuur worden aangesloten die eveneens poort 31 voor input gebruikt, zoals de AMX-mouse. Daarom ook verdragen de Disciple en de AMX-mouse elkaar niet: ze gebruiken allebei poort 31.

Het is dus van belang dat u weet welke poortadressen een bepaald interface gebruikt. Het is zodoende ook mogelijk dat uw Spectrum is opgetuigd met twee verschillende RS232-interfaces, zolang ze niet dezelfde poorten gebruiken voor input en output.

Een probleem is, dat veel randapparatuur niet volledig is uitgedecodeerd. Dat wil zeggen dat die hardware al wordt aangesproken als 1 of 2 bits in de adressering goed zijn. Het bekendste voorbeeld is de ULA in de Spectrum. Officieel wordt de ULA benaderd via poort 254, maar in werkelijkheid is alleen bit 0 van belang; de ULA reageert dus op alle even poortadressen.

Voor hardware-ontwikkelaars is de keuze van poortadressen veelal moeilijk: conflicten met andere randapparatuur moeten zoveel mogelijk worden voorkomen. Sinclair Research Ltd heeft hierin helaas nooit een adviserende of sturende rol gespeeld.

UITGANGSSPANNING

De RS232-richtlijnen beschrijven oa welke spanningen op de connectorpinnen moeten staan bij seriele overdracht van gegevens:

- tussen +3 volt en +25 volt bij verzenden van een "0";
- tussen -3 volt en -25 volt bij verzenden van een "1".

In de praktijk blijkt die negatieve spanningen niet per se nodig te zijn; 0 volt werkt meestal ook. Vandaar dat veel RS232-interfaces op de Spectrum werken met de volgende spanningen:

- +5 volt bij verzenden van een "0";
- 0 volt bij verzenden van een "1".

Een nadeel hiervan is, dat de RS232-kabel niet te lang kan zijn, bv 1-3 meter. Voor veel toepassingen is dat echter geen bezwaar.

SP

RS232-INTERFACES VOOR DE ZX SPECTRUM

SP

SERIELE OMZETTING

In het computergeheugen worden gegevens meestal in byte-vorm opgeslagen, dwz in 8 bits. Bij parallele overdracht worden de bits alle 8 tegelijk verstuurd (vandaar de brede flat cable), bij seriële overdracht gebeurt dat een voor een. Het omzetten en versturen van zo'n byte in 8 losse bits (ook het ontvangen van bits en daarvan weer een byte maken) noem ik: seriële omzetting. Voor dat werkje zijn speciale chips ontwikkeld, meestal UART of USART genoemd. Ze zijn betrouwbaar en "intelligent", zodat besturingssoftware kort kan zijn. Zenden en ontvangen kan tegelijkertijd.

Veelal ontbreekt in RS232-interfaces zo'n USART. Alles wordt dan gedaan door een "zo intelligent mogelijke" MC-routine. Tegelijk zenden en ontvangen is onmogelijk, zodat aan ontvangstzijde soms wat fouten gemaakt worden. Als printer-interface voldoen ze goed (voor communicatie wat minder) en ze zijn meestal niet zo duur.

We onderscheiden dus hard- en software-matige seriële omzetting. Bij de laatste vorm wordt de snelheid afgeleid van de kloksnelheid van de ZX Spectrum (SP48: 3,5 MHz en SP128: 3,546894 MHz). In de MC-routines wordt de meeste tijd doorgebracht met wachten, nl tot de "bit-tijd" om is. Daarna kan het volgende bit ontvangen/verstuurd worden en dan is het weer wachten tot een bit-tijd verstreken is. In de hierna volgende assembly-listing kunt u dat nazoeken.

Hardwarematige RS232-interfaces hebben voor de timing een eigen kristal, bv 2.4576 MHz in het VTX5000-interface. Deze interfaces zijn sneller te maken door het plaatsen van een sneller kristal, bv 4.9152 of 9.8304 MHz. Zo'n kristal kost ongeveer een tientje!

EEN VOORBEELD

Het voorgaande is bedoeld als inleiding op de "RS232-drivers" in deze IMPULS. Het plaatsen van de complete assembly-listings zou weliswaar instructief zijn, maar nogal veel plaats innemen. Daarom volsta ik met de becommentarieerde listing voor het RS232-interface van de ZX Spectrum 128. Wie geïnteresseerd is in een van de andere drivers moet maar een disassembler gebruiken. De opzet van alle softwarematige RS232-drivers is gelijk.

In IMPULS 84-42 staat de assembly-listing van een hardwarematige RS232: die in de VTX5000. De listings voor de VTX711-interfaces hebben dezelfde structuur.

In grote lijnen bestaan alle drivers uit drie delen:

- initialisatie: de routines worden gekoppeld aan het OS;
- TX-deel: een door het OS aangeboden byte wordt verstuurd;
- RX-deel: een byte wordt ontvangen en doorgegeven aan het OS.

SP

RS232-INTERFACES VOOR DE ZX SPECTRUM

SP

* * TORNADO *

; Driver voor RS232 van ZX128. Werkt ook in 48K mode.

; Instelbaar: - baudrate,

; - borderflashkleur,

; - t/b-channel ("t" alleen voor OUTPUT).

; [c] Kees Versluis - 19 oktober 1993

ORG 65200:DUMP60000

START JR INIT ; aanroep-adres

; Nu volgt een tabel tbv samenwerking met de
; installatie-BASIC.

STATUS DEFB "b" ; "b" of "t"-werking

IF_NR DEFB 2 ; interface-nummer

LENGTB DEFW EIND-START ; lengte van de code

TIMER DEFW #000C ; 9600 baud

IOBRDR DEFB #02 ; rood als knipperkleur

SER_FL DEFB #00,#00 ; "run time variable"

INIT LD HL,(23631) ; de routine wordt

LD DE,15 ; gekoppeld aan stream #3

ADD HL,DE ; (org. voor ZX Printer).

LD DE,OUTBYTE ; TX-routine...

LD (HL),E

INC HL

LD (HL),D

LD DE,IN_BYTE ; RX-routine...

INC HL

LD (HL),E

INC HL

LD (HL),D

XOR A

LD (SER_FL),A

INIT2 LD BC,#FFFD ; initialisatie van het

LD A,#0E ; interface (poort 65533)

OUT (C),A ; OUT 65533,14

RET

; Het "Operating System" van de Spectrum biedt het te
; verzenden byte aan in het A-register.

OUTBYTE LD B,A ; bewaar A

LD A,(STATUS) ; "b"/"t"-channel?

OR \$00100000 ; hoofdletter mag ook.

CP "t"

LD A,B ; herstel A

JR NZ,TXBYTE ; "b"? Dan versturen.

FILTER CP #A5 ; "t"? Dan token-conversie

JP NC,#0B52

RES 0,(IY+1)

CP #7F

JR C,NOGRAF

LD A,"?"

CP #0D

JR NZ,NOT_CR

CALL TXBYTE

LD A,#0A

JR TXBYTE

CP #20

RET C

JR NZ,TXBYTE

SET 0,(IY+1)

TXBYTE DI ; geen interrupts

LD D,11 ; 1 startbit, 8 databits,

; 2 stopbits (totaal 11).

CPL ; inverteer A

LD E,A ; bewaar A

CALL INIT2 ; de hardware wordt steeds

; geïntialiseerd; chip is

; ook voor de MIDI-poort.

LD A,(IOBRDR) ; laat de BORDER knipperen

OUT (#FE),A

CALL BREAK ; bekijk of CTS hoog is

IN A,(C) ; (bit 6 van poort 65533)

AND #40 ; en scan de BREAK-toets.

JR NZ,TEST_CTS ;

LD B,#BF ; BC-poortadres (49149)

; In de zendroutine is de waarde van de Carry Flag
; bepalend voor het hoog/laag maken van de TX-lijn
; (bit 3 van poort 49149). Het eerste bit is altijd het
; startbit (TX wordt hoog), dus de Carry wordt geset.

SCF ; startbit versturen.

SER_OUT JP C,EENUIT ; deze lus is zeer tijd-

LD A,#FE ; kritisch! 0 of 1 zenden

JP BITOUT ; moet precies even lang

LD A,#F6 ; duren.

EENUIT NOP ;

DEC HL ;

BITOUT OUT (C),A ; OUT 49149,A

LD HL,(TIMER) ; haal timer op

DEC HL ;

DELAY2 DEC HL ; en wacht tot

LD A,H ; de bit-tijd

OR L ; om is

JR NZ,DELAY2 ;

SRL E ; roteer volgende bit in

; de Carry Flag

DEC D ; tot alle bits (11 stuks)

JP NZ,SER_OUT ; verstuurd zijn.

SP

RS232-INTERFACES VOOR DE ZX SPECTRUM

SP

```

RETURN    EI                ; laat interrupts toe.
BORD_REST LD  A,(23624)      ; haal de oorspronkelijke
          AND  #38           ; BORDER-kleur op
          RRCA              ; en
          RRCA              ;
          RRCA              ;
          OUT  (#FE),A        ; herstel de BORDER-kleur
          RET                ; KLAAR!!

```

; In de ontvangstroutine wordt eerst gekeken of (vanuit
; BASIC) er sprake is van INKEY\$#3 of INPUT #3;n\$ (dus
; een los byte of een string, afgesloten met CHR\$ 13).

```

IN_BYTE   RES  3,(IV+2)      ; ???
          BIT  5,(IV+55)      ; INKEY$ of INPUT
          JR   Z,RXBYTE
INPUT     LD  SP,(23613)      ; ???
          POP  HL
          POP  HL
          LD  (23613),HL
INPUT2    CALL RXBYTE         ; bij INPUT #3;n$ worden
          JR   NC,INPUT2      ; alle bytes binnengehaald
          CP   #0D            ; tot de CHR$ 13 er is.
          RET  Z
          CALL #0F85          ; alle bytes in de string.
          JR   INPUT2

```

; Bij elke aanroep van de RX-routine wordt geprobeerd
; om twee bytes binnen te halen. De zender kan nl. nog
; een tweede byte sturen, omdat de DTR/RTS-lijn pas
; laag gemaakt wordt als het eerste byte geheel
; ontvangen is. Dat (mogelijke) tweede byte wordt
; tijdelijk opgeslagen (in SER_FL) en later opgehaald.

```

RXBYTE    LD  HL,SER_FL      ; het eerste byte van
          LD  A,(HL)         ; SER_FL is 1 of 0
          AND  A              ; 0? dan geen byte meer.
          JR   Z,REC_BYTE    ; 1? maak eerste byte 0
          LD  (HL),#00
          INC  HL             ; en haal het bewaarde
          LD  A,(HL)         ; byte op.
          SCF                ; Carry=1 (hoera, gelukt)
          RET                ; KLAAR

```

; Is er geen bewaard byte meer, dan wordt geprobeerd
; een byte te ontvangen.

```

REC_BYTE  DI                ; geen interrupts
          CALL BREAK          ; BREAK gedrukt?
          LD  DE,(TIMER)      ; DE bevat baudrate-timer
          LD  HL,800          ; 800 ontvangst-pogingen
          CALL INIT2          ; weer hardware-initiali-
                              ; satie

```

```

LD  B,#BF                ; maak DTR/RTS hoog
LD  A,#FA                ; (bit 2 van poort 49149)
OUT  (C),A               ; OUT 49149,A
CALL POLLEN              ; probeer een byte te
                          ; ontvangen
PUSH AF                  ; bewaar byte en Carry
ADD  HL,DE               ; HL=DE (baudrate-timer)
DB_DELAY2 DEC  HL         ; wacht tot de zender
          LD  A,H          ; stopbits gaat sturen.
          OR  L
          JR  NZ,DB_DELAY2 ;
          ADD HL,DE        ; aantal ontvangstpogingen
          ADD HL,DE        ; voor tweede byte is afh.
          ADD HL,DE        ; van baudrate (dan duren
                          ; de stopbits nl. langer).
          CALL POLLEN      ; probeer het tweede byte
                          ; te ontvangen
          JR  NC,END_RS_IN ; geen succes? dan verder
          LD  HL,SER_FL    ; wel succes? sla dan het
          INC (HL)         ; byte op in SER_FL
          INC HL
          LD  (HL),A
END_RS_IN POP  AF         ; haal eerst ontvangen
          EI              ; byte en Carry weer op
          RET              ; KLAAR!!

```

; Het volgende is de eigenlijke RX-routine. Wat gebeurt
; daar allemaal? Eerst wordt het startbit opgewacht.
; Zodra dat er is, wordt 1,5 bittijd gewacht (het
; startbit is dan voorbij plus de "helft" van het
; eerste bit). Daarna wordt het eerste databit gelezen,
; midden in de bittijd dus. Alle volgende bits worden
; gelezen na een heel bittijd. Zodoende wordt voorkomen
; dat er gelezen wordt tijdens een "flank" (de overgang
; tussen twee bits).
; Tenslotte wordt de DTR/RTS-lijn weer laag gemaakt.

```

POLLEN    LD  B,#FF        ; BC=65533
POLLEN1   DEC  HL          ; tel aantal ontvangst-
          LD  A,H          ; pogingen af
          OR  L
          JR  NZ,READ_RS
          LD  B,#BF        ; niets ontvangen.
          LD  A,#FE        ; maak DTR/RTS laag
          OUT  (C),A        ; (bit 2 van poort 49149)
          RET              ; Carry=0 (geen succes)

```

```

READ_RS   IN  A,(C)        ; voor de zekerheid wordt
          RLCA              ; wel vier keer na elkaar
          JR  C,POLLEN1     ; getest of er een
          IN  A,(C)        ; startbit op de RX-lijn
          RLCA              ; staat
          JR  C,POLLEN1     ; (bit 7 van poort 65533)

```

 SP RS232-INTERFACES VOOR DE ZX SPECTRUM SP

```

IN A,(C)      ;
RLCA          ;
JR C,POLLEN1  ;
IN A,(C)      ;
RLCA          ;
JR C,POLLEN1  ;
STARTBIT LD A,(IOBRDR) ; laat de BORDER knipperen
OUT (#FE),A
LD H,D        ; HL wordt de helft van DE
LD L,E        ; (DE=baudrate-timer)
SRL H         ;
RR L          ;
DEC HL        ; en nog iets minder omdat
DEC HL        ; er alweer wat tijd
DEC HL        ; verstreken is
DEC HL        ;
DEC HL        ;
LD B,#80      ; bit 7 van B wordt !!!

```

; Het B-register krijgt nu twee functies: alle ontvan-
 ; gen bits worden erin geroteerd EN dient als teller
 ; (na 8 rotaties staat bit 7 in de Carry).
 ; Na het detecteren van het startbit bevat HL de helft
 ; van de baudrate-timer en er wordt bij BD-DELAY1 1,5
 ; bittijd gewacht. Alle volgende keren bevat HL 0, dus
 ; wordt er bij BD-DELAY1 1 bittijd gewacht.

```

SER_IN ADD HL,DE ; HL wordt HL + timer
LD A,(HL) ; tijdvlul-instructie
BD_DELAY1 DEC HL ; en wacht...
LD A,B ;
OR L ;
JR NZ,BD_DELAY1 ;
DEC A ; A=#FF
IN A,(#FD) ; IN A,#FFFD
RLCA ; roteer bit 7 in Carry
RR B ; roteer Carry in B
JR NC,SER_IN ; totaal 8 maal
PUSH BC ; ontvangen byte op stack
LD B,#BF ; maak DTR/RTS laag
LD A,#FE ; (bit 2 van poort 49149)
OUT (C),A ; OUT 49149,A
CALL BORD_REST ; herstel BORDER-kleur
POP AF ; haal ontvangen van stack
SCP ; Carry=1 (hoera, succes)
RET ; klaar

```

```

BREAK CALL #1F54 ; ROM-routine BREAK-toets
RET C ; geen BREAK? terug
CALL RETURN ; BREAK? herstel BORDER
RST #08 ; geef foutmelding
DEFB #0C ; "BREAK - CONT repeats"

```

EIND END

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Hierna treft u software aan die hoort bij de acht verschillende RS232-interfaces die de SGG ondersteunt. Vanzelfsprekend zijn ze hetzelfde genummerd als de XCOM- en SPECTERM-versies. Met deze drivers kunt u zenden en ontvangen. Ze worden verbonden aan het altijd aanwezige "P"-channel (stream #3), dat oorspronkelijk bedoeld was voor de ZX Printer.

(Disciple-gebruikers moeten in hun "sys-file" aangeven of ze het "P"-channel willen gebruiken voor een parallelle printer of voor de ZX Printer. Voor gebruik van een van de volgende drivers moet in de "sys-file" voor de ZX Printer gekozen zijn.)

Welke BASIC-opdrachten kunt u met deze drivers gebruiken?

- LPRINT of PRINT #3 ; bv LPRINT a\$
- LLIST of LIST #3 ; kies eerst werking als "t"-channel
- INPUT #3 ; bv INPUT #3; LINE a\$
- INKEY\$#3 ; bv LET a\$=INKEY\$#3

Vanuit eigen MC-programma's kunt u deze drivers zo aanspreken:

```
LD    A,3    ; maak stream #3 tot
CALL #1601   ; het "current channel"
en daarna
RST #10    ; voor het versturen van het byte in het A-reg.
of
CALL #15E6   ; voor een ontvangst-poging
JR    C,...   ; Carry=1? Het ontvangen byte staat in het A-reg.
JR    NC,...   ; Carry=0? Geen byte ontvangen
```

Hoe krijgt u zo'n driver aan de praat?

Eerst toetst u het BASIC-programma "serioset" in. Daarmee kunnen alle mogelijkheden ingesteld worden. Daarna volgt het CODE-deel. De hier gepresenteerde vorm is in HEXDATAREGELS. Zet bij het in-toetsen wel DATA na het regelnummer. Voeg dan nog de volgende HEXLOADER toe en geef CLEAR 6e4: RUN.

```
50 LET A=10: LET B=11: LET C=12: LET D=13: LET E=14: LET F=15:
INPUT "beginadres",P: LET S=0: READ B$
60 FOR P=P TO P+LEN B$/3:
POKE P,VAL B$(1)*16+VAL B$(2): LET S=S+PEEK P:
LET B$=B$(4 TO ): NEXT P: PRINT S: READ B$: GO TO 60
```

Het beginadres van alle CODE-delen is gelijk: 65200. De getallen die dan verschijnen dienen ter controle. Na "Out of DATA" kunt u de CODE gaan SAVEN; de lengte ervan staat bij de HexDataRegels.

Hoe kunt u de CODE gaan gebruiken? Dat gaat met drie opdrachten:

```
CLEAR 65199: LOAD "Cserio#"CODE : RANDOMIZE USR 65200
```

Met POKE 65202,CODE "t" of "b" kunt u de OUTPUT-mode instellen: "tokens/tekst" of "binair". De rest wijst zich hopelijk vanzelf.

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

HET INSTELPROGRAMMA "serio" LINE 800 - EdW -

```

1 DEF FN P(A)=PEEK A+PEEK (A+1)*256: LET C=65200:
  LET S=PEEK (C+1)>5: LET F=136419-(1804 AND PEEK 76=2):
  LET N$="""Cserio"+STR$ PEEK (C+3)+""": LET L=FN P(C+4):
  LET M$="serio": LET B$="NOT USED": LET F$=B$
2 LET T=PEEK (C+2): LET T=(T<>CODE "T")+(T<>CODE "t"):
  IF S THEN LET B$=STR$ INT (F/(2+FN P(C+6))+.5):
  LET F$=STR$ PEEK (C+8)
3 PRINT AT 7,0;"SETTINGS OF ";N$'
  ' "1 CHANNEL TYPE", "TB"(T) '
  ' "2 BAUDRATE", B$, '
  ' "3 BORDER FLASH ", F$ '
  ' "4 OPEN #3;"";"TB"(T);"": STOP "'
  ' "5 SAVE ";N$;" CODE ";C;";";L'
  ' "6 SAVE "";"M$;"" LINE 800";
  #0;"?": BEEP .2,5 [? knippert]
4 LET K=CODE INKEY$-CODE "0": IF K<1 OR K>6 THEN GO TO 4
5 INPUT ;: GO SUB K*10: GO TO 2

10 POKE C+2, CODE "BT"(T): RETURN

20 IF S THEN INPUT "2)", B: RANDOMIZE F/B-2:
  POKE C+6, PEEK 23670: POKE C+7, PEEK 23671
21 RETURN

30 IF S THEN POKE C+8, VAL F$+1 AND F$<"7"
31 RETURN

40 RANDOMIZE USR C: CLEAR : STOP

60 INPUT +K;" ) 0 CR / 1-8 MD OD DD / 9 BD ";D:
  GO TO K*100+D AND D>=0 AND D<=9

500 SAVE VAL$ N$CODE C,L: RUN
508 SAVE "*"M";D;VAL$ N$CODE C,L: RUN
509 RANDOMIZE USR 15363: REM : SAVE VAL$ N$CODE C,L
510 RUN

600 SAVE M$ LINE 800: RUN
608 SAVE "*"M";D;M$ LINE 800: RUN
609 RANDOMIZE USR 15363: REM : SAVE M$ LINE 800
610 RUN

800>CLEAR 65199: LET N$="Cserio#"
899 INPUT ,," LOAD """+N$;""";CHR$ 8;CHR$ 8; LINE N$(7);""";
  "0 CR / 1-8 MD OD DD / 9 BD ";D:
  GO TO 899+(1+D AND D>=0 AND D<=9)

900 LOAD N$CODE : RUN
908 LOAD "*"M";D;N$CODE : RUN
909 RANDOMIZE USR 15363: REM : LOAD N$CODE
910 RUN

```

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Met nummer "1" worden ZX-IF1 en SGG-IF1 aangeduid. SGG-IF1 werd destijds ontworpen om modem-software voor ZX-IF1 ook met eenvoudiger hardware te kunnen gebruiken (modem-software maak geen gebruik van de handshake-lijnen). In alle eenvoud gebruikt SGG-IF1 slechts een poortadres: 247. Daarmee worden zowel de data-lijnen als de handshake-lijnen mee bestuurd. (ZX-IF1 gebruikt voor zijn handshake-lijnen poort 239).

De CODE hieronder is alleen bestemt voor SGG-IF1 en beslist niet voor ZX-IF1. ZX-IF1 heeft immers een ROM waarin alle software al aanwezig is. Bovendien is het poortadres voor de handshakelijnen niet gelijk.

DE MC "Cserial"CODE 65200,302 IN HEXDATAREGELS - KeesV -

1	"18 09 62 01.2E 01 0C 00.02 00 00 2A.4F 5C 11 0F"	438
2	"00 19 11 D6.FE 73 23 72.11 38 FF 23.73 23 72 AF"	2014
3	"32 B9 FE D3.F7 C9 47 3A.B2 FE F6 20.FE 74 78 20"	4523
4	"23 FE A5 D2.52 0B FD CB.01 86 FE 7F.38 02 3E 3F"	6435
5	"FE 0D 20 07.CD 04 FF 3E.0A 18 09 FE.20 D8 20 04"	7848
6	"FD CB 01 C6.F3 06 0B 2F.4F 3A B8 FE.D3 FE 2A B6"	10074
7	"FE CD D5 FF.DB F7 1F 30.F8 AF 37 CE.00 D3 F7 54"	12772
8	"5D 1B 7A B3.20 FB 1B AF.CB 39 10 EF.FB 3A 48 5C"	14666
9	"E6 38 0F 0F.0F D3 FE C9.FD CB 02 9E.FD CB 37 6E"	16900
10	"28 16 ED 7B.3D 5C E1 E1.22 3D 5C CD.58 FF 30 FB"	18959
11	"FE 0D C8 CD.85 0F 18 F3.21 B9 FE 7E.A7 28 06 36"	20911
12	"00 23 7E 37.C9 F3 CD D5.FF ED 5B B6.FE 21 20 03"	23076
13	"3E 80 D3 F7.CD 8F FF F5.19 2B 7C B5.20 FB 19 19"	25278
14	"19 CD 8F FF.30 06 21 B9.FE 34 23 77.F1 FB C9 2B"	27374
15	"7C B5 20 03.D3 F7 C9 DB.F7 07 30 F3.DB F7 07 30"	29658
16	"EE DB F7 07.30 E9 DB F7.07 30 E4 3A.B8 FE D3 FE"	32360
17	"62 6B CB 3C.CB 1D 2B 2B.2B 2B 2B 06.80 19 2B 7C"	33601
18	"B5 20 FB 29.DB F7 07 CB.18 30 F2 AF.D3 F7 CD 2D"	35979
19	"FF 78 2F 37.C9 CD 54 1F.D8 CD 2C FF.CF 0C"	37916

POORTADRESSEN

Dit SGG-IF1 gebruikt poort 247, maar is niet volledig uitgedeco-deerd. Er wordt alleen naar bit 3 van de adressering gekeken.

TX : bit 0 van poort 247
 RX : bit 7 van poort 247
 DTR: bit 7 van poort 247
 CTS: bit 0 van poort 247

UITGANGSSPANNING

De uitgangsspanning is op TTL-niveau: 0 en 5 Volt

SERIELE OMZETTING

De seriele omzetting gebeurt software-matig.

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Met nummer "2" wordt het RS232-interface van de ZX Spectrum 128 aangeduid. Alle versies van SP128 hebben dezelfde seriële poort, dus ook de niet in Nederland uitgebrachte ZX128+2A en ZX128+3.

De in SP128-ROM ingebakken software heeft een aantal vervelende kenmerken. Zo is het interface niet bruikbaar in de SP48-stand, is de hoogst in te stellen snelheid 9600 baud en is er officieel geen keuze tussen emulatie van een "b"- of "t"-channel. De hieronderstaande driver heeft al deze bezwaren niet.

DE MC "Cserio2"CODE 65200,336 IN HEXDATAREGELS - KeesV -

1	"18 09 62 02.50 01 0C 00.02 00 00 2A.4F 5C 11 0F"	473
2	"00 19 11 DB.FE 73 23 72.11 4B FF 23.73 23 72 AF"	2073
3	"32 B9 FE 01.FD FF 3E 0E.ED 79 C9 47.3A B2 FE F6"	4513
4	"20 FE 74 78.20 23 FE A5.D2 52 0B FD.CB 01 86 FE"	6669
5	"7F 38 02 3E.3F FE 0D 20.07 CD 09 FF.3E 0A 18 09"	7859
6	"FE 20 D8 20.04 FD CB 01.C6 F3 16 0B.2F 5F CD D3"	9886
7	"FE 3A B8 FE.D3 FE CD F7.FF ED 78 E6.40 20 F7 06"	12744
8	"BF 37 DA 2A.FF 3E FE C3.2E FF 3E F6.00 2B ED 79"	15026
9	"2A B6 FE 2B.2B 7C B5 20.FB CB 3B 15.C2 22 FF FB"	17195
10	"3A 48 5C E6.38 0F 0F 0F.D3 FE C9 FD.CB 02 9E FD"	19283
11	"CB 37 6E 28.16 ED 7B 3D.5C E1 E1 22.3D 5C CD 6B"	21175
12	"FF 30 FB FE.0D C8 CD 85.0F 18 F3 21.B9 FE 7E A7"	23581
13	"28 06 36 00.23 7E 37 C9.F3 CD F7 FF.ED 5B B6 FE"	25812
14	"21 20 03 CD.D3 FE 06 BF.3E FA ED 79.CD A7 FF F5"	28289
15	"19 2B 7C B5.20 FB 19 19.19 CD A7 FF.30 06 21 B9"	29919
16	"FE 34 23 77.F1 FB C9 06.FF 2B 7C B5.20 07 06 BF"	31917
17	"3E FE ED 79.C9 ED 78 07.38 EF ED 78.07 38 EA ED"	34342
18	"78 07 38 E5.ED 78 07 38.E0 3A B8 FE.D3 FE 62 6B"	36564
19	"CB 3C CB 1D.2B 2B 2B 2B.2B 06 80 19.7E 2B 7C B5"	37907
20	"20 FB 3D DB.FD 07 CB 18.30 F1 C5 06.BF 3E FE ED"	40193
21	"79 CD 40 FF.F1 37 C9 CD.54 1F D8 CD.3F FF CF 0C"	42613

POORTADRESSEN

Deze SP128-RS232 gebruikt de poorten 65533 en 49149. Het is mij niet bekend of het volledig is uitgedecodeerd. Sinclair kennende zou dit wel eens niet het geval kunnen zijn.

TX : bit 3 van poort 49149
 RX : bit 7 van poort 65533
 DTR: bit 2 van poort 49149
 CTS: bit 6 van poort 65533

UITGANGSSPANNING

De uitgangsspanning is op RS232-niveau.

SERIELE OMZETTING

De seriële omzetting gebeurt software-matig.

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Met nummer "3" wordt SGG-IF3 aangeduid. Het werd oorspronkelijk ontworpen als gecombineerd joystick-/parallel printer-interface tbv de Beta-Disk. Later werd op eenvoudige wijze een RS232-poort toegevoegd. Een compleet schema staat in IMPULS 72-58.

In de onderstaande driver is de initialisatie van de hardware zo gedaan, dat alle interface-delen ook in combinatie gebruikt zouden kunnen worden. Bv: lees de joystick-stand in en verstuur de stand via de RS232.

DE MC "Cserio3"CODE 65200,328 IN HEXDATAREGELS - KeesV -

1	"18 09 62 03.48 01 0C 00.02 00 00 2A.4F 5C 11 0F"	466
2	"00 19 11 E6.FE 73 23 72.11 4C FF 23.73 23 72 AF"	2078
3	"32 B9 FE 3E.3F D3 5F AF.D3 1F 2F D3.7F 3E F0 D3"	4313
4	"7F 3E 02 D3.3F C9 47 3A.B2 FE F6 20.FE 74 78 20"	6340
5	"23 FE A5 D2.52 0B FD CB.01 86 FE 7F.38 02 3E 3F"	8252
6	"FE 0D 20 07.CD 14 FF 3E.0A 18 09 FE.20 D8 20 04"	9681
7	"FD CB 01 C6.F3 06 0B 2F.4F CD D3 FE.3A B8 FE D3"	12099
8	"FE 2A B6 FE.CD EF FF DB.3F E6 20 28.F7 AF 37 CE"	14797
9	"02 D3 3F 54.5D 1B 7A B3.20 FB 1B AF.CB 39 10 EF"	16578
10	"FB 3A 48 5C.E6 38 0F 0F.0F D3 FE C9.FD CB 02 9E"	18664
11	"FD CB 37 6E.28 16 ED 7B.3D 5C E1 E1.22 3D 5C CD"	20702
12	"6C FF 30 FB.FE 0D C8 CD.85 0F 18 F3.21 B9 FE 7E"	23049
13	"A7 28 06 36.00 23 7E 37.C9 F3 CD EF.FF ED 5B B6"	25185
14	"FE 21 20 03.CD D3 FE 3E.06 D3 3F CD.A6 FF F5 19"	27415
15	"2B 7C B5 20.FB 19 19 19.CD A6 FF 30.06 21 B9 FE"	29273
16	"34 23 77 F1.FB C9 2B 7C.B5 20 05 3E.02 D3 3F C9"	31096
17	"DB 3F 07 30.F1 DB 3F 07.30 EC DB 3F.07 30 E7 DB"	33034
18	"3F 07 30 E2.3A B8 FE D3.FE 62 6B CB.3C CB 1D 2B"	35082
19	"2B 2B 2B 2B.06 80 19 2B.7C B5 20 FB.29 DB 3F 07"	36374
20	"CB 18 30 F2.3E 02 D3 3F.CD 41 FF 78.2F 37 C9 CD"	38382
21	"54 1F D8 CD.40 FF CF 0C"	39456

POORTADRESSEN

SGG-IF3 gebruikt de volgende poortadressen: 31, 63, 95 en 127. Het is niet volledig uitgedecodeerd, maar bekijkt alleen naar de adreslijnen A5, A6 en A7 (bits 5, 6 en 7 dus).

TX : bit 0 van poort 63
 RX : bit 7 van poort 63
 DTR: bit 2 van poort 63
 CTS: bit 5 van poort 63

UITGANGSSPANNING

De uitgangsspanning is op TTL-niveau: 0 en 5 volt.

SERIELE OMZETTING

De seriele omzetting gebeurt software-matig.

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Nummer "4" is het seriële interface van de Opus Discovery. Gemis aan een RS232-poort op de OD was een stimulans voor dit ontwerp, en de behoefte het Seiko RC-1000 Datahorloge te kunnen programmeren. Hardware-matig stelt het weinig voor, maar het werkt even goed als bv SGG-IF1. Het volledige schema staat in IMPULS 74-18.

Software voor installatie in IC6116 stond al in IMPULS 84-49, de software hierna is dus meer volledigheidshalve, voor deze serie.

DE MC "Cserio4"CODE 65200,334 IN HEXDATAREGELS - KeesV -

1	"18 09 62 04.4E 01 0C 00.02 00 00 2A.4F 5C 11 0F"	473
2	"00 19 11 E6.FE 73 23 72.11 51 FF 23.73 23 72 AF"	2090
3	"32 B9 FE C9.CD 08 17 21.03 30 36 00.2D 36 0F 2C"	3312
4	"36 04 2D 36.08 C9 47 3A.B2 FE F6 20.FE 74 78 20"	5039
5	"23 FE A5 D2.52 0B FD CB.01 86 FE 7F.38 02 3E 3F"	6951
6	"FE 0D 20 07.CD 14 FF 3E.0A 18 09 FE.20 D8 20 04"	8380
7	"FD CB 01 C6.F3 06 0B 2F.4F CD D4 FE.3A B8 FE D3"	10799
8	"FE CD F5 FF.CB 76 28 F9.DA 2F FF 3E.08 18 04 3E"	13048
9	"0C 18 00 77.ED 5B B6 FE.1B 1B 7A B3.20 FB CB 39"	14865
10	"10 E6 CD 48.17 FB 3A 48.5C E6 38 0F.0F 0F D3 FE"	16680
11	"C9 FD CB 02.9E FD CB 37.6E 28 16 ED.7B 3D 5C E1"	18918
12	"E1 22 3D 5C.CD 71 FF 30.FB FE 0D C8.CD 85 0F 18"	21046
13	"F3 21 B9 FE.7E A7 28 06.36 00 23 7E.37 C9 F3 ED"	23051
14	"5B B6 FE 01.20 03 CD D4.FE CD F5 FF.CB CE CD B0"	25780
15	"FF F5 19 2B.7C B5 20 FB.19 19 19 44.4D 21 02 30"	27239
16	"CD B0 FF 30.06 21 B9 FE.34 23 77 F1.FB C3 48 17"	29389
17	"0B 78 B1 20.03 CB 8E C9.CB 6E 28 F4.CB 6E 28 F0"	31468
18	"CB 6E 28 EC.CB 6E 28 E8.3A B8 FE D3.FE 62 6B CB"	34011
19	"3C CB 1D 2B.2B 2B 2B 06.08 19 2B 7C.B5 20 FB 3A"	35203
20	"02 30 07 07.07 CB 19 10.F0 3E 08 32.02 30 CD 46"	36203
21	"FF 79 2F 37.C9 D7 54 1F.D8 CD 42 FF.CF 0C"	38173

POORTADRESSEN

Een bijzonderheid van de OD is dat de besturing niet plaatsvindt via I/O-poorten, maar via "Memory Mapped In/OUT". Daardoor vormt de OD-hardware dus nooit een beperking voor andere hardware!

TX : bit 2 van adres #3002
 RX : bit 5 van adres #3002
 DTR: bit 1 van adres #3002
 CTS: bit 6 van adres #3002
 5 V: bit 3 van adres #3002

UITGANGSSPANNING

De uitgangsspanning is op TTL-niveau: 0 en 5 volt.

SERIELE OMZETTING

De seriële omzetting gebeurt software-matig.

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Nummer "5" is de RS232-poort op het Disciple Disk-interface. De hardware is minimaal, getuige het schema in IMPULS 74-18. Het is ontworpen om ook seriele printers op de DD te kunnen aansluiten. Een vervelende complicatie in het hardware-ontwerp is, dat maar een input-lijn op de printerpoort aanwezig is (BUSY). Voor volledige handshake is evenwel een tweede input-lijn (CTS) nodig.

In de driver hieronder wordt voor zowel RX als CTS dezelfde lijn gebruikt. De synchronisatie kan hierdoor dus wel eens in de war raken. De remedie: kies een lagere baudrate voor de transmissie.

DE MC "Cserio5"CODE 65200,312 IN HEXDATAREGELS - KeesV -

1	"18 09 62 05.38 01 0C 00.02 00 00 2A.4F 5C 11 0F"	452
2	"00 19 11 D8.FE 73 23 72.11 3A FF 23.73 23 72 AF"	2032
3	"32 B9 FE 3E.80 D3 FB C9.47 3A B2 FE.F6 20 FE 74"	4583
4	"78 20 23 FE.A5 D2 52 0B.FD CB 01 86.FE 7F 38 02"	6522
5	"3E 3F FE 0D.20 07 CD 06.FF 3E 0A 18.09 FE 20 D8"	8026
6	"20 04 FD CB.01 C6 F3 06.0B 2F 4F 3A.B8 FE D3 FE"	10064
7	"2A B6 FE CD.DF FF DB 1F.E6 40 20 F7.37 CE 80 D3"	12648
8	"FB 54 5D 1B.7A B3 20 FB.1B AF CB 39.10 EF FB 3A"	14713
9	"48 5C E6 38.0F 0F 0F D3.FE C9 FD CB.02 9E FD CB"	16946
10	"37 6E 28 16.ED 7B 3D 5C.E1 E1 22 3D.5C CD 5A FF"	18873
11	"30 FB FE 0D.C8 CD 85 0F.18 F3 21 B9.FE 7E A7 28"	21064
12	"06 36 00 23.7E 37 C9 F3.CD DF FF ED.5B B6 FE 21"	23264
13	"20 03 3E 84.D3 FB CD 91.FF F5 19 2B.7C B5 20 FB"	25461
14	"19 19 19 CD.91 FF 30 06.21 B9 FE 34.23 77 F1 FB"	27365
15	"C9 2B 7C B5.20 05 3E 80.D3 FB C9 DB.1F E6 40 20"	29380
16	"F0 DB 1F E6.40 20 EA DB.1F E6 40 20.E4 DB 1F E6"	31714
17	"40 20 DE 3A.B8 FE D3 FE.62 6B CB 3C.CB 1D 2B 2B"	33779
18	"2B 2B 2B 06.80 19 2B 7C.B5 20 FB C6.00 DB 1F 17"	35169
19	"17 CB 18 30.F0 3E 80 D3.FB CD 2F FF.78 37 C9 CD"	37447
20	"54 1F D8 CD.2E FF CF 0C"	38503

POORTADRESSEN

De Disciple is volledig uitgedecodeerd, maar voor dit interface is dat niet relevant. Het werkt immers uitsluitend op de DD.

TX : bit 0 van poort 251
 RX : bit 6 van poort 31
 DTR: bit 2 van poort 251
 CTS: bit 6 van poort 31
 5 V: bit 7 van poort 251

UITGANGSSPANNING

De uitgangsspanning is op TTL-niveau: 0 en 5 volt.

SERIELE OMZETTING

De seriele omzetting gebeurt software-matig.

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Nummer "6" is het RS232-interface in de VTX5000. Het betreft dus niet de complete VTX5000 of het modem-deel. De RS232-print is de groene print in het kastje. Op de bandkabel staan wel de noodzakelijke RS232-signalen, maar op TTL-niveau. In IMPULS 72-09 werd beschreven hoe de signalen op RS232-niveau zijn te brengen. Het VTX5000-RS232-interface kan dan werken als een algemeen toepasbaar serieel interface. Ook werd daar uitgelegd, hoe een schakelaar aangebracht kan worden om verschillende baudrates te selecteren. Voeg daarbij de optie om een sneller kristal te plaatsen. en je hebt het best denkbare RS232-interface voor je Spectrum.

DE MC "Cserio6"CODE 65200,184 IN HEXDATAREGELS - KeesV -

1	"18 04 62 06.B8 00 2A 4F.5C 11 0F 00.19 11 DF FE"	1080
2	"73 23 72 11.20 FF 23 73.23 72 06 03.3E 20 D3 FF"	2516
3	"10 FA 3E 60.D3 FF 3E CE.D3 FF 3E 35.D3 FF C9 47"	4993
4	"3A B2 FE F6.20 FE 74 78.20 23 FE A5.D2 52 0B FD"	7293
5	"CB 01 86 FE.7F 38 02 3E.3F FE 0D 20.07 CD 0D FF"	8974
6	"3E 0A 18 09.FE 20 D8 20.04 FD CB 01.C6 47 CD 54"	10632
7	"1F D2 00 0D.DB FF E6 85.FE 85 20 F2.78 D3 7F C9"	13043
8	"FD CB 02 9E.FD CB 37 6E.28 16 ED 7B.3D 5C E1 E1"	15305
9	"22 3D 5C CD.40 FF 30 FB.FE 0D C8 CD.85 0F 18 F3"	17402
10	"DB FF 0F 0F.38 1F CD 54.1F D2 00 0D.F3 21 80 3E"	19002
11	"3E 37 D3 FF.DB FF 0F 0F.38 05 2B 7C.B5 20 F5 3E"	20837
12	"35 D3 FF FB.D0 DB 7F C9"	22362

POORTADRESSEN

Het VTX-RS232-interface is volledig uitgedecodeerd. De poorten:

poort 127 OUT: schrijven van een byte
 poort 127 IN : lezen van een byte
 poort 255 OUT: beschrijven van Command Reg. en Mode Reg.
 poort 255 IN : lezen van Status Register

UITGANGSSPANNING

De uitgangsspanning is op RS232-niveau.

SERIELE OMZETTING

De seriele omzetting gebeurt hardware-matig door de USART 8251A. Volledige documentatie vindt u in de 'data sheet' van deze chip.

Met nummer "7" wordt het VTX711-interface aangeduid. Het is een ietwat wijzigd Microsource-interface (en daarmee compatible). In IMPULS 84-05 is beschreven dat de connector eigenlijk herbedraad moet worden voor een goede werking. Daar staat ook hoe de baudrate-schakelaar gewijzigd kan worden voor het kiezen van snelheden tot 19200 baud. Er ontstaat dan een uiterst handig interface.

Het oorspronkelijke VTX711-interface gebruikt oa poort 31. Daarmee is het onbruikbaar op de Disciple. In IMPULS 84-05 is tevens beschreven hoe dit interface toch kan samengaan met de Disciple.

 SP DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES SP

Nummer "8" is het aldus gemodificeerde VTX711-interface. Opnieuw bedraden van de D-connector en de baudrate-schakelaar blijft onverkort aanbevolen! Zo krijgt ook de Disciple een echte RS232.

DE MC "Cserio7"CODE 65200,170 IN HEXDATAREGELS - KeesV -

1	"18 04 62 07.AA 00 2A 4F.5C 11 0F 00.19 11 D3 FE"	1055
2	"73 23 72 11.14 FF 23 73.23 72 3E 03.D3 1F 3E 51"	2360
3	"D3 1F C9 47.3A B2 FE F6.20 FE 74 78.20 23 FE A5"	4618
4	"D2 52 0B FD.CB 01 86 FE.7F 38 02 3E.3F FE 0D 20"	6375
5	"07 CD 01 FF.3E 0A 18 09.FE 20 D8 20.04 FD CB 01"	7943
6	"C6 47 CD 54.1F D2 00 0D.DB 3F E6 0A.FE 02 20 F2"	9807
7	"78 D3 5F C9.FD CB 02 9E.FD CB 37 6E.28 16 ED 7B"	12093
8	"3D 5C E1 E1.22 3D 5C CD.34 FF 30 FB.FE 0D C8 CD"	14366
9	"85 0F 18 F3.DB 3F 0F 38.1E CD 54 1F.D2 00 0D F3"	15950
10	"21 80 3E 3E.11 D3 1F DB.3F 0F 38 05.2B 7C B5 20"	17232
11	"F6 3E 51 D3.1F FB D0 DB.7F C9"	18869

DE MC "Cserio8"CODE 65200,170 IN HEXDATAREGELS - KeesV -

1	"18 04 62 08.AA 00 2A 4F.5C 11 0F 00.19 11 D3 FE"	1056
2	"73 23 72 11.14 FF 23 73.23 72 3E 03.D3 B7 3E 51"	2513
3	"D3 B7 C9 47.3A B2 FE F6.20 FE 74 78.20 23 FE A5"	4923
4	"D2 52 0B FD.CB 01 86 FE.7F 38 02 3E.3F FE 0D 20"	6680
5	"07 CD 01 FF.3E 0A 18 09.FE 20 D8 20.04 FD CB 01"	8248
6	"C6 47 CD 54.1F D2 00 0D.DB B7 E6 0A.FE 02 20 F2"	10232
7	"78 D3 F7 C9.FD CB 02 9E.FD CB 37 6E.28 16 ED 7B"	12670
8	"3D 5C E1 E1.22 3D 5C CD.34 FF 30 FB.FE 0D C8 CD"	14943
9	"85 0F 18 F3.DB B7 0F 38.1E CD 54 1F.D2 00 0D F3"	16647
10	"21 80 3E 3E.11 D3 B7 DB.B7 0F 38 05.2B 7C B5 20"	18201
11	"F6 3E 51 D3.B7 FB D0 DB.F7 C9"	20110

POORTADRESSEN

De VTX711 is niet volledig uitgedecodeerd. Hier wordt voor beide versies aangegeven welke poorten in de software gebruikt zijn.

	VTX711	VTX711-mod
CONTROLPORT:	31 OUT	183 OUT
STATUSPORT :	63 IN	183 IN
TX-PORT :	95 OUT	247 OUT
RX-PORT :	127 IN	247 IN

UITGANGSSPANNING

De uitgangsspanning is op TTL-niveau: 0 en 5 volt.

SERIELE OMZETTING

De seriele omzetting gebeurt hardware-matig door de MC6850 ACIA. Volledige documentatie vindt u in de 'data sheet' van deze chip.

 OD LOADEN EN SAVEN VAN ZEER LANGE CODE-FILES OD

Het kan voorkomen dat RAMTOP heel laag moet staan, zeg onder de 24E3 bv, voor een heel lang stuk MC of een voor spelletje uit de tijd dat er alleen nog maar een SP16K bestond. Dan resten er nog geen 250 bytes voor de BASIC en/of voor een (tijdelijk) M-kanaal van 33 bytes, exclusief een blokbuffer tbv het LOADen en SAVEn. En de standaardblokgrootte bedraagt bij een Opus al 256 bytes.

Dit artikel is bedoeld om u ideeën aan de hand te doen teneinde toch te kunnen LOADen en SAVEn. U zult deze oplossingen meestal moeten aanpassen aan de omstandigheden. Als voorbeeld zullen we hier werken met de file: "naam"CODE 24E3,41536 (=65535-23999). Deze gaan we van CR-tape in het SP-geheugen LOADen en vandaar op ODisk SAVEn. Daarna doen we hetzelfde in omgekeerde volgorde.

 VAN CR NAAR OD:

- ```

1 SAVE *1;"naam"CODE 24E3,41536:
 CLEAR 23999: LOAD "naam"CODE

2 CLEAR 29999: OPEN #3;1;"naam"RND1:
 LET H$=INKEY$#3+INKEY$#3+INKEY$#3+INKEY$#3+INKEY$#3:
 POINT #3;6E3: SAVE *#3CODE 3E4,35536

3 LOAD "naam"CODE 3E4:POINT #3;1: SAVE *#3CODE 3E4,6E3:
 POINT #3;1: LPRINT H$: CLOSE #3

```

Regel 1 creeert een diskfile met de juiste naam en header en zet RAMTOP onder het adres waarboven de file van tape geLOAD wordt.

Regel 2 verhoogt RAMTOP, zodat er ruimte is om de diskfile voor IN- en OUTput te OPENen. De 5 significante bytes van de header worden in H\$ opgeslagen (type, filelengte en beginadres). Dan wordt de CODE, op de eerste 6E3 bytes na, naar de file ge-  
SAVED, nadat de POINTER op de plaats voor de header werd gezet.

Regel 3 LOADt nogmaals de file van tape, maar nu op 3E4, waarbij de laatste 6E3 bytes naar de SP-ROM gaan (zonder effect, 64K=0). De POINTER wordt vooraan gezet en de eerste 6E3 bytes worden in de file geSAVED. Hierbij worden de headers allebei overschreven. Dus wordt de POINTER weer teruggezet en de header hersteld met H\$ (en een onbelangrijke CHR\$ 13). Dan wordt de file geCLOSEd.

We veronderstellen dat er zich op een disk in ODrive 1 een file bevindt die geSAVED is met "naam"CODE 24E3,41536 (=65535-23999). Met het volgende programma kunnen we die in het geheugen en vervolgens op CR-tape krijgen.



---

OD                      LOADEN EN SAVEN VAN ZEER LANGE CODE-FILES                      OD

---

VAN OD NAAR CR:

1 CLEAR 29999: LOAD \*1;"naam"CODE 3E4:  
SAVE "hulp"CODE 3E4,6E3

2 OPEN #3;1;"naam"IN :POINT #3;6E3+7:  
OPEN #4;"CODE ":POINT #4;3E4:  
MOVE #3 TO #4: CLEAR #

3 CLEAR 23999: LOAD "hulp"CODE 24E3:  
SAVE "naam"CODE 24E3,41536

Regel 1 zet RAMTOP net onder 3E4 en LOADt daarboven de diskfile, waarbij de laatste 6E3 bytes naar de OD-ROM gaan (geen effect). De eerste 6E3 bytes worden als hulp-CODE-file naar tape geSAVED.

Regel 2 OPENT de diskfile voor INPUT, zet de POINTER net achter de eerste 6E3 bytes (en header!), OPENT dan ook een CODE-kanaal (geheugen), zet die POINTER op adres 3E4 en MOVET de rest van de diskfile daarheen. Daarna worden alle streams geCLEARD, hetgeen extra kanalen opruimt en daarmee ruimte schept voor het vervolg.

Regel 3 zet RAMTOP vlak onder 24E3, herLOADt "hulp" van CR, maar nu op 24E3 ev, en SAVET de file "naam" in haar geheel naar CR.

E H F Weijgers                      -                      H Marsmanlaan 29                      -                      2624 TJ Delft

PC

DE ZX-SPECTRUM-EMULATORS "Z80" EN "SPECTATOR"

QL

De redactie heeft mij gevraagd een artikeltje te schrijven over de SP-emulators die zijn ontwikkeld voor PC en QL. Ikzelf ben auteur van de emulator voor de QL en als zodanig hopelijk redelijk in staat een objectieve en kritische recensie te schrijven over die voor de PC van de hand van landgenoot Gerton Lunter. Het feit dat de QL- en de PC-wereld onafhankelijk van elkaar bestaan, vermindert mijns inziens de kans op gekleurde informatie. Ik zal in dit stukje proberen de specificaties van de emulator voor de PC op te sommen en ze te vergelijken met die voor de QL.

Voor degenen die niet weten wat een emulator is: een combinatie van programmeertechnieken en speciale machine-eigenschappen, waardoor een bepaald computersysteem programma's kan uitvoeren die voor een ander systeem werden geschreven (aldus het PBNA-woordenboek automatisering). Eenvoudiger gezegd: het is een programma dat draait op een computer en het gedrag van een geheel andere computer (met zelfs een andere processor) zo goed mogelijk nabootst. Dat nabootsen moet zo letterlijk mogelijk genomen worden: het toetensbord, het scherm, de diskdrives en verdere randapparatuur gaan zich allemaal gedragen alsof ze direct door die andere computer werden aangestuurd cq uitgelezen. Dat de computer waarop de emulator draait in alle opzichten groter, beter en sneller moet zijn dan de computer die geemuleerd wordt, spreekt voor zich. De meeste moderne computers zijn in principe in staat een SP te emuleren, waaronder dus ook de QL en PC. Nadelen van een emulator zijn de vaak lage(re) snelheid en de kans dat net geen 100% van de bestaande software gebruikt kan worden, maar bv "slechts" 99%. Voordelen kunnen zijn: een beter leesbaar beeldscherm, minder kwetsbare hardware, snellere en modernere opslagmogelijkheden (harddisk), speciale opties die op de originele computer niet beschikbaar waren (bv reset, single-stepping), maar soms ook: hogere snelheid.

Alvorens verder te gaan met dit verhaaltje, moet gezegd worden dat de SP-emulator voor de QL, "Spectator" genaamd, zich nog in een vroegtijdig stadium van ontwikkeling bevindt. Enkele van de mogelijkheden waarvan ik de emulator wens te voorzien zijn nog niet geïmplementeerd, of wel beschikbaar, maar nog onvoldoende getest. De SP-emulator voor de PC (met de verwarrende en weinig originele naam "Z80") is al geruime tijd op de markt en mag zonder meer een goed afgewerkt, goed getest en goed gedocumenteerd product genoemd worden. Na veelvuldig gebruik was mij slechts een enkele uiterst hinderlijke eigenschap opgevallen: terwijl het programma draait loopt de PC-klok veel sneller dan gebruikelijk en als je stopt staat je klok ettelijke uren voor (of is de diepere gedachte: "time flies when you're having fun!").

Beide emulators werken volgens hetzelfde principe: in het PC- of QL-geheugen is een blok van 64 Kb gereserveerd als SP-geheugen. De emulator haalt, beginnende bij adres 0, de Z80-instructies een voor een uit het geheugen en vertaalt ze telkens "real-time" naar een set 8086- of 68000-instructies, die voor de gebruiker hetzelfde resultaat hebben als bij de originele SP onder gelijke



PC

DE ZX-SPECTRUM-EMULATORS "Z80" EN "SPECTATOR"

QL

omstandigheden. Hardwarezaken zoals toetsenbordinvoer en scherm-uitvoer worden door aparte routines verzorgd. Het is zaak om het omzetten van de instructies (het eigenlijke emuleren) zo snel mogelijk te doen, want hoe minder T-cycles er per vertaalslag nodig zijn, hoe sneller de emulator wordt. Een handicap op de QL is dat diens 68000-processor maar weinig op de Z80-processor van de SP lijkt. Neem alleen al de manier waarop een "word" in het geheugen wordt opgeslagen: voor de Z80 staat de lage voor de hoge byte, voor de 68000 is het precies andersom. De 8086 daarentegen, de basisprocessor van de PC's, is in feite de grote broer van de Z80 en er zijn slechts weinig instructies nodig voor een goede omzetting. Dat brengt me meteen bij de snelheid van de emulators. Je hebt een redelijk snelle QL nodig om Spectator op een bevredigende wijze te kunnen gebruiken. Een Gold Card is in feite vereist (16 MHz dus), maar nog sneller zou wenselijk zijn, want met de Gold Card wordt ongeveer een derde van de snelheid van een "echte" SP bereikt. Z80 draait veel sneller. Op een 12 MHz 286 haal je de dubbele snelheid, rond de 60% dus. Op mijn 33 MHz 386dx haal ik zelfs 230%, meer dan het dubbele van de echte SP. Op een PC/XT ligt de snelheid tussen de 10% en 30%, afhankelijk van de klokfrequentie.

Als SP-toetsenbord is dat van een PC of QL alles behalve gemakkelijk bruikbaar ivm de vele functies die een toets kan hebben: een commando, kleine letter, hoofdletter, leesteken, functie, extended commando, grafisch symbool en kleurcodering. De auteur van Z80 lost dit probleem op door alle commando's en symbolen in de handleiding op te sommen en aan te geven hoe je ze bij de PC moet intikken. Dat betekent: afdrucken en bij de hand houden. Spectator bezit hiervoor een hulpscherm: na een speciale toetsaanslag krijg je het complete SP-toetsenbord te zien. Volgens mij werkt dit laatste sneller en efficiënter.

Nog een detail betreffende het toetsenbord is het feit dat de SP geen type-ahead buffer heeft zoals de PC en de QL. Als de emulator langzaam is moet je de toetsen nadrukkelijk indrukken en loslaten voor een goede responsie. Bij serieuze toepassingen werkt dit niet goed. Z80 biedt als oplossing een dubbele interruptsnelheid voor iets betere prestaties. Spectator bezit twee toetsenbordmodes: een met Z80 vergelijkbare directe mode, met name voor spelletjes, en een type-ahead mode voor serieuze toepassingen. In die laatste mode kun je gewoon doortypen, je hoeft niet te wachten tot de SP een aanslag verwerkt heeft.

Het hangt grotendeels van de bereidwilligheid en interesse van de programmeur af welke hardware zoal ondersteund wordt door een emulator. Z80 geeft je de controle over een SP48 met evt (naar keuze) een ZX-Interface-1 en/of SamRam (een door de auteur van Z80 ontworpen uitbreidingskaart voor de SP met oa kraaksoftware en een disassembler). De IF1-emulatie van Z80 werkt prima, maar feitelijk wordt van de hardware alleen de RS232-poort gebruikt. Je hebt geen toegang tot microdrives en ook het netwerk is niet ondersteund. Spectator emuleert eveneens een SP48, maar met een

---

PC DE ZX-SPECTRUM-EMULATORS "Z80" EN "SPECTATOR" QL

---

ZX-Interface-1 (issue 2), waarbij zowel de microdrives als het netwerk gebruikt kunnen worden. Hierover later meer.

Bij aanroep van Z80 kun je een aantal switch-parameters meegeven om het programma aan je eigen wensen aan te passen. Hoewel Z80 uitzoekt welke videokaart is aangesloten, kun je er ook zelf een opgeven: Hercules, Plantronics, CGA en EGA/VGA. Alleen de laatste maakt gebruik van alle acht SP-kleuren plus bright mogelijk. Op de QL is er van verschillende grafische kaarten (nog) geen sprake. Spectator beeldt het SP-scherm af in standaardmode-8, zodat alle acht kleuren zichtbaar zijn, bright wordt genegeerd.

Ook heeft Z80 een aantal switches om aan te geven waar de in- en output van de RS232-poort vandaan gehaald of heen gestuurd moet worden (COM, LPT of file). Op Spectator zit RS232 (van Interface 1) direct verbonden met SER1; opties voor het gebruik van andere poorten of van file zijn nog niet beschikbaar.

Tenslotte heeft Z80 switches voor joystickaansturing, geluid aan of uit, vertragen van de emulator, dubbele interruptsnelheid, snellere flash en afregelen van finesses in de Z80-emulatie. Spectator ondersteunt geen joystick, geluid is altijd uit (oef), vertragen heeft hoegenaamd geen zin, interrupt- en flashsnelheid kunnen via jobprioriteiten ingesteld worden (want multitasking!) en Z80-emulatie heeft geen speciale afregelingen.

Als je als SP-freak (wat ik overigens niet ben) met een emulator gaat werken, is het de vraag of spelletjes je enige interesse zijn. Als je de handleiding van Z80 leest, krijg je stellig de indruk dat de auteur er alles voor heeft gedaan om met name de ontelbare games die voor de SP zijn geschreven allemaal te laten werken. Niet-officiële Z80-instructies, zwevende in/out-poorten, R-register- en correcte LDIR-emulatie; aan alles is gedacht! Het resultaat is er dan ook naar: petje af, want inderdaad draaien zelfs de meest ingenieus beveiligde spelletjes allemaal (bijna) vlekkeloos op Z80. Dat kan ik van Spectator helaas nog niet zeggen... Maar als je zoals ik een programmeur pur sang bent, kom je er met Z80 verder nogal bekaaid vanaf want een eerste vereiste voor een goed gebruik van de emulator als werk- en programmeeromgeving is dat hij een flexibel data-opslagsysteem biedt.

Helaas, de enige opslagmogelijkheid die Z80 biedt, is het maken van een "snapshot" van de gehele 48k RAM plus alle Z80-registers zodat het programma later identiek kan worden teruggeLOAD en voortgezet. Dat lijkt op het eerste gezicht wel voldoende, maar bij nader inzien is dat toch niet zo. Hoe moet je in 's hemelsnaam een disassembler of monitor LOADen om een reeds aanwezig programma te onderzoeken? Hoe plak je twee assembler-sources aan elkaar? Hoe haal je een MS/DOS-tekstfile in het SP-geheugen om die dan in je tekstverwerker te editen? Hoe zet je een zojuist geassembleerde utility op disk zonder de hele assembler mee te SAVen? Hoe kun je twee BASIC-programma's MERGEN? Kortom, snapshots zijn alles behalve afdoende en een beter file-management-



PC

DE ZX-SPECTRUM-EMULATORS "Z80" EN "SPECTATOR"

QL

systeem binnen Z80 is zeer gewenst. Een truuk zou kunnen zijn om data naar de RS232-poort te schrijven en later weer daarvandaan in te lezen, maar dat is vrij onhandig en absoluut niet gebruikersvriendelijk.

Al van meet af is het mijn bedoeling geweest om Spectator van een deugdelijk filesysteem te voorzien. Op dit moment zijn drie opties beschikbaar: (1) het laden van de snapshots die Z80 heeft opgeslagen (enkel en alleen omwille van vlotte data uitwisseling); (2) het onderscheppen van de reguliere tape commando's (Save, Load, Merge, Verify) die hun data naar QL disk schrijven en van disk halen; (3) het onderscheppen van Interface-1 microdrive operaties, zodat ook alle acht Spectrum microdrives direct aangeropen kunnen worden (de inhoud van de microdrive sectoren komt in een QL diskfile te staan). De tape-files die naar disk geschreven worden, zijn met een slechts kleine verandering direct bruikbaar als de bekende 'P' en 'H' files. Over het nut en de haalbaarheid van emulatie van bestaande Spectrum diskdrive systemen wordt nog nagedacht.

Desgevraagd heeft de auteur van Z80 verklaard niet van plan te zijn een andere opslagmogelijkheid dan het "snapshot" in zijn emulator in te bouwen. Zijn belangrijkste argument was het feit dat de virtuele Spectrum binnen de PC op dit moment optimaal is afgeschermd van diens omgeving (de PC dus) en bij een crash van de Spectrum blijft de PC altijd "onaangetast". Bij het onderscheppen van tape commando's wordt die afscherming minder duidelijk. Klinkt heel plausibel, maar zijn argument is inconsequent met de bestaande emulator, aangezien de RS232 stream wel onderschept wordt en gegevens naar diskfile kan sturen (dat is dus een gat in de afscherming). Een ander argument was het feit dat de betreffende ROM routines door bepaalde software soms op niet-officiële wijze worden aangeropen, zodat de emulator teveel "denkwerk" moet verrichten (bv. de filenaam). Mijn tegenargument: dan moet je dat soort derderangs software maar gewoon niet gebruiken op de emulator!

De handleiding die bij Z80 geleverd wordt, geeft informatie over de emulator op alle niveaus: beginner tot expert kunnen volgens mij prima met de tekst overweg. Jammer is alleen dat de auteur niet meteen in het Engels heeft geschreven, want Z80 is een programma met internationale mogelijkheden. Nu is het afzetgebied beperkt tot Nederland, België en willicht een stukje Duitsland. Spectator wordt wel geleverd met Engelse handleiding. De beoordeling of die duidelijk is, laat ik aan de lezer over.

Z80 is deels in C en deels in assembler geschreven; via een menu dat je met F2 oproept, kun je naar DOS shellen. Spectator is geheel in assembler geschreven, werkt met multitasking en is WMan/PtrGen- en Minerva-compatibel.

Z80 en Spectator worden beide verspreid als ShareWare. Registratie voor Z80 (PC) kost 25 gulden en daarvoor krijg je de meest

PC

DE ZX-SPECTRUM-EMULATORS "Z80" EN "SPECTATOR"

QL

recente versie (1.25), een aantal extra utilities en de complete source (maar matig becommentarieerd, dus moeilijk toegankelijk). Registratie voor Spectator (QL) kost 50 gulden (verschil moet er zijn) en daarvoor krijg je een laserprinterafdruk van de handleiding en regelmatig nieuwe, verbeterde versies. U kunt in het bezit komen van een gratis (niet geregistreerde) testversie van Spectator door mij vier geformatte 3.5" DS/DD merkdiskettes te sturen. U krijgt dan de meest recente versie plus enkele SP-programma's teruggestuurd (twee diskettes hou ik voor portokosten).

Voor meer informatie over Z80, de SP-emulator voor de PC, of voor registratie, kunt U zich wenden tot de auteur:  
Gerton Lunter, Johan de Wittstraat 66, 9716 CH Groningen.

Voor meer informatie over Spectator, de SP-emulator voor de QL, voor een test-versie (zie boven), of voor opmerkingen over dit artikelje kunt U zich wenden tot de ondergetekende.

Carlo Delhez - Emmastraat 3 - 4651 BV Steenberg



-----  
SP

ACCENTEN AFDRUKKEN MET TASWORD TWEE

SP  
-----

Als u min of meer regelmatig correspondentie "over de grens" voert, bijvoorbeeld met familie, vrienden en/of zakenrelaties in Duitsland, Frankrijk of Scandinavische landen, weet u dat men in het buitenland gebruik maakt van accenten en "vreemde" letters. Streepjes voor- en achterover, puntjes op - niet alleen - de "I", de cedille, de ringel-S enz.

#### ASCII

Alle hoofd- en kleine letters, cijfers, leestekens en enige besturingscodes zijn gerangschikt in een tabel volgens de ASCII-norm. De eerste 128 elementen (0 t/m 127) bevatten een vast teken, de tweede helft wil nog weleens verschillen. De Spectrum heeft er grafiekjes en toetswoorden, de IBM (compatibele) PC heeft er accenten, enkel- en dubbellijns kaders, wetenschappelijke tekens en vlakvullingen.

#### SCHERM

Op het scherm kan de Spectrum z'n eigen ASCII-tekens met CODES boven 31 afbeelden. Door accentletters onder UDG-toetsen te programmeren kunt u ook accenten afdrukken.

#### ZX-PRINTER

De ZX-printer gebruikt voor het afdrukken van tekst de systeemvariabelen CHARS en UDG. Als u op de Spectrum een andere teken-set maakt en gebruikt en vervolgens LPRINT of LLIST intypt, wordt het "font" waar CHARS naar wijst gebruikt om de naaldjes aan te sturen. Via de UDG's zijn ook accenten afdrukbaar.

#### 80-KOLOMS PRINTER

Een 80-koloms printer wordt met de Spectrum verbonden via een printer-interface met daaraan een kabel. Wanneer u LPRINT of LLIST intypt, worden de ASCII-waarden van de tekens verzonden. De letters, woorden en zinnen worden afgedrukt volgens het patroon dat in de ROM-chip van de printer is opgeslagen. Elk merk printer kent zo z'n eigen teken-opbouw. Als u het niet gelooft, kunt u eens enkele tijdschriften pakken waarin LISTINGS staan afgedrukt. Elke tijdschriftredactie heeft z'n eigen printer en elke printer drukt de letters weer anders af. Het uiterlijk van de tekens van mijn printer verschilt bijvoorbeeld enorm van dat van de tekens die Ed Weygers' printer afdrukt.

Mijn printer, de Star NL-10, biedt de mogelijkheid om softwarematig landcodes te selecteren. Als na zo'n selectie de printer een bepaalde ASCII-waarde in de tekst tegenkomt, wordt een ander teken afgedrukt. Dit werkt zowel in draft mode (kladstand) als NLQ (correspondentiekwaliteit).

Gelukkig zijn de grafiekjes van Tasword Twee volledig (her)-programmeerbaar en is er onder BASIC nog wel wat ruimte vrij.

SP

ACCENTEN AFDRUKKEN MET TASWORD TWEE

SP

Dat biedt mogelijkheden om onder een der grafiekjes de zogenaamde "International Character Set Selector Code"

27 82 nn (decimaal) of 1B 52 nn (hexadecimaal)

te plaatsen. Hierbij vult u voor nn een van deze landcodes in:

dec hex land

|    |    |                 |
|----|----|-----------------|
| 0  | 00 | USA (default)   |
| 1  | 01 | France          |
| 2  | 02 | Germany         |
| 3  | 03 | England         |
| 4  | 04 | Denmark type I  |
| 5  | 05 | Sweden          |
| 6  | 06 | Italy           |
| 7  | 07 | Spain           |
| 8  | 08 | Japan           |
| 9  | 09 | Norway          |
| 10 | 0A | Denmark type II |

In BASIC kunt u bijvoorbeeld een extra menukeuze inbouwen, die u een keuze laat maken uit de landcodes die uw printer ter beschikking heeft.

Op het scherm ziet het er een beetje raar uit, maar op den duur bespaart het u een heleboel werk. U hoeft niet achteraf alle accenten met een zwarte pen bij te tekenen. Houdt u er wel rekening mee, dat de omgewisselde CODEs niet langer beschikbaar zijn als u een andere landcode selecteert. Bij mij zijn de tekens

# \$ @ [ \ ] ° ' { | } ~

dan niet langer beschikbaar.

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087



---

INTERRUPT BIJ DE Z80

---

Een interrupt is een onderbreking van het programma dat op dat moment uitgevoerd wordt. De interrupt dwingt de Z80 om eerst een andere routine uit te voeren. Meestal wordt daarna weer verdergegaan met datgene waarmee de Z80 voor de interrupt bezig was. We zullen een door een interrupt aangeroepen routine in het vervolg interrupt-serviceroutine noemen en het binnenkomen van een interrupt noemen we een interrupt-request.

Waarom eigenlijk interrupt? Ik zal hier een voorbeeld geven voor Input/Output (I/O).

We hebben een printer die alle handelingen die plaatsvinden op een computer afdruckt. De data komt eerst in een buffer.

- GEPROGRAMMEERDE I/O. We moeten telkens het uit te voeren programma onderbreken en mbv van een I/O-routine kijken of de printer in staat is om te printen. Zo'n routine kost tijd en moet elke keer worden uitgevoerd om de printerstatus bij te houden.

- INTERRUPT-I/O. Hierbij geeft de printer aan wanneer hij af kan drukken. We verliezen dan geen tijd met het onderzoeken van de printerstatus, want zodra er een interrupt-request binnenkomt voeren we een interrupt-serviceroutine uit die het afdrukken verzorgt, waarna we gewoon doorgaan met het programma.

Natuurlijk is dit maar een eenvoudig voorbeeld en zijn er vele manieren om gebruik te maken van interrupt. Ook de ZX-Spectrum maakt er gebruik van op het moment dat deze wordt aangezet.

De Z80 heeft vier verschillende interruptlijnen, namelijk:

RESET op pen 26  
BUSRQ op pen 25

NMI op pen 17  
INT op pen 16

- RESET. Dit is geen echte interrupt maar een dwingen van de Z80 om op adres 0000h verder te gaan met het uitvoeren van het programma. Na een reset zijn de interrupt-flipflops, het I- en het R-register gereset en is interrupt-mode 0 geselecteerd.

- BUSRQ (Bus Request). Met een laag signaal op de BUSRQ van de Z80 vraagt een chip of hij de controle van de adres- en databus mag overnemen. De Z80 maakt de lopende instructie af, maakt dan BUS-ACKnowledge (BUSACK) laag en plaatst zichzelf in Tree State, dwz dat de Z80 zichzelf van de bussen ontkoppelt door alle busaansluitingen in hoge impedantie te schakelen. Als BUSACK laag geworden is weet de andere chip, bv een DMA, dat hij de controle van de bussen kan overnemen. Elke T-state test de Z80 de BUSRQ-ingang en op het moment dat deze weer hoog wordt weet de Z80 dat hij de controle over de bussen weer kan overnemen en maakt hij BUSACK hoog. Tijdens BUSRQ worden interrupt-requests op NMI en INT genegeerd en voert de Z80 geen refresh meer uit. Een DMA (Direct Memory Access controller) kan oa zeer snel blok-

---

 INTERRUPT BIJ DE Z80
 

---

ken data verplaatsen binnen het geheugen of tussen I/O zonder de tussenkomst van de Z80.

- NMI (Non Maskable Interrupt). Als er een interrupt-request op de NMI binnenkomt, dan maakt de Z80 de lopende instructie af en springt naar de interrupt-serviceroutine. Deze interrupt is niet te maskeren (Disable) en wordt dus altijd uitgevoerd behalve tijdens BUSRQ.

- INT (Maskable interrupt). Als er een interrupt-request binnenkomt op de INT dan maakt de Z80 eerst de lopende instructie af; daarna kijkt hij of de interrupt-request is toegestaan (Enabled of Disabled) en welke interrupt-mode is geselecteerd. Een interrupt-request op de INT valt dus te maskeren. Wordt niet uitgevoerd tijdens BUSRQ.

Onder INT hebben we drie interrupt-modes die met een instructie te selecteren zijn: IM0, IM1 en IM2. Voor we hiermee verder gaan moeten we dieper ingaan op de twee interne flipflops van de Z80.

DE INTERRUPT-FLIPFLOPS IFF1 en IFF2. We zagen hierboven al dat IM0 t/m IM2 te maskeren zijn. Dit doen we met de instructies enable-interrupt (EI) en disable-interrupt (DI). Hiermee zetten of resetten we IFF1 en IFF2. Zijn deze flipflops geset, dan is er in de Z80 een intern-signaal-interrupt-enable (INTE) geset, waardoor de Z80 een binnenkomende interrupt-request op de INT-pen accepteert. Verder zullen we de interrupt-flipflops nog tegenkomen bij het behandelen van de interruptsoorten en -modes.

#### DE WERKING VAN DE NON MASKABLE INTERRUPT (NMI)

Een NMI request komt binnen op pen 17 van de Z80 en is edge-triggered, dwz dat een dalende flank van het signaal op de NMI-aansluiting een interrupt-request veroorzaakt.

Nu wordt de status van IFF1 opgeslagen in IFF2 en IFF1 gereset, zodat een interrupt-request op de INT-pen niet meer geaccepteerd wordt. De inhoud van de programmateller (PC) wordt op de stack geplaatst en daarna geladen met 0066h, maw de interrupt-serviceroutine van de NMI begint op adres 0066h. Nadat de interrupt-serviceroutine is uitgevoerd wordt deze afgeloten met een RETN-instructie (geen RET). Deze RETN-instructie zorgt ervoor dat de status van IFF1 wordt hersteld door de inhoud van IFF2 terug te kopiëren (een RET-instructie zou IFF1 niet herstellen).

Dus na een interrupt-request op NMI:

- (1) (IFF2) <- (IFF1)
- (2) (IFF1) <- 0
- (3) (stack) <- (PC)
- (4) (PC) <- 0066h
- (5) interrupt-serviceroutine start op adres 0066h
- (6) RETN: (IFF1) <- (IFF2)
- (7) (PC) <- (stack)



---

 INTERRUPT BIJ DE Z80
 

---

## DE WERKING VAN DE MASKABLE INTERRUPT (INT)

Als er een interrupt-request binnenkomt op de INT-pen en de interrupt is enabled, dan genereert de Z80 het INTERRUPT-Acknowledge-sigitaal (INTA). Mbv dit sigitaal weet de interrupt-bron die een interrupt-request heeft veroorzaakt dat deze is geaccepteerd en deze bron plaatst nu een interrupt-vector op de databus. Het INTA is geen bestaand sigitaal of pen aansluiting, het is een EN-functiebeeld van IOREQ en M1. Elke voor de Z80 ontwikkelde chip bevat intern deze EN-poort. Normaal is M1 alleen actief tijdens de opcode-fetch-cyclus samen met MEMRQ. Op de interrupt-vector kom ik nog terug.

## - INTERRUPT MODE 0 (IM0) -

IM0 selecteren we door de instructie IM 0. Deze mode is ook geselecteerd na een reset. Een INT-request komt binnen op pen 16 van de Z80 en is level-triggered, dwz dat een laag sigitaal voor een interrupt-request zorgt.

Nu worden IFF1 en IFF2 gereset. Daarna worden IORQ en M1 actief gemaakt, dus het bovengenoemde-INTA sigitaal. Nu plaatst de interruptbron, na het actief worden van het INTA-sigitaal, een RST- of CALL-instructie op de databus. Na 2 wait-states wordt deze in het instructieregister geplaatst. Is het een CALL-instructie dan moeten er nog twee bytes van de databus worden gehaald. De instructie wordt nu uitgevoerd.

Er gebeurt nu het volgende: de inhoud van de programmateller wordt op de stack geplaatst en daarna geladen met het startadres van de interrupt-serviceroutine. Deze wordt beëindigd met een RETI-instructie. Dit omdat de code van de RETI-instructie op de databus komt te staan tijdens de opcode-fetch. Hieraan kunnen andere interrupt-bronnen zien dat de interrupt is afgehandeld, zodat zij op hun beurt een interrupt-request kunnen genereren.

Dus na een interrupt request in IM0 op INT:

- (1) (IFF1) <- 0 en (IFF2) <- 0
- (2) INTA actief
- (3) 2 wait-states
- (4) RST- of CALL-instructie op databus
- (5) instructie in instructieregister
- (6) RST/CALL: (stack) <- (PC)
- (7) (PC) <- startadres interrupt-serviceroutine
- (8) interrupt-serviceroutine
- (9) RETI: (PC) <- (stack)

Er zijn 8 restart-instructies en elk is opgebouwd uit een vaste waarde met een 3-bits vector. Hieruit wordt het restartadres cq het beginadres van de interrupt-serviceroutine berekend.

|                    |                                                  |
|--------------------|--------------------------------------------------|
| Interrupt-vector   | [X][X][X]                                        |
| Restart-instructie | [1][1][X][X][X][1][1][1]                         |
| Rest-adres         | [0][0][0][0][0][0][0][0][0][0][X][X][X][0][0][0] |

---

 INTERRUPT BIJ DE Z80
 

---

| Restart-instructie | Interrupt-vector | Instructie-code | Adres |               |
|--------------------|------------------|-----------------|-------|---------------|
| RST 00             | 000              | C7              | 0000  |               |
| RST 08             | 001              | CF              | 0008  |               |
| RST 10             | 010              | D7              | 0010  | alle getallen |
| RST 18             | 011              | DF              | 0018  |               |
| RST 20             | 100              | E7              | 0020  | hexadecimaal  |
| RST 28             | 101              | EF              | 0028  |               |
| RST 30             | 110              | F7              | 0030  |               |
| RST 38             | 111              | FF              | 0038  |               |

## - INTERRUPT-MODE 1 (IM1) -

Deze interrupt-mode is identiek aan IM0, met dit verschil dat er geen RST- of CALL-instructie van de databus wordt gelezen (wordt dus genegeerd). Er wordt altijd naar adres 0038h gesprongen.

Dus na een interrupt-request in IM1 op INT:

- (1) (IFF1) <- 0 en (IFF2) <- 0
- (2) INTA actief
- (3) 2 wait states
- (4) negeer RST/CALL-instructie of onbekend op databus
- (5) (stack) <- (PC)
- (6) (PC) <- 0038h
- (7) interrupt-serviceroutine start op adres 0038h
- (8) RETI: (PC) <- (stack)

## - INTERRUPT-MODE 2 (IM2) -

IM2 is eigenlijk DE interrupt-mode op de Z80. Deze is speciaal ontwikkeld om samen te werken met allerlei randapparatuur. Daarom zijn alle Z80-chips voorzien van de mogelijkheid om met IM2 te werken. Dus de Z80-PIO, Z80-SIO, Z80-CTC, Z80-DMA, etc. IM2 is zeer flexibel omdat de gebruiker kan bepalen waar de interrupt-serviceroutines beginnen. De al genoemde 2 wait-states zijn er om ervoor te zorgen dat de chips de tijd hebben om een interrupt-vector op de databus te zetten.

IM2 lijkt op IM0, maar nu wordt er een 8-bits interrupt-vector op de databus gezet. Deze vector is het onderste gedeelte (lob) van een 16 bits vectoradres. Het bovenste gedeelte (hob) wordt gevormd door de inhoud van het interrupt-register (I). Dit vectoradres (I-register + vector) is een adres binnen een vectortabel van maximaal 256 bytes. De inhoud van dit aangewezen adres en het volgende adres is het startadres (restartadres) van de interrupt-serviceroutine. De inhoud van het I-register is het basisadres van de vectortabel.

Omdat een startadres in de tabel twee bytes in beslag neemt zijn er maximaal 128 startadressen mogelijk en daarmee dus 128 verschillende interrupt-serviceroutines. De tabel begint altijd op



---

 INTERRUPT BIJ DE Z80
 

---

een even adres en alweer omdat ieder startadres in de tabel twee bytes lang is bezit een door een interrupt-bron geleverde 8-bits interrupt-vector altijd een even waarde.

## EEN PRAKTIJKVOORBEELD

We hebben een Z80-PIO, waarvan we poort A gebruiken als printer-interface. Als de printer ready is geeft de PIO een interrupt-request. De vectortabel begint op adres 6000h en het startadres van de printer-interrupt-serviceroutine staat op adres 6004h en 6005h in de vectortabel.

Dus I-register: 60 interrupt-vector PIO-poort A: 04

```

6000 INTVCTRIBL 01 lob = 1001
 10 hob alle getallen
 06 lob = A006
 A0 hob hexadecimaal
 06 lob = E006
 E0 hob
 etc

```

Dus na een interrupt-request in IM2 op INT:

- (1) (IFF1) <- 0 en (IFF2) <- 0
- (2) INTA actief
- (3) 2 wait-states
- (4) lees interrupt-vector van de databus (04)
- (5) bereken adres in vectortabel (60 ++ 04 = 6004)
- (6) (stack) <- (PC)
- (7) (PC) <- startadres interrupt-serviceroutine  
(inhoud 6004 en 6005: E006)
- (8) interrupt-serviceroutine start op adres E006
- (9) RETI: (PC) <- (stack)

## INTERRUPT EN PRIORITEIT

De NMI heeft bij de Z80 de hoogste prioriteit, dwz dat een NMI altijd wordt uitgevoerd, ook al is op dat moment INT actief.

In IM0 kunnen we alle interrupt-bronnen in een vaste volgorde testen om vast te stellen welke bron de interrupt-request heeft geleverd. Deze volgorde van testen bepaalt de prioriteit. Dit heet polled-interrupt. Ook kunnen we gebruik maken van een interrupt-controller die de prioriteit voor ons afhandelt. Deze levert de code voor de RST-instructie en onthoudt interrupt-requests met een lagere prioriteit.

In IM2 maken we gebruik van een daisy-chain. Iedere voor de Z80

---

-- INTERRUPT BIJ DE Z80 --

---

ontwikkelde chip bezit een interrupt-enable-ingang (IEI) en een interrupt-enable-uitgang (IEO). De eerste chip ligt met zijn IEI aan +5V. De IEI van iedere volgende chip is verbonden met de IEO van de vorige. Zo ontstaat er een ketting, daisy-chain genaamd. De chip boven in de ketting heeft de hoogste prioriteit: als deze een interrupt-request genereert dan disablet hij de volgende chip(s) door zijn IEO laag te maken. Daarnaast bestaat er binnen een Z80-chip ook prioriteit. Bij een Z80-PIO heeft poort A een hogere prioriteit dan poort B.

---

SLOTOPMERKINGEN

---

Een interrupt kan op een willekeurig moment plaatsvinden. Begin in de interrupt-service-routine altijd met het bewaren van de registers die in de interrupt-service-routine worden gebruikt en herstel deze aan het eind van de interrupt-service-routine. In IM0 t/m IM2 moeten we na de interrupt-service-routine de interrupt zelf weer enablen, dit in tegenstelling tot NMI.

Met de instructies LD A,I en LD A,R kunnen wij de status van IFF2 testen. De flag P/V bevat de status van IFF2.

Een volgende keer wil ik het hebben over de mogelijkheden van interrupt bij de ZX-Spectrum. Ik wil u nu al waarschuwen dat de ZX-Spectrum op een wat eigenwijze manier omgaat met interrupt.

Wilt U dieper ingaan op de Z80 en de bijbehorende hardware, dan vindt u in het volgende waarschijnlijk nog nuttige informatie.

Z80 microprocessor family.  
SGS electronics.

Z80 applications by James W Coffron.  
Uitgeverij Sybex. ISBN 0-89588-094-6

Z80 assembly language programming by Lance A Leventhal.  
Uitgeverij Osborne / McGraw-Hill. ISBN 0-931988-21-7

Zakboekje Z80 by J B Vonk.  
Uitgeverij Kluwer technische boeken. ISBN 90-201-1808-0

Programming the Z80 / Programmeren van de Z80 by Rodney Zaks.  
Uitgeverij Sybex. ISBN 3-88745-100-7

Cursussen MP/MC en MCI, Dirksenopleidingen te Arnhem.

J Leyenaar - J de B Kemperstraat 49 - 3207 EA Spijkenisse



SP

P-OMZETTERS - DE KIP EN HET EI

SP

Uit databanken kunt u onze programma's meestal downloaden in de vorm van zg P-files: pakketjes met BASIC, CODE en handleidingen. Om P-files uit te pakken en de files op uw eigen opslagmedium te zetten is een P-omzetter nodig (om in te pakken ook natuurlijk). Omdat er verschillende opslagsystemen bestaan hebben we vijf van zulke programma's, gepubliceerd in IMPULS 81-20, 82-14, 91-10 en 91-29 (aanvulling). Ze staan dus ook op cassette (IMPULSOFT 15), disk (DUCDISK-21 en DDiSC-21) en als P-files in onze databank:

"SINCLAIRBOX" (ZIE DE COLOFON)

"CRPFILER.P" bevat "cr">"p" (BASIC met de CODE in C\$)

"ODPFILER.P" bevat "od">"p" (BASIC met de CODE in C\$)

"DDPFILER.P" bevat "dd">"p" (BASIC) en "Cdd">"p" (CODE)

"MDPFILER.P" bevat "md">"p" (BASIC) en "Cmd">"p" (CODE)

"BDPFILER.P" bevat "bd">"p" (BASIC) en "Cbd">"p" (CODE)

Wat moet je echter met P-files als je nog niet uitpakken kunt? Daartoe dient nu de sinds kort eveneens te downloaden CODE-file "P2CR.CDE", waarmee P-files uit te pakken zijn op cassetteband:

- LOAD "P2CR.CDE" op adres 25000
- LOAD een P-file op adres 25100
- pak uit met RANDOMIZE USR 25E3

Voer dit uit met een der bovengenoemde P-files om de P-omzetter voor uw eigen systeem op een bijbehorend opslagmedium te krijgen (zet vooraf de teller op 0). Was dat "CRPFILER.P", dan bent u nu klaar: "cr">"p" staat op tape. Voor de andere omzetters spoelt u de tape terug tot voor de uitgepakte files (0-stand) en geeft u

- LOAD "" om de BASIC van tape te LOADen

Zodra de BASIC geLOAD is wordt, behalve bij "od">"p", gepoogd om de CODE van disk of cartridge te LOADen. Geef eventueel BREAK en

- LOAD ""CODE om de CODE van tape te LOADen
- RUN 99 om de buffer te legen en het menu te krijgen

Tenslotte met de P-omzetter voor uw eigen opslagsysteem in uw SP (verander wanneer nodig met optie 6 (<> DRIVE) het drivenummer):

- LOAD met optie 3 (LOAD P) "??PFILER.P" in de P-buffer
- pak met optie 2 (SAVE P>N) de omzetter uit op uw medium

NB: P-omzetters zijn heel goed bruikbaar als kopieerprogramma's: met optie 1 (LOAD N>P) BASIC-, ARRAY- en CODE-files inpakken en met optie 2 (SAVE P>N) deze daarna weer uitpakken bij een andere disk, cartridge of tape (elke P-omzetter kan dit nl ook via CR).

E H F Weijgers - H Marsmanlaan 29 - 2624 TJ Delft

---

BB      SNEL TEKSTPASSAGES OP EEN DISK ZOEKEN MET BETA-BASIC      DD

---

Er zijn waarschijnlijk maar heel weinig computers waarvoor zoveel verschillende opslaginterfaces bestaan als voor de ZX-Spectrum. Het begon met Sinclairs eigen ZX-Interface 1 voor MicroDrive. Daarna kwamen er systemen als de WafaDrive, de BetaDisk, de Opus Discovery en zelfs een apparaat om een Spectrum te kunnen koppelen aan een drive van rivaal Commodore. Een van de laatst uitgekomen interfaces is het Disciple-interface.

In het land van Sinclair Research zit ook nog de firma Betasoft. Zij zijn, terecht, beroemd geworden door hun BASIC-uitbreidingskit die zij "Beta BASIC" noemden (net zo origineel trouwens als TasWORD van TasMAN software). Door uitbreiding van SinclairBASIC met Beta-BASIC ontstaat een BASIC die met recht S-BASIC mag worden genoemd.

Door het per sector uitlezen van de diskette kan een schijfje doorzocht worden op het voorkomen van een bepaald woord. En dan kan het gemak de programmeur dienen. De Disciple is een behoorlijk snel diskinterface en Beta-BASIC is, behalve flexibel en gebruikersvriendelijk, ook nog zeer snel. Door dus in Beta-BASIC een programma te schrijven dat een diskette track na track en sector na sector afstruimt op zoek naar een door u op te geven tekenreeks, wordt het resultaat kort en snel. Het is het beste van twee snelle werelden. Het programma maakt gebruik van Beta-BASIC-opdrachten voor het snel en vakkundig doorlichten van een stukje Spectrum-geheugen, waarin mbv een standaard Disciple-opdracht een sector is geplaatst. Onder in beeld wordt bijgehouden met welke track en daarbinnen met welke sector het programma bezig is.

U dient te beschikken over Beta-BASIC versie 3 +D (niet te verwarren met 3 D voor de Opus Discovery!). Of het programma ook op een 128K Spectrum draait weet ik niet, omdat ik alleen een Plus heb. U typt of leest het programma in en typt RUN. In z'n huidige vorm gaat het uit van 80 tracks DS/DD, dus wie iets anders heeft moet even wat spitten. Het beste resultaat krijgt u door of hoofdletters of kleine letters als zoekargument op te geven. Anders dan kan het gebeuren dat ook Beta-BASIC niet alle plaatsen kan vinden, zelfs niet, zoals ik heb gedaan, met drie SHIFT-modes (Spectrum en SPECTRUM blijven dan "ongelijk!").

Om wegscrollen van het beeld te voorkomen moet u eigenlijk WINDOWS programmeren. Dat mag u echter zelf doen. Dit programma is bedoeld als IMPULS voor het gebruik van de combinatie DD/BB en als demonstratie van "hoe programmeer ik mijn Spectrum zonder GOTO-opdrachten".

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087



---

BB      SNEL TEKSTPASSAGES OP EEN DISK ZOEKEN MET BETA-BASIC      DD

---

HET BETABASICPROGRAMMA "BBDDTZ" - RW -

```

100 LET dn=1
110 DO WHILE dn<>0
120 Inv
130 Exe
140 Rep
150 LOOP
190 STOP

200 DEF PROC Inv
210 CLS#
220 LET ok=0
 WHILE ok=0
 INPUT "Doorzoek schijf in drive (1/2;0=Esc):";dn
 IF dn<0 OR dn>2 OR dn<>INT (dn)
 THEN LET ok=0
 ELSE LET ok=1
230 LOOP
290 END PROC

300 DEF PROC Exe
310 INPUT "Zoekargument: ";z$
320 CLS
 PRINT "Doorzoek schijf #";dn;" op"z$: PRINT
330 FOR t=4 TO 207
340 FOR s=1 TO 10
350 PRINT #1;AT 0,0;"Doorzoek sector ";USING "###";s;
 " van track ";USING "####";t
360 FOR m=1 TO 3
370 LOAD @dn,t,s,3e4
380 LET x=1
390 IF INSTR(1,MEMORY$(3e4 TO 30512)
 ,SHIFT$(m,z$))<>0 THEN
 PRINT USING "Track ##";t;USING "Sector ###";s
400 NEXT m
410 NEXT s
420 IF t=80 THEN
 LET t=128
430 NEXT t
490 END PROC

500 DEF PROC Rep
510 INPUT "Nog een schijf doorzoeken (j/n) ";x$
520 IF x$="n" OR x$="N" THEN
 LET dn=0
590 END PROC

```

SP

## TOOLKITS VOOR DE SPECTRUM

SP

De SP is van huis uit voorzien van een behoorlijk goede BASIC. Kleur, geluid, tekst en grafiek, rekenen, het zit allemaal in het kastje. Maar soms blijken er toch dingen te ontbreken. Op zo'n moment wordt het tijd, te gaan kijken naar programma's die het programmeren eenvoudiger maken.

Door de jaren heen zijn er verschillende "handige hulpprogramma's" uitgebracht. Daarvan wil ik er drie onder de loep nemen. Ik zal dat doen in de volgorde Bruikbaar-Beter-Best. De namen van de te bespreken programma's luiden, in deze volgorde: ZXED, Supercode III en Beta BASIC 3.0. Al lezende zal u ongetwijfeld het waarom van deze volgorde duidelijk worden. Ik heb overigens geen van die programma's kunnen testen met een SP 128.

## ZXED

ZXED is een zuivere toolkit. Daarmee bedoel ik, dat dit programma alleen functies bevat die het programmeren eenvoudiger maken. Het is duidelijk een vrij oud beestje. De allereerste Spectrums werden namelijk opgeleverd met slechts 16 Kb RAM. Pas later kregen ze allemaal 48 Kb mee. Van het ZXED-hulpprogramma bestaat nu een speciale 16-Kb-versie. Er bestaat dus een standaard 48-Kb-versie en een 'gestripte' (niet Sam Fox poker) voor de 16 Kb SP. Beide versies zijn echter bruikbaar met een 48 Kb SP. En ze zijn ook allebei verplaatsbaar in het geheugen. Als u dus maar enkele functies nodig hebt zet u gewoon de 1.675 bytes lange 16-Kb-versie zo hoog mogelijk in het 48 Kb geheugen. Hebt u een kort spaghetti-programma vol GOTO's van ca 6 Kb, dan kan de 3.612 bytes lange ZXED bovenin het geheugen mooi orde in de chaos scheppen. Zelfs al wilt u per se op uw oude rubber 16 Kb SP werken.

## PAKKET

Het ziet er allemaal wat goedkoop uit. Het gebruikelijke bandje met een vrij dun handboekje, in een klein model boterhamzakje.

## HANDBOEK

Zoals gezegd, vrij dun en klein. Maar het betreft een eenvoudig hulpprogramma, of beter gezegd een verzameling eenvoudige hulpprogrammatjes, waarover gewoon niet zo veel valt uit te leggen.

## TOEGANG

Zodra ZXED in het geheugen zit, krijgt u toegang tot de diverse functies door RUN 9900 <ENTER> in te tikken. Alle eventueel door het programma gebruikte variabelen gaan dan verloren. Bovendien gebruikt ZXED de BASICregels 9900-9916. Als uw programma toeval-  
lig ook wat regels heeft in die nummering, mag u ze eerst handmatig hernummeren. Zeker voor de 48 Kb uitvoering vind ik dat een grove misser. Er staat u bij een 48 Kb SP zo'n 40 Kb ter beschikking. Zelfs de standaardversie gebruikt slechts 3,5 Kb voor de MC en nog wat BASIC. Waarom dan niet ietsje meer MC en geen beslag op BASIC? En dan ipv een variabelenvernietigende RUN 9900 een USSR-aanroep. Nog mooier is natuurlijk een programma dat be-



-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

kijkt of u op de onderste regels niet een sterretje intypt, om daarna direct over te springen naar een menu. Bovendien kent ZXED niet de mogelijkheid van een deel SAVEn. De in regel 9900 t/m 9916 zittende toegangsBASIC wordt dus meegesaved. De enige oplossing is om na het programmeren tot in de kleine uurtjes ZXED z'n eigen BASIC te laten vernietigen en pas daarna het programma te SAVEn. Nogmaals, een blunder.

ZXED voorziet u van de volgende programmeerhulpjes oftewel toolkits oftewel utilities (de met een \* gemerkte komen niet in de beperkte versie voor):

\*) ALTER doorloopt uw programma van regel X tot regel Y. Zoekt naar een opgegeven reeks en vervangt die door iets anders. In- en uitschakelbare vervangbevestiging zit er echter niet bij. U zou dan trouwens ongetwijfeld met uw (wijs)vinger op de Y-toets moeten blijven zitten tot u vingerkramp hebt.

\*) APPEND biedt u de mogelijkheid, om aan lange regels iets toe te voegen zonder eerst met Cursor-naar-rechts vijf minuten lang naar het einde van uw regel te hoeven sjokken. Na het intypen van de toe te voegen tekenreeks vraagt ZXED u net zo lang om een regelnummer waar die reeks achter moet, tot u regel nul intikt. Voor het inkorten van lange regels is APPEND echter ongeschikt.

BYTES zit in beide versies. Deze functie houdt u op de hoogte van de hoeveelheid RAM die u nog voor BASIC beschikbaar hebt. U kunt echter net zo goed in BASIC even

PRINT 65535 - USR 7962

intikken. Met de stelling in de handleiding dat de 16 Kb-versie alleen de belangrijkste hulpjes bevat, ben ik het hier oneens.

\*) COPY kopieert regel X tot regel Y naar achter regel Z. De regelnummers achter GOTO-, GOSUB- en dergelijke instructies worden niet aangepast. Daarvoor moet u de RENUM hebben (zie verderop). In het gekopieerde blok regels loopt de nummering op met stappen van 1. Als het moet, wurmt COPY de te kopiëren regels tussen de al bestaande door die te hernummeren. Dat wordt op het beeldscherm gemeld.

DELETE wist regel X tot regel Y uit uw BASIC-programma. Vul niet per ongeluk voor X of Y een getal tussen 9900 en 9916 in!

\*) FIND zoekt het hele BASIC-programma af naar de tekenreeks die u hebt opgegeven. Iedere regel waarin die reeks voorkomt, wordt op het beeldscherm getoond. Tot besluit wordt ook het aantal getoonde regelnummers afgedrukt. Het aantal keren dat de opgegeven tekenreeks werd gevonden echter niet. Dat getal is echter altijd groter dan of gelijk aan het aantal regels, want het aantal 'voorkomens' per gemelde regel is immers minstens gelijk aan 1.

-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

LINE geeft het adres waar de BASIC-regel met het door u opgegeven nummer begint. Ook dit is weer een voorbeeld van een functie die weinig nut heeft, maar die toch in beide ZXED-versies zit.

\*) MOVE verplaatst een blok BASIC-regels. Werkt net zoals COPY, maar nu worden de oorspronkelijke regels gewist. In de beperkte versie bereikt u uw doel dus met eerst COPY en daarna DELETE.

RENUMBER hernummert uw BASIC-programma. U geeft eerst aan welk deel van de regels lijdend voorwerp is. Daarna met welk regelnummer u opnieuw wilt beginnen en tenslotte de stapgrootte. Daarna is RENUM uw meewerkende voorwerp. Alle regelnummervervijzingen worden meehernummerd. Behalve die waarin een berekening voorkomt. Bijvoorbeeld niet in zo iets als:

```
GOSUB 900+100*(VAL K$-1)
```

SEQUENCE oftewel auto line number(er). Na opgave van het eerste gewenste regelnummer en de gewenste stapgrootte krijgt u na elke <ENTER> een vers regelnummer voorgeschoteld. U hoeft dus alleen nog programmeertekst in te typen. U stopt deze automatische regelnummeruitgifte door het eerstvolgende ongewenste nummer te wissen met Caps Shift Nul of de Delete-toets. Vervolgens typt u Shift A oftewel STOP, gevolgd door <ENTER> en klaar is Kees. Ook zonder deze tikhulp valt nog wel te leven. Uit de 16 Kb-versie had dKtronics dit weg mogen laten.

### GEMIS

Voor het zelf schrijven van BASIC-programma's mis ik een paar hulpjes in deze HulpBijProblemen-doos. Allereerst een variabelen-lister, die de namen en de inhoud van de variabelen geeft. Juist variabelen willen nog wel eens problemen geven. Ten tweede een trace-commando. Om te achterhalen, waarom een bepaald programmadeel 'het maar steeds niet doet', is juist een overzicht van alle programmaregels die het programma 'aandoet' bijzonder nuttig. Het ontbreken ervan is daarom een des te groter gemis.

## SUPERCODÉ III

### NAAM

Met de naam van het pakket ben ik het in ieder geval volstrekt oneens. Supercodé III bestaat uit 152 losse MC-programmaatjes. Daarvan is echter 10 %, oftewel 15 stuks, alleen een beschrijving van een trucje met de Spectrum-ROM. Bladzijde 3 van het handboek begint met te stellen (vertaald): "Supercodé Drie is de fijnste toolkit die voor enige computer ter wereld bestaat". Op bladzijde 11 gaat de maker nog even door en stelt dat "Supercodé III wel concurrenten heeft, maar geen gelijken en zeker geen superieuren". Het woord "valse bescheidenheid" is Mr Freddy A. Vaccha B.Sc kennelijk totaal onbekend.



-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----**PAKKET**

Ik zal Supercode III voortaan SC noemen. SC dus, wordt, zoals gebruikelijk, geleverd op een bandje. Daarbij zit een handboekje in A6-formaat ongeveer. Het geheel zit in een keurig doosje. Wel worden vieze vingers op het doosje erg snel pijnlijk goed zichtbaar door de gebruikte soort papier.

**HANDBOEK**

Aangezien SC zoals gebruikelijk uit het land van de IJzeren Dame komt, is het handboek Engelstalig. Het valt aan te raden, om voordat u het programma gaat gebruiken, eerst even met 15 kwartjes of dubbeltjes langs een schrijf- of kantoorbenodigdhedenwinkel te gaan waar een fotokopieermachine staat die kan vergroten. Het voor de handleiding gebruikte lettertype lijkt wel van een verzekeringspolis. Zo klein zijn die lettertjes. Als u die tekst gaat lezen, bent u snel aan een bril toe (declareren dus, maar let op de kleine lettertjes!). Een kopie van twee kantjes tekst tegelijk op ca. 120 % bespaart u een hoop ellende.

**TOEGANG**

SC bestaat uit 4 blokken. Daarvan zijn het laadprogramma, de eigenlijke MC en een laadschermje identiek. Alleen de demonstratieprogramma's verschillen. Als u het programma overzet (naar CR, MD, OD, BD of DD) hoeft u alleen het CODEblok te kopiëren. Het demonstratieprogramma vertelt u in telegramstijl wat er in het handboek staat. Het handboek werkt echter wel sneller.

**DIAGNOSIS**

Zodra het originele programma is ingelezen, volgt er een scherm waarop staat of het programma een ZX Interface One heeft ontdekt die aan uw Spectrum bungelt. Ik heb het programma met een bandnaar-disk-kopieerprogramma in sneltempo op DD gezet en het toen teruggelezen. Nog steeds hield SC vol dat ik met een bandje had zitten prutsen. Waarschijnlijk "kijkt" SC dus naar de systeemvariabele PROG op 23635/6. Die geeft echter bij de DD nog steeds de standaardwaarde. Alleen door van te voren een ZX-IF1-emulatie uit te voeren wordt PROG verhoogd. Het demonstratie- en installatieprogramma is tegen inbreken en systeemfouten beveiligd. Met de 'E'xit optie van mijn Multiprint druk- en kraakinterface had ik die beveiliging echter al gauw 'buiten gevecht gesteld'.

**SOORTEN**

Zoals ik hierboven al vermeldde, bestaat SC eigenlijk uit 137 (=152-15) MC-programma's. Die vallen onder te verdelen in:

- Toolkit programma's;
- Versnellings- en verfraaiingsroutines;
- ZX Interface One hulpjes.

**TOOLKIT**

Die bestaat uit zaken als:

- regels hernummeren, kopiëren en wissen;
- variabelen overzicht, programma-analyse en trace;
- programma en/of regels inkrimpen of juist uitrekken.

SP

TOOLKITS VOOR DE SPECTRUM

SP

### VERFRAAIING EN VERSNELLING

Is zowat het grootste aanbod. In deze categorie bevinden zich:

- het 'scrollen' van (een deel van) het beeldscherm in alle richtingen, zowel hoge- als lage resolutie. Ook de attributen (PAPER, INK, FLASH, BRIGHT, INVERS) worden niet vergeten.
- geluiden in alle soorten en maten. Zelfs het via de oorplug opnemen van geluid en dat vervolgens weergeven is mogelijk.
- karakterset manipulatie. SC bevat een nieuwe karakterset (naar mijn smaak overal te lelijk voor, maar dat terzijde) en een serie accenten. Voor wie dat nog niet genoeg is, kan een willekeurige tekenset horizontaal en/of verticaal gespiegeld worden of eventueel geroteerd. U moet dan of op uw zij liggend computeren of uw scherm kantelen.
- programmebeveiliging. Het uitschakelen of afvangen van de BREAKtoets, het onderscheppen van systeemfouten, headerless LOADen en SAVEn, het zit erin. Ook het onwrikbaar verankeren van copyrightregels en het listig onLISTbaar maken van programma's komt aan bod.
- conversies. Hexadecimaal naar decimaal en omgekeerd, hoofd-naar kleine letters en omgekeerd.
- diversen.

### ZX INTERFACE ONE

Zowel communicatie via de RS-232-seriepoort als ZX-MD-hulpjes. Het inpagen van de ZX-IF1-systeemvariabelen, een uitgebreide MD-CAT, zenden en ontvangen van bytes en data via de seriepoort, het selecteren van een drive en het wissen van een programmablok zijn dingen die u hier tegenkomt.

### ONHANDIG

Vind ik, dat voor het aanpassen van elke routine allerlei POKES en aanroepadressen tbv USR nodig zijn. Het gebruik van SCROLL-effecten en geluiden vereist zo het overpennen van het halve handboek om niet steeds weer al die getallen te hoeven opzoeken. En allemaal plakkertjes met POKES en adressen op het toetsenbord is ook niet alles. Precies de werking van iedere routine uitleggen heeft dan ook weinig zin. Daar is het handboek voor.

### HANDIG

Dat is naar mijn smaak het feit, dat vrijwel alle routines (zij het soms na enige bewerking) verplaatsbaar zijn. Dat biedt u de mogelijkheid om bijvoorbeeld een programmeertoolsamen te stellen met alleen die routines die u strikt nodig heeft. En als u een spelletje schrijft hebt u binnen het uiteindelijke spel geen boodschap meer aan die tools. U wilt dan scrolling, schermcompressie, geluiden en eventueel beveiliging. Ook in dat geval propt u alleen die routinetjes in een CODEblok.

### ROYALTIES

Als u een programma hebt gemaakt, waarin SC-routines het geheel flitsend maken, wordt u geacht de naam van het SC-programma luid en duidelijk op de verpakking en in het programma te zetten. U hoeft de maker echter geen deel van de eventuele verkoopop-



-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

brengst van uw programma ter hand te stellen. Dat systeem werkt gemakkelijker dan gedoe met licencies en registratienummers. Om die administratie af te handelen, zijn weer extra mensen nodig, dig, die betaald moeten worden. Dat maakt het programma alleen maar (te) duur. En een registratiekaart insturen in een niet portvrije envelop, nou, ons bin zunig...

#### BEVEILIGING

Het demo- en installprogramma biedt u eenmalig de kans een werk-kopie te maken. Daarna zijn origineel en kopie kopieerbeveiligd. U kunt dan alleen nog een kopie maken door de SP te RESETten en de originele SC opnieuw te LOADen. De ONERROR-afhandeling werkt goed bij bandfouten. Of hij ook ZX-IF1 aankan heb ik niet kunnen testen, maar DD-fouten worden niet afgehandeld. Deze kunnen wel worden afgevangen, maar niet met behulp van SC. De inmiddels ter ziele gegane Disciple Nieuwsbrief heeft ooit uit de doeken gedaan hoe het kan en moet. Zoekt en gij zult vinden.

#### BETA-BASIC

##### ONJUIST

Ook de naam van Beta BASIC vind ik niet erg gelukkig gekozen. Beta-versies van programma's worden nl nogal eens gebruikt om de reactie van de ontvanger te peilen, door de (on?) gelukkige en voorlopig exemplaar in handen te geven. Beta BASIC daarentegen, degradeert de in ROM ingebakken SP-BASIC bijna tot beta-versie. Waar SC zichzelf verregaand aanpreept als state-of-the-art MC-toolkit for the Spectrum, unbeatable and superior, ready to make programming easy and exciting en nog wat van dergelijke kreten, blinkt Betasoft uit door bescheidenheid. Hoogstens af en toe een tamelijk kleine advertentie. "Enhanced Basic for the Spectrum", heette het dan, met een opsomming van wat mogelijkheden en waar het verkrijgbaar was en tegen welke prijs. Einde advertentie.

##### VERSIES

De allereerste versie was, hoe kan het ook anders, 1.0. Vervolgens kwam 1.9, zeg maar versie 2. Later kwam 3.0. En toen de Spectrum 128 het levenslicht zag kwam versie 4.0. Toen deze uit was, werd die samen met versie drie verkocht. Omdat ik geen SP 128 ter beschikking heb, kon ik het niet testen, maar ik vermoed dat Betasoft en de importeur voor Nederland dat gedaan hebben om de SP128-gebruikers zowel in de 128-mode als in de 48-stand van dienst te kunnen zijn. Mijn aankoop bestond uit een bandje met versie vier en versie drie erop. In eerste instantie die voor CR en MD. Later heb ik een conversie laten uitveren om Beta BASIC 3 met mijn DD te laten werken. \*)

##### PAKKET

Beta BASIC werd mij geleverd als een soort boterhamzakje, waarin zich een bandje bevond en een handboek. Dat handboek paste maar net in het zakje. Het eruit wurmen kostte enig geweld. Achter op het handboek stond, door het zakje heen goed leesbaar, wat voor

-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

consequenties alleen al het openen van de verpakking heeft, nog voor het bandje zelfs maar in uw CR zit.

#### HANDBOEK

Wat het handboek betreft, eerst maar het goede nieuws. Het is zowel een syntaxis-naslagwerk als een leer- en oefenboek. Vrijwel alles wat erin hoort, staat erin. Dan nu het slechte nieuws. Ten eerste vind ik de rode kleur van het papier nogal onhebbelijk. Bij daglicht is de tekst nog redelijk leesbaar. Maar 's avonds hebt u bij wijze van spreke een mijnwerkershelm met ingebouwde koplamp nodig om nog door te kunnen werken. Ten tweede vind ik de opmaak niet geweldig. Als regel voor het schrijven van werkstukken, boeken, scripties en rapporten geldt, dat men voor ieder nieuw hoofdstuk een nieuwe bladzijde neemt. Zo niet echter het, gelukkig wel Nederlandstalige, Beta-Basic-handboek. Een paar extra regeltjes overslaan vormt de hoofdstukindeling. Daarnaast is het, gezien de noodzakelijke uitgebreidheid van de beschrijving van elke functie, geen overdadige luxe om ook voor elke functie een nieuwe bladzijde te nemen. Ook niet dus. Omdat voor de hoofdstuk- en paragraafindeling bovendien ook nog hetzelfde lettertype is gebruikt als voor de rest van de tekst, is het handboek al met al niet echt prettig bruikbaar.

Omdat BB4 (Beta Basic versie 4.0 dus) slechts een uitbreiding en aanpassing voor de SP128 betreft tov BB3 voor de SP48 (+), heeft Betasoft volstaan met een bijlage voor alle extra mogelijkheden. Voor wat u niet in het BB4 handboek vindt, gelieve u dus ook het handboek van BB3 door te spitten. BB4 heb ik, zonder SP128, niet kunnen bekijken.

Zoals ik hierboven al opmerkte, staat het meeste wat u moet of zou willen weten wel in het handboek. In een geval heeft een stukje informatie dat in het BB3-handboek ontbrak, mij een avond dubbele ergernis gekost. BB3 wordt namelijk met een USR-functie aangeroepen. Maar het programma kan met een andere USR ook weer worden "uitgezet". Pas na minstens een uur zoeken, vond ik de uitzet-aanroep. Niet in het BB3-handboek, maar in de bijlage voor BB4. Daar stonden wat "belangrijke adresjes" en daar stond wat ik zocht. Nog groter werd mijn woede, toen ik vervolgens ontdekte, dat datzelfde adres ook in de BASIC-LOADER stond.

#### SOFTWARE

Beta BASIC bevat zowel een uitgebreide en zeer gebruikersvriendelijke toolkit als diverse extra BASIC-opdrachten. Hoewel ik het zou hebben over toolkits, zal ik toch ook de extra BASIC-opdrachten bespreken. Anders doe ik BB gewoon tekort.

BB en SC overlappen elkaar qua mogelijkheden. SC bevat echter ook een geluidenbibliotheek, beveiligingstips en -trucs en nogal wat MD- en RS232-hulpjes. BB blinkt uit door extra opdrachten voor gestructureerd, oftewel modulair, programmeren. Wat dat precies inhoudt, zal ik hier verder niet uitleggen. Daar zijn ettelijke boeken en cursussen voor, die de materie beter en di-



-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

didactischer uitleggen dan ik het zou kunnen. BB is echter heel erg superieur aan SC, als het gaat over de manier waarop die extra opdrachten te bereiken zijn. Bij SC zit u USSR- en POKE-adressen van buiten te leren alsof het Duitse voorzetsels met de bijbehorende naamval betreft. Bij BB typt u gewoon toetswoorden in zoals u het gewend was bij SP-BASIC. Alleen moet u de toetscombinatie CAPS SHIFT 9 blindelings kunnen vinden.

BB3 bevat, volgens de handleiding, tientallen extra toetswoorden en 26 extra functies. De functieverwijzingen zitten in regel 0, maar om met de functies en/of de opdrachten te kunnen werken, moet en regel 0 en de ca 18 Kb lange MC aanwezig zijn. BB snoept dus van het RAMgeheugen totaal ca 19 Kb af.

De toetswoorden kunnen worden onderverdeeld in:

- Toolkitopdrachten;
- Programmeeropdrachten ter verfraaiing en versnelling.

Om een vergelijking tussen BB3.0 en BB1.9 mogelijk te maken, heb ik voor opdrachten die BB 1.9 niet kent een \* gezet. Als BB iets toevoegt, staat het oude SP-woord in kleine letters. Dat is bv zo bij alle (L)LIST-toevoegingen.

#### TOOLKIT

Die bevat opdrachten zoals:

\* ALTER vervangt een naam of een inhoud van een variabele door niets of iets anders.

AUTO niet de heilige koe, maar voor het automatisch laten afgeven van regelnummers. Startregel en stapgrootte zijn instelbaar.

DEF KEY slaat een aantal opdrachten op onder een toets, vergelijkbaar met een functietoets of een macro. Zo kunt u de lastige toetsvolgorde voor het gebruik van MD vastleggen, maar ook een emulatie van het CLS#-commando. Dit voor het geval u dat niet ter beschikking hebt en u toch BORDER en PAPER wit wilt maken, INK zwart en FLASH, BRIGHT, INVERS en OVER wilt RESETten.

DELETE regel TO regel geeft u de kans overbodige regels te wissen. Met 0 TO 0 kunt u de functieverwijzingen wissen. Handig als u ruimte tekort komt en u de functies toch niet gebruikt.

EDIT biedt u de mogelijkheid een "ver weg gelegen" regel te bewerken zonder eerst met de cursor te moeten knoeien of moeilijk te doen met LIST.

JOIN voegt twee na elkaar komende programmaregels samen. Handig om programmalengte te sparen door zo weinig mogelijk, maar dan extreem lange, regels te maken.

KEYWORDS zorgt ervoor dat u zichzelf, indien gewenst, kunt verlossen van de SP-toetswoord-intik-methode. U kunt opdrachten zoals IN, OUT en INK dus voortaan snel en handig gebruiken. Woor-

-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

den als RANDOMIZE, CONTINUE, RETURN en LIST kunt u echter beter in de geforceerde K-stand alsnog via achtereenvolgens de toetsen T,C,Y en K intypen.

\* List FORMAT maakt de LISTing overzichtelijk. Afhankelijk van de gekozen stand wordt er bij FOR-NEXT-lussen ingesprongen en krijgen alle opdrachten hun eigen regel. Dat geldt ook voor de nog te behandelen PROCedures en If-Then-ELSE constructies.

\* List DATA geeft een variabelendump. De variabelen worden gesorteerd op type en daarbinnen op alfabet (A-Z dus; niet Z-A). Het kan echter nog mooier:

\* List VAL geeft alleen de numerieke variabelen, gesorteerd en wel, met hun resp waarden, terwijl

\* List VAL\$ zich uitsluitend bezighoudt met alfanumerieke ofwel stringvariabelen.

List TO beeldt de programmeregels met door u te bepalen lettergrootte af van regel X tot regel Y. Dit werkt ook prachtig bij niet onder BB geschreven regels, want zo krijg je pas echt mooie printout. Mits u althans NLQ gebruikt met een goed lint.

\* List REF speurt voor u het hele geheugen af, zoekend naar een toetsreeks die u achter REF invult. Handig om te zien of een bepaalde variabelennaam nog vrij is. En voor dat doel zelfs mooier dan DATA/VAL/VAL\$, want dan zoekt u zich alsnog een ongeluk.

\* List DEF KEY toont een overzicht van gedefinieerde toetsen.

RENUM hernummert standaard vanaf regel 10 in stapjes van 10. Maar u mag ook een deel van uw programma hernummeren, zelf een nieuw eerste regelnummer opgeven en/of de stapgrootte wijzigen naar eigen goeddunken. Met een sterretje achter het woord kopieert u regels in plaats van de oude regels te wissen. Dat is nog eens wat anders dan RENUMmen in MicroSoft(R) GW-Basic(R).

SPLIT zit op de ongelijktoets (W dus) en biedt de mogelijkheid regels te splitsen en het gesplitste deel vervolgens een eigen regelnummer toe te kennen.

TRACE toont u, welke programmeregels uw BASIC doorloopt.

PROGRAMMEEROPDRACHTEN. Die vallen uiteen in drie soorten, nl

- OPSLAG,
- VERWERKING en
- PRESENTATIE van gegevens.

- OPSLAG Gaat snel en vooral eenvoudiger dankzij:

\* DEFAULT Waarmee u het opslagmedium kiest. U mag kiezen uit:  
m voor ZX Microdrive



SP

TOOLKITS VOOR DE SPECTRUM

SP

m1 voor ZX Microdrive 1  
nx voor netwerk station x  
b voor het B(inaire) kanaal van de ZX-If1-RS232-seriepoort  
t voor tape (standaardinstelling)

\* Save TO om een deel van uw programma te bewaren ipv alles,  
\* Verify TO ligt in het verlengde hiervan.

\* Save DATA bewaart alleen de variabelen. Het tegenovergestelde dus van een standaard CLEAR: SAVE <..>. Daarmee bewaart u immers alleen het programma, maar niet de variabelen. Met \* Verify TO controleert u of Save z'n werk goed gedaan heeft.

- VERWERKING Is met BB een fluitje van een cent dankzij:

\* JOIN om een tabel, of een deel ervan, over te hevelen naar een andere tabel. Zonder gegevensverlies natuurlijk. COPY heeft er onder BB een functie bij, die sterk lijkt op die van JOIN.

\* DELETE wist overbodige gegevens uit een array, zonder dat wat bewaard moet blijven ook foetsie gaat.

SORT zet arrays op volgorde. Naar keuze van klein naar groot en A tot Z of van groot naar klein en Z tot A. Let wel op het, in dit geval belangrijke, verschil in ASCII-waarde tussen A en a, Z en z en Z en a. U hoeft niet eens een hele tabel te sorteren. Met TO geeft u aan welk arraydeel BB onder handen moet nemen.

INSTRING geeft aan of een ingevoerde string voorkomt in een andere string. Gemakkelijk om, in combinatie met SHIFT\$, menu's te ontwerpen met letterkeuze in plaats van nummertjes.

\* INARRAY is INSTRING voor arrays. Hiermee doorzoekt u dus tabellen op de aanwezigheid van een zoekstring. En razendsnel. Volgens het handboek zet BB zelfs "dure software" (bv FoxBase) op een "kantooromachine" (zoals bv een 80386 PC) in de hoek als het er op aankomt.

CHAR\$ zet gehele getallen om in strings van twee tekens. Zo gaan er extra veel getallen in het werkgeheugen.

NUMBER haalt voor u de getalwaarde van CHAR\$ terug.

- PRESENTATIE van gegevens wordt fraai en eenvoudig mbv:

ALTER om schermkleuren aan te passen zonder een CleasScreen opdracht.

\* CSIZE waarmee u naar keuze zeer grote letters (max 32 bij 22) of hele kleintjes kunt gebruiken, tot wel 85 tekens per regel en 25 tekstregels onder elkaar.

\* Draw TO voor het eenvoudiger PLOTten van grafieken in beeld.

-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

FILL vult snel en eenvoudig elke vorm op uw scherm, bv cirkels, taartdiagrammen, verkeersborden en de puntjes van de letter B.

Poke \$ zet de inhoud van een stringvariabele in het geheugen vanaf het door u opgegeven adres. In combinatie met MEMORY\$ zet BB plaatjes dan zo snel op het scherm, dat zelfs mijn DD ervan bloost.

ROLL beweegt (een deel van) het schermbeeld in elke gewenste richting. U hebt de keuze uit beeldinformatie, attributen of al-lebei. Wat er links afrolt komt rechts weer terug en omgekeerd. Dat geldt ook voor boven en onder.

SCROLL werkt als ROLL, maar nu geldt: weg is weg. Wat eraf rolt komt niet meer terug.

USING om getallen in een "masker" te zetten, zodat ze netjes onder elkaar komen. Ook het aantal van de cijfers achter de punt is instelbaar.

#### SPELLETJES

Ook voor de spelletjesprogrammeurs heeft BB het nodige. Behalve de al besproken grafische opdrachten staan te uwer beschikking:

\* CLEAR zet RAMTOP maximaal 767 bytes hoger of lager, zodat u uw eigen MC boven RAMTOP kwijt kunt.

CLOCK stelt de interne klok in op een bepaalde tijd. Tevens kan de tijd rechtsboven in beeld meelopen (niet verplaatsbaar). En u kunt een alarmtijd en -actie programmeren.

GET leest het toetsenbord uit als een veredelde INKEY\$. Er wordt nu echter netjes gewacht tot u een toets indrukt.

ONERROR vangt inbraken en systeemfouten af. Er staan u speciale variabelen ter beschikking om te kijken waar de fout optrad en welke fout roet in het eten gooide.

TIMES\$ bevat de waarde van CLOCK. Deze functie is handig om bv de tijd te meten of een eenregelige klok te schrijven die in het midden van het scherm loopt. De klok bevat drie getallenparen en twee dubbele punten, dus met CSIZE 32 komt u een heel eind.

#### STRUCTUUR

De SP is standaard niet ruim voorzien van opdrachten om zonder GOTO's en dus gestructureerd te programmeren. Wel aanwezig is de IF-THEN-constructie. Die is onder BB uitgebreid met ELSE, bv:

```
IF schuldig THEN straffen
ELSE vrijlaten
```

Om van de onafgemaakte FOR-NEXT-lussen en het GOTO-gespring af te komen, kent BB de constructies DO .. LOOP, DO WHILE .. LOOP,



-----  
SP

TOOLKITS VOOR DE SPECTRUM

SP  
-----

DO UNTIL .. LOOP en DO: EXIT IF .. LOOP. FOR-NEXT- en DO-LOOP-lussen mag u nesten zoveel u wilt, bij PROCedures is dat streng verboden. Met LIST FORMAT 2 behoudt u altijd het overzicht.

Voor de bit-peuteraars bestaan er ook nog dec-hex conversies en omgekeerd. In SP-BASIC kunt u BINaire getallen POKEn. Met BB kunt u nu binair PEEKen. Handig voor het bekijken van matrixpatronen of vlagbits.

#### MINPUNTJES

Kent de BB-MC ook. Als u een werkkopie wilt maken, moet u regel 1 en 2 met MERGE terughalen, eventueel wat aanpassen en daarna de boel SAVEN. Waarom kan dat MERGE nou niet worden vervangen door een toetswoord PUT om regel 1 en 2 boven RAMTOP te zetten en een extra GET om ze weer terug te halen? Regel 0, die zoals gezegd de functiedefinities bevat, wordt buiten beeld gehouden. Dat is prima zolang u de BB-regel 0 aanhoudt. Maar uw eigen programma kan ook een regel 0 bevatten. Het is helemaal niet gezegd dat u in zo'n programma die regel 0 nodig hebt. U kunt best alleen wat willen RENUMmen. Een extra boole-opdracht LIST LINE 1 voor toon regel nul of LIST LINE 0 voor verberg regel nul zou erg welkom zijn geweest.

Voorts vind ik het erg jammer dat een paar complete voorbeeldprogramma's ontbreken. Om te leren hoe u optimaal gebruik kunt maken van de extra's die BB u te bieden heeft, zou ik een paar programma's, waarvan de LISTing met toelichting afgedrukt zou moeten worden, geen overbodige luxe vinden. Ik denk dan aan bv:

- Een database met vrije kaartlayout (tbv CSIZE)
- Een - liefst pulldownmenu gestuurd - tekenprogramma, evt in combinatie met
- Een UDG-programma met meerdere banks, combinatie van twee naast/onder elkaar of vier tegelijk, spiegelen, roteren, kopiëren, inverteren, ROLL-optie en deel-SAVE, liefst zodanig dat er ook karaktersets mee gemaakt kunnen worden (5 sets UDG's nodig), dec-hex-bin-notatie en bijbehorende KEYIN- en LPRINT-optie.
- Een spreadsheet voor gehele getallen (cijferadministratie voor school met alle cijfers x 10 tbv CHAR\$) met grafieken.

En tot slot zou ik een extra font, dat 7 van de 8 matrixpixels gebruikt in plaats van de SP-standaard 6, ook op prijs hebben gesteld. Een extra opdracht FONT (1: standaard en 2: BB) hoeft niet veel meer te doen dan CHAR\$ te POKEn.

#### INTERFACES

Voor diverse interfaces kent BB1.9 en BB3.0/4.0 aangepaste versies. Voor DD werk ik momenteel met Beta BASIC Versie 3.00+D. OD-gebruikers kennen 1.9D/3.0D. BB 3+D herkent en onderschept echter geen DD-fouten en gezien de toestand van de SP komt het er waarschijnlijk niet meer van. Omdat, voor zover ik heb kunnen nagaan, de DD-versie iets langer is dan de CR/MD-versie, lijkt

SP2

TOOLKITS VOOR DE SPECTRUM

SP

het erop, dat de aanpassingen als het ware onder het programma zitten vastgeplakt. Dat is een veel elegantere methode dan zeg 1 Kb te reserveren. De HH fabrikanten van diverse interfaces kunnen door Betasoft tenslotte niet worden verplicht, om hun DOS zo te schrijven, dat de voor BB noodzakelijke aanpassingen binnen die 1.024 bytes passen. Als het minder kost, is het zonde van de geheugenruimte. En als het meer kost, worden de mensen van Betasoft verplicht slordig te programmeren. Dan raakt de MC immers versnipperd.

Al met al kun je wel stellen, dat met BB talrijke losse hulpprogramma's naar de rommelzolder mogen.

Het zal u zijn opgevallen, dat de bespreking van BB nogal lang is, zeker ten opzichte van SC en ZXED. En nog steeds ben ik met BB selectief te werk gegaan. Sommige opdrachten heb ik weggelaten. Dat heb ik gedaan om twee redenen. Ten eerste omdat er voor een volledige opsomming het handboek is. Ten tweede omdat als ik BB werkelijk heel diepgaand zou willen doorlichten, deze Impuls meteen geheel gevuld zou zijn. Dat is wat teveel van het goede.

#### CONCLUSIE

Nog even de belangrijkste plus- en minpunten van de besproken pakketten in volgorde van doorlichting.

##### ZXED

- + Kort programma, is altijd wel plaats voor
- + Relocatable MC
- Fors beslag op BASIC
- Geen Trace

##### SUPERCODE

- + Veel routines
- + Demonstratie- en Hulpprogramma tbv verplaatsen sommige routines
- + Mogelijkheid tot samenstellen eigen toolkit door samenvoegen van routines
- Veel POKE- en USR-adressen te onthouden
- Handboek in priegelschrift

##### BETA BASIC

- + Zeer uitgebreid
- + Zeer gebruikersvriendelijk
- + Speciale, aangepaste versies voor diverse interfaces
- + Zowel geschikt als zuivere toolkit als voor luxe programmeren
- Handboek (kleur papier en layout)
- Om eruit te halen wat er in zit: vrij lange leertijd

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087



-----  
SP

## DE FUNCTIE VAL\$ - TOEPASSING 2

SP  
-----

In IMPULS 82-20 werd een toepassing van VAL\$ gepubliceerd, met de vraag naar andere gebruiksmogelijkheden. Ik had er eentje.

In een programma van mij kwam een constructie in deze trant voor (om diverse redenen kon ik slechts ongeDIMde strings gebruiken):

```
xx10 IF c=1 THEN LET x$=a$
xx20 IF c=2 THEN LET x$=b$
xx30 IF c=3 THEN LET x$=c$
...
t/m
...
xx90 IF c=9 THEN LET x$=i$
```

Na vele hoofdbrekens vond ik een vrij simpele manier om al deze negen regels door een enkele regel te vervangen:

```
xx10 LET x$=VAL$ "abcdefghi$(2*c-1 TO 2*c)
```

Eindelijk een nuttige bestemming voor de VAL\$-functie, waarvan het nut mij tot nu toe ontgaan was!

[Voor de volledigheid wil ik hier nog een mogelijkheid aangeven:

```
xx10 LET x$=(a$ AND c=1)+(b$ AND c=2)+ ... +(i$ AND c=9)
```

Die regel van Wil is evenwel veel korter en veel eleganter. red]

Maar nu het omgekeerde, ook dat deed zich voor:

```
xx10 IF c=1 THEN LET a$=x$
xx20 IF c=2 THEN LET b$=x$
xx30 IF c=3 THEN LET c$=x$
...
t/m
...
xx90 IF c=9 THEN LET i$=x$
```

Naarstig experimenteren, oa ook met VAL\$, leverde hiervoor geen bekorting op. Wie helpt? Ikzelf kom niet verder dan deze weinig fraaie constructie:

```
xx10 POKE PEEK 23637+256*PEEK 23638+5,96+c
xx20 LET a$=x$
```

Mbv de systeemvariabele NXTLIN wordt hierdoor het programma zelf veranderd, hetgeen een niet erg aanbevelenswaardige techniek is.

Wil G Haggenburg - J v Lennepkade 157' - 1054 ZL A'dam

-----  
SP

DE ZX-PRINTER ZOEMT WEER

SP  
-----

Als scholier kreeg ik te maken met informatica. De school had voor dat vak Spectrums aangeschaft. Die dingen fascineerden mij en zo'n computer moest ik ook hebben. In de winkel mocht ik kiezen tussen wat spelletjes en een Seikosha GP-50S printer. Ik nam de printer.

#### SCHOOLSOFTWARE

Vervolgens besliste de school dat de informaticales gevuld moest worden met programmeren, tekstverwerking, database en tekenen. Voor dat laatste werd een volslagen onbekend en Franstalig programma gebruikt, dat niet eens met kleur kon werken. Voor de database werd Masterfile ingezet (volgens leerboeken betekent dat "stambestand", niet "superprogramma", maar goed...) Nu haal ik meer uit dBASE op een PC dan uit Masterfile op een Spectrum, maar dat terzijde. Voor tekstverwerking werd Spectral Writer van band GELOAD. Zelf heb ik later voor Tasword Twee gekozen en dat programma zelf maar vertaald.

#### VERSIES

Juist van DE Tekstverwerker voor de Spectrum, Tasword Twee dus, zijn talloze "eigen" versies in omloop. In wijlen de Sinclair Gids heeft de heer Schoneveld een volledig omgewerkte Tasword-BASIC beschreven, in RAM (oude stijl, Willem Bos) heeft Jaap Kuiper tips en trucs beschreven. Voor de eveneens wijlen Disciple Nieuwsbrief heeft dezelfde Jaap Kuiper een volledig nieuwe Tasword-BASIC gemaakt en van elke versie bestaan vast en zeker talloze varianten.

#### TYPE > PRN

Nu ondersteunen TW2 en SW beide zowel de ZX-Printer (en de GP50) als Centronics A4 printers. Maar de afdruk die in beide gevallen op de ZX-Printerrol komt, is een exacte kopie van het scherm. Lelijk dus. Beide printers werken, evenals het scherm, met 32 tekens per regel. Beide tekstverwerkers geven 2 x 32 tekens per regel. Om toch een acceptabele afdruk te krijgen, maakte Filo-soft ooit Tasdwars. De tekst verscheen dan in drie lange stroken gekanteld op papier. Die oplossing zat mij dwars en ik heb mijn eigen printmanier gemaakt.

Ik beschouwde een TW2-tekst als twee stroken van 32 tekens naast elkaar. Omdat een afdruk op de ZX-Printer gemaakt wordt met de characterset waar CHARS naar wijst, is de uitvoer afhankelijk van hoe de letters gevormd zijn (elke byte bestaat dan immers uit een BIN xxxxxxxx patroon). Bij een Centronics-printer wordt doorgegeven welk teken moet worden afgedrukt, waarna dat met een vast lettertype wordt gedaan, al dan niet vet of cursief. De benodigde afdrukroutine in BASIC is voor TW2 en SW bijna gelijk. Alleen de startadressen van de CODE zijn anders (32000 / 32768).



-----  
SP

DE ZX-PRINTER ZOEMT WEER

SP  
-----

## OUDE OPLOSSING

Bij TW2 had ik de eerste hulppagina herschreven, zodat die op negen regels paste. De bovenste 12 x 64 bytes bevatten een gewone, 768 bytes lange karakterset. De BASIC heb ik herschreven, zodat de printer codes-instellingen zijn vervangen door een "Laad karakterset" deel en de oorspronkelijke afdruk-BASIC door de ZX-Printroutine. Naar BASIC voerde nu een RAND USR 0 uit.

De oude oplossing had nog een groot voordeel. Verschillende spellen en tape-magazines bevatten een nieuwe tekenset. Door die tekensets eruit te lichten en te bewaren, zou u een Tasprint kunnen maken met zoveel fonts als u maar te pakken kunt krijgen of kunt aanmaken. U hoeft alleen CHARS aan te passen. De Impulsversie gebruikt de Spectrum ROM tekenset en dus geen enkele POKE, want die set wordt standaard gebruikt.

Bij de oude versie zou de nieuwe hulppagina en de nieuwe BASIC worden meegeleverd. In de nieuwe opzet gaat het wat simpeler.

## NIEUWE OPLOSSING

Op verzoek van de heer Weijgers heb ik de door mij gevonden oplossing wat aangepast voor de publicatie. De software zal nu op papier in plaats van op schijf uitkomen.

Ik heb de BASIC voor mijn eigen "Tasword GP" vanuit het niets herschreven. Met enig extra werk is het programmadeel voor het in BASIC afdrukken van de tekst echter overal bruikbaar. U dient slechts wat regelnummers aan te passen.

Om te beginnen zoekt u in uw TaswordBASIC alle regels op die het afdrukken van tekst via niet-ZXPrinters regelt. Die moeten er eerst uit.

Lees nu eerst het stuk tekst onder "LET OP" en typ vervolgens deze LISTing over:

```
300 REM menu voor load, save, print enz.
4000 CLS: PRINT TAB (10); "Print tekst"
4010 LET a$ = "linker" : GOSUB 4100
4020 LET a$ = "rechter": GOSUB 4100
4090 GOTO 300

4100 GOSUB 4300
4110 FOR l=1 TO a STEP 64
4120 IF INKEY$="q" OR INKEY$="Q" THEN GOTO 300
4130 FOR c=1 TO 32
4140 LET char=PEEK (32000+l+c-2+(32 AND a$="rechter")):
 IF NOT char THEN LET char=32
4150 LPRINT CHR$ char;
4160 NEXT c
4170 NEXT l
4180 GOSUB 4400
4190 RETURN
```

-----  
SP

DE ZX-PRINTER ZOEMT WEER

SP  
-----

```
4200 PRINT AT 10,1; "De "; a$; "kolom wordt geprint"
4210 PRINT #1; FLASH 1;" Druk Q om te stoppen "
4290 RETURN
```

```
4300 CLS
4310 FOR f=1 TO 20: LPRINT: NEXT f
4320 PRINT AT 10,6; "SCHEUR DE STROOK AF""
 TAB 7;"DRUK OP EEN TOETS"
4330 PAUSE 0: CLS
4390 RETURN
```

#### LET OP

Regel 300 heeft u niet over te nemen. Hij staat ertussen om aan te geven, dat alle regels "GOTO 300" betekenen "ga terug naar het menu". Als dat bij u regel 400 is, vervangt u overal de opdracht "GOTO 300" door "GOTO 400".

Als bij uw versie de printafdeling toevallig andere regelnummers heeft, typt u de bovenstaande regels over met de voor u juiste regelnummers. Past u wel op met GOTO's en GOSUBs!

#### WRITER

Voor Spectral Writer geldt in grote lijnen hetzelfde. Als u de BASIC toch gaat aanpassen, kunt u ook proberen om de Wafadrivesyntax aan te passen voor uw Opus, Disciple of MicroDrive. Als u geen verkortingen gebruikt heeft u slechts een syntax nodig!

#### TEN SLOTTE

Het afdrukken geschiedt nu in BASIC en dat betekent langzaam. U spaart echter wel geld uit voor Tasprint, Tasdwars en/of een Centronics A4 80-koloms printer.

Weet iemand waar TW2 en SW (Spectral Writer, niet SpecWord) hun kleurinstellingen bewaren, zodat ze in BASIC aan te passen zijn?

Kan iemand mij helpen aan het oorspronkelijke programma van Tasman waarmee TW2-teksten naar TW3-formaat kunnen worden omgezet en vice versa? Via Faas heb ik ooit een op zich werkende versie van Tasword III verkregen, maar wegens de onmogelijkheid tot formaatconversie ben ik nooit overgestapt, hoezeer mij de editor van TW3 ook beter bevalt dan die van TW2.

Wist u dat u met TW2, door Uitvullen en Woordomslog uit te zetten, prima software kunt vertalen? U leest gewoon de machinecode in vanaf adres 32.000 en u speurt de bytes vervolgens af op zoek naar leesbare tekst. Die wijzigt u en als u alles hebt gehad dan SAVET u de CODE weer. Hebt u een vermoeden dat er ook maar een



SP

DE ZX-PRINTER ZOEMT WEER

SP

enkele byte werd gewijzigd die moet blijven zoals hij was, begin dan helemaal opnieuw. SAVE de CODE op een ander bandje of op een schijf onder een andere naam. Als blijkt dat de vertaalde CODE vastloopt hebt u dan het origineel nog; met daarbij de zekerheid dat u opnieuw moet beginnen omdat u een fout hebt gemaakt zonder het op te merken. De "R.Justify" en de "WordWrap" moeten "uit" staan, omdat anders bytes kunnen verschuiven, zodat bij het later RUNnen van de CODE de bytes op de verkeerde plaats staan. Gevolg: @\$%&!\* een hangup, een crash! Controleer dat de te bewerken MC niet langer is dan 20 Kb of 20.480 bytes. Is de CODE langer, dan loopt de TW2-MC zelf gevaar ( $32.000+20.480=52.480$ ).

U kunt een dergelijke "tekst" maar beter niet afdrukken met TW2. Niet via "Extended Mode P", niet via de hier afgedrukte GP-50 printeroutine en niet via de standaard afdrukoutine. De kans dat er iets zinnigs uitkomt is nihil. Waarschijnlijk komt er een hoop papier en een hoop gepiep uit de printer. Als u de papierberg inspecteert, blijkt er weinig op te staan en wat er op staat slaat nergens op. Spaar dus uw humeur, uw portemonnee en het kwetsbare, schaarse regenwoud!

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087

MD

FILE NOT FOUND - REPARATIE

MD

Een vervelende foutmelding, zeker indien u niet over een recente kopie van uw programma beschikt. Bij het inlezen vanaf cartridge behoeft u toch niet alles opnieuw in te typen, want doorgaans is het meeste wel goed ingelezen en ontbreekt er slechts een enkele tapesectie die door ??? vervangen is. Wanneer u deze vraagtekens maar kwijt zou kunnen raken hoefde u slechts een klein gedeelte opnieuw te typen. Jammer genoeg faalt de normale delete-methode, maar gelukkig is er een oplossing die uit deze stappen bestaat:

- Zoek met LIST de laatste incomplete regel.  
We noemen het regelnummer daarvan even xxi.
- Zoek de eerste complete regel na alle ???.  
We noemen het regelnummer daarvan even xxc.
- Voeg ergens voor de ??? de volgende programmaregels toe,  
vul daarbij de gevonden regelnummers in en geef RUN xx1.

```
xx1 DEF FN P(A)=PEEK A+PEEK (A+1)*256
```

```
xx2 RANDOMIZE xxi: LET A=FN P(23635): GO SUB xx5
```

```
xx3 RANDOMIZE xxc: LET B=A+4: GO SUB xx5
```

```
xx4 RANDOMIZE A-B:
```

```
POKE B-1,PEEK 23670: POKE B,PEEK 23671: STOP
```

```
xx5 FOR A=A TO FN P(23627):
```

```
IF PEEK A=13 THEN
```

```
IF PEEK (A+1)=PEEK 23671 AND
```

```
PEEK (A+2)=PEEK 23670 THEN RETURN
```

```
xx6 NEXT A: STOP
```

- Nu kunt u de incomplete regel deleten door regelnummer xxi in te typen gevolgd door ENTER. De ??? zijn dan verdwenen.

#### TOELICHTING

Subroutine xx5 doorzoekt het programma van A tot VARS met terugkeer zodra de regel gevonden is waarvan SEED het nummer bevat. Regel xx1 ontbindt regelnummer xxi in SEED, zet adres PROG in A en roept de subroutine aan.

Regel xx2 ontbindt regelnummer xxc in SEED, zet het adres van de tweede lengtebyte van regel xxi in B en roept de subroutine aan. Regel xx3 ontbindt tenslotte het verschil van de adressen van de regels xxc en xxi minus 4 in SEED en POKEt dit in de lengtebytes van regel xxi, die daardoor alle vraagtekens gaat omvatten.

Een advies aan mijn medeslachtoffers van deze "FILE NOT FOUND"-errors: maak backup-kopieën!

Wil G Haggenburg - J v Lennepkade 157' - 1054 ZL A'dam



-----  
SP

DE BLAST SPECTRUMBASIC COMPILER

SP  
-----

BASIC is een vrij gemakkelijk te leren computertaal, die voor veel inzetbaar is. Daarom heet hij ook "Beginner's All-Purpose".

#### STANDAARD

BASIC is een wijdverbreide taal. Dat komt omdat hij bij de meeste computers is ingebakken (homecomputers) of wordt meegeleverd (PC). Om de computer reeksen opdrachten te geven, hoeft er dus niet eerst (veel) extra geld te worden uitgegeven voor een los aan te schaffen taal. Zo blijft de computer(hobby) betaalbaar.

#### LANGZAAM

BASIC is dus voor veel toepassingen inzetbaar. Vervelend is echter wel, dat het een betrekkelijk langzame taal is. De reden is, dat SpectrumBASIC een interpreter is. In het Engels betekent dat "tolk" en dat is ook precies wat gebeurt. Zodra u uw programma laat uitvoeren, wordt er voor elke opdracht in ROM gekeken wat er moet gebeuren en ook of een programmaregel misschien een grammaticale oftewel syntaxisfout bevat. Als een programmadeel 1000 keer wordt doorlopen, wordt die controle dus 1000 keer verricht. De computer is immers snel en dom. Die weet na elke regel niet meer, wat hij aan eerdere opdrachten heeft uitgevoerd. En duizend kleine vertraginkjes maken een fiks oponthoud.

#### SNELLER

Om uw programma's te versnellen zijn er meerdere oplossingen te bedenken.

De eerste is, dat u rechtstreeks in Z80-machinecode (MC) leert programmeren. Dat is lastig en omslachtig. U hebt speciale invoerprogrammatuur nodig die zowel hexadecimale weergave kent als disassembleerfaciliteiten. Het is min of meer verplicht om voor u uw maaksel probeert, het eerst te SAVEn. Een foutje laat de computer meestal vastlopen en dan moet u weer van voren af aan beginnen. Voordeel is, dat programma's in MC absoluut de snelste zijn. Een ROMtest wordt mbv pure MC binnen enkele seconden uitgevoerd. In BASIC wordt dat nachtwerk.

De tweede is, dat u overstapt op een nieuwe taal. Voor de SP is bv ook C, FORTH en PASCAL beschikbaar. PC-mensen zijn hier trouwens in het voordeel. Ook als u namelijk in PASCAL op een SP programmeert, blijft het ROM-geheugen met de BASIC gewoon aanwezig, ook al hebt u daar niets aan. Een PC beschikt kaal alleen over DOS of een ander besturingssysteem als "omgangstaal", zonder overbodige ingebouwde BASIC-ballast. Bij zo'n andere taal typt u dan uw programma in en dat wordt vervolgens door een compiler omgezet in een brok MC. Een eenregelig LOAD-programmatje ervoor en klaar. Nadeel is, dat u na iedere fout het gehele programma opnieuw moet compileren. En u moet bereid zijn om een nieuwe taal te leren (en dan meteen modulair programmeren, want PASCAL en GOTO is water en vuur).

-----  
SP

DE BLAST SPECTRUMBASIC COMPILER

SP  
-----

NB: Beta-BASIC is GEEN "nieuwe" taal en OOK GEEN compiler. Het is alleen een UITBREIDING van de bestaande SP-BASIC. Dit voor geval u door de handleiding van Beta BASIC in verwarring raakt, omdat Beta-BASIC programma's ook enigszins kan versnellen.

De derde mogelijkheid combineert het gemak van BASIC met de snelheid van MC. Driemaal is scheepsrecht zullen we maar zeggen. U schrijft gewoon uw programma in de taal die u gewend bent. Zodra het foutvrij is, haalt u het door de compiler. Daarna is uw programma iets langer en vooral veel sneller. Over zo'n compiler-programma wil ik het hier hebben.

#### BLAST

Mijn versie 3.7 kwam, zoals voor SP-programmatuur gebruikelijk, op band. De makers hebben zelfs een weinig geslaagde poging gewaagd om er een klaphoes omheen te bouwen. Helaas is het gebleven bij een slap soort multiband-omslag zonder spiraal, waardoor het pakket niet stapelbaar is en het alleen verticaal kan worden neergezet. In dezelfde verpakking bevindt zich ook het 27 pagina's dikke handboek en een vel met gekleurde puntjes.

#### BAND

Het bandje bevat dus de programmatuur waar het allemaal om gaat. Op kant A staat de compiler en een programma voor het dupliceren van op MD gecompileerde programma's. Kant B bevat een speciale toolkit. Helaas staat elk programma dus maar een keer op de band, dus als er iets niet wil LOADen is dat jammer.

#### DISKDRIVE

De compiler moet volgens opgave kunnen samenwerken met de MD. Nu heb ik geen ZX-Interfacel met MicroDrives, maar wel een snellere en betrouwbaardere Disciple. En dat is maar goed ook. Het inlezen van de hele compiler duurt, mede door een volstrekt nutteloos en weinig fraai laadscherm, (te) lang. Bovendien is het programma beveiligd. Het is daarom tegen de regels maar wel aan te raden, om met een snapshotinterface een kopie op schijf te maken. U leest daartoe de compiler in en beantwoordt de vraag "Backup to M.Drive" met NO. Vervolgens doet u eenmalig de "authorisatiecontrole" en bij het tussenscherm waarop het aantal vrije bytes staat ramt u even op de snapshotknop. Dan is het programma in het vervolg veel sneller gebruiksklaar en die beveiliging kunt u voortaan vergeten. En omdat DD-snapshots niet te herkopieren zijn, worden er geen (extra) illegale kopieën van het programma in omloop gebracht.

#### COMPILER

U stopt het bandje in uw CR en typt LOAD "". Vervolgens kunt u de zaterdagkrant gaan spellen en daarna de auto een poetsbeurt geven. Hopelijk is het LOADen dan klaar. Maar nog steeds is de



-----  
SP

DE BLAST SPECTRUMBASIC COMPILER

SP  
-----

compiler niet klaar voor gebruik. In het pakket zit ook een vel met 40 maal 26 gekleurde spikkeltjes. Het programma vraagt u nu de kleuren in te tikken van vier willekeurige coördinaten. Houd dus een liniaal bij de hand, want een telfoutje is zo gemaakt en na een reset "Redo from start" is niet leuk. Uw BASIC-programma mag in RAM zitten of van CR of MD worden geload. De gecompileerde CODE kan naar dezelfde gegevensdragers worden geschreven. Van RAM naar RAM is niet aan te raden, want dan mag een programma niet groter zijn dan 2,5 Kb. Het programma vertikt het bij mij om DD-programma's in te lezen, dus moet de CR er weer bij komen.

#### MOGELIJKHEDEN

Dat zijn er nogal wat. Een programma kan worden omgezet. Maar een programma kan ook worden onderbroken, de variabelen bekeken, MC kan worden aangeroepen. Het is zelfs mogelijk, heen en weer te springen tussen BASIC en geBLASTe BASIC, en tussen gecompileerde delen en MC. Ook is er een voorziening aanwezig om aanwezige variabelen NIET te laten verdwijnen. Om ervoor te zorgen dat ook zeer grote programma's omgezet kunnen worden, kunt u schakelen tussen "p-code", een soort pseudo-MC die minder snel is en ook minder ruimte opslokt, en zuivere MC. Bij elk programma wordt een RunTime System of RTS meegeSAVEd voor diverse rekenkundige- en stringbewerkingen. SAVEn kan autostartend gebeuren, maar het programma begint dan altijd met de eerste regel. U wordt dus in veel gevallen gedwongen om weer een of meer extra GOTO-opdrachten in uw programma te stoppen. Zo wordt het eindprodukt bijna een spaghetti-kweekvijver. U mag trouwens zoals gewoonlijk met RAMTOP spelen en overall POKEn, maar het handboek waarschuwt u wel om dat liefst alleen boven RAMTOP te doen. Mits het programma in RAM past, kunt u ook proefdraaien. Omdat dan de BASIC bewaard blijft, hoeft u in geval van een fout niet van voren af aan te beginnen. Na een compilatie van CR naar CR is de compiler "opgebruikt".

#### TOOLKIT

Dat is niet zomaar een aardigheidje als bonus voor het feit dat u zo vriendelijk was een exemplaar te kopen. Als u een te vertalen programma op band wilt zetten, bent u verplicht om het Blast SAVE-commando van de toolkit te gebruiken. Anders kan de compiler de aanbieding niet accepteren. De diverse functies zien er veelbelovend uit, alleen leven de toolkit en uw recensent blijkbaar op voet van oorlog met elkaar. Zowel de compiler als de toolkit kennen een uitgang dmv "\* Q" en <ENTER>. De toolkit kan daarna opnieuw worden gestart. De compiler niet.

#### IBLAST

Dat is, afgaande op het handboekje, een compiler die alleen gehele getallen kan verwerken. Dat programma wordt echter alleen geleverd bij "gewone" BLAST-compilers van versie 4.0 of hoger. Ik heb versie 3.7, dus kan ik daar verder niets over zeggen. Wel

-----  
SP

DE BLAST SPECTRUMBASIC COMPILER

SP  
-----

is aannemelijk, dat programma's waarin berekeningen worden gemaakt, meestal niet voor IBLAST geschikt zijn, want de kans dat die berekeningen altijd gehele getallen opleveren is klein. En omdat BEEP ook met niet-gehele getallen werkt, kan dat ook niet. Wat er dan nog over blijft...??

#### HANDBOEK

Dat telt 27 bladzijdes A5, met daarin niet alleen uitleg over hoe het programma werkt en wat het kan, maar ook nog wat basisbegrippen (source code, object code, compiler, interpreter enz.) Een trefwoowdenlijst onbreekt echter. Aannemelijk is, dat het pakket alleen op 48K-versies draait en misschien in de 48-stand van de 128K SP+. In de handleiding wordt echter geen uitsluitel gegeven, misschien omdat het programma en de SP 128 bijna hetzelfde jaar van uitgifte kennen. Daar wil ik dus niet over vallen. De kleurcodekaart is een los velletje. Dat is dus makkelijk weg en dan kunt u uw hele compiler weggooien (of u moet viermaal erg veel mazzel hebben).

#### CONCURRENTIE

Veel verschillende BASIC-compilers zijn er niet. Wat dat betreft heeft BLAST weinig concurrentie te verwachten. Echter, aannemende dat u toch voornamelijk actiespellen zult willen compileren, is SUPERCODE wel een concurrent. Die toolkit barst namelijk van de schermmanipulaties en de geluidseffecten. Met eenvoudige POKE opdrachten zijn daar zaken instelbaar en u kunt dan gewoon vanuit BASIC RAND-USR-opdrachten geven. U kunt uw programma's dan meteen beveiligen ook. Alleen als u uw eigen S-Calc (spreadsheet) wilt maken, is SUPERCODE geen partij.

#### CONCLUSIE

Erg veel mogelijkheden. Bovendien een van de weinige beschikbare BASIC-compilers voor de SP. Ik vind de bediening echter niet gemakkelijk en de beveiliging "terrible". Wat de toolkit betreft, heb en houd ik een voorkeur voor Beta BASIC 3.0+D. Ik begrijp bovendien niet goed, waarom BLAST en mijn DD zo slecht met elkaar overweg kunnen, want de DD kan MD-opdrachten verwerken. Voor mensen die zowel een diskdrive als een CR hebben, is het een aardig programma. Als u alleen met bandjes werkt, kan ik "Blast, THE Spectrum Basic Compiler" toch niet echt aanbevelen.

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087



-----  
SP

## HET SAVEN EN HERLOADEN VAN VARIABLEN (2)

SP  
-----

In IMPULS 82-24 heeft Ed Weijgers een methode van Andrew Pennell om alleen de variabelen van een programma te SAVEN verbeterd \*). In deze routines moet er twee keer geSAVED en dus geLOAD worden. Het is echter mogelijk om slechts eenmaal te SAVEN en te LOADen. De enige restrictie is dan, dat de eerste variabele (na CLEAR of RUN) een geDIMde string is, die niet later opnieuw geDIMd wordt. Deze methode kan in elk programma worden gebruikt door daarin de volgende regels op te nemen. Eerst de (zelfde) functiedefinitie:

```
rn0 DEF FN P(A)=PEEK A+PEEK (A+1)*256
```

Als voorbeeld voor de genoemde variabele gebruiken we hier:

```
rn1 CLEAR : DIM V$(32)
```

De lengte LEN in V\$ op de adressen VARS+1&2 bedraagt nu 35 (& 0) en er geldt dat  $LEN = E\_LINE - VARS - 4$  (zie zonodig IMPULS 82-06). Om V\$ later weer het gehele variabelengeheugen te laten beslaan, kunnen we dan de daarvoor noodzakelijke LEN in SEED krijgen met:

```
rn2 RANDOMIZE FN P(23641)-FN P(23627)-4
```

SAVE dan na het verPOKEN van VARS+1&2 met SEED (herstel nadien):

```
rn3 POKE FN P(23627)+1,PEEK 23670:
 POKE FN P(23627)+2,PEEK 23671:
 SAVE .. "naam" DATA V$():
 POKE FN P(23627)+1,35: POKE FN P(23627)+2,0
```

LOAD gewoon, maar herstel daarna wel de oorspronkelijke V\$ met:

```
rn4 CLEAR : LOAD .. "naam" DATA V$():
 POKE FN P(23627)+1,35: POKE FN P(23627)+2,0
```

Deze truc maakt het ook mogelijk een deel van de variabelen te SAVEN, zonder allerlei niet essentiële hulpvariabelen dus. Zorg er dan voor dat de gewenste variabelen gecreeerd zijn voordat rn2 doorlopen wordt. Dit kan bv met opdrachten als

```
LET a$="Jan": DIM b(17,5): LET t=9: FOR n=1 TO 1
```

Deze variabelen mogen tussen rn2 en rn3 nog wel van waarde, maar niet meer van plaats veranderen. Dus geen DIM b(...) meer, of LET a\$=..., wel LET a\$()=..., LET n=3 enz (zie IMPULS 82-07). Er mogen nu nog wel nieuwe variabelen gecreeerd worden, maar die worden daarna door rn3 niet meer in V\$ meegeSAVED.

Wil G Haggenburg - J v Lennepkade 157' - 1054 ZL A'dam

\*) Nu wordt SEED geSAVED ipv KDATA en geen eigen lengtevariabele L meer gebruikt. Deze methode is het kortste, het gemakkelijkste in het gebruik en tast ook het variabelengeheugen niet aan. rEd.

SP

BASICPROGRAMMA VOOR FILIPPIENPUZZELS

SP

Filippiens zijn leuke puzzels die oa vaak in Elsevier voorkomen. Net als bij kruiswoordpuzzels moeten er in een diagram woorden worden ingevuld volgens omschrijvingen, die nu echter niet zijn genummerd maar geletterd. In het diagram staat zo'n letter voor een aantal genummerde hokjes waarin het woord moet komen (horizontaal). In gelijkgenummerde hokjes moet dezelfde letter staan. Deze letters moeten eveneens overgenomen worden in de doorlopend genummerde hokjes onder in het diagram, waarin dan doorgaans een citaat en de naam van de geciteerde verschijnt. Wanneer de omschrijvingen cryptisch zijn spreekt men van een cryptofilippien.

Het bij elke ingevulde letter afzoeken van het hele diagram naar gelijkgenummerde hokjes is een veel minder leuke bezigheid, die nog kans op fouten geeft ook. En bij een verkeerd ingevuld woord krijg je een extra puzzel om alles weer ongedaan te maken. Dit, vond ik, vraagt om zo'n oerdomme, maar hoogstaccurate computer.

Met het volgende programma krijgt u het diagram met omschrijvingen op uw scherm. Een knipperend hokje kunt u met de pijltoetsen over het gehele diagram bewegen, ook over het onderste gedeelte. Met een lettertoets kunt u een letter invullen, die dan in alle gelijkgenummerde hokjes verschijnt, of met de spatietoets al die hokjes weer leeg maken. Leeg, want nummers zijn nu niet meer van belang en verschijnen daarom niet. Wilt u weten welke hokjes met elkaar overeenkomen dan kunt u bijvoorbeeld even een x invullen.

HET BASICPROGRAMMA "filippien" LINE 5 - EdW - 281292

```

1 LET I$=INKEY$: POKE 22495+R*32+K,71: LET N=N(R,K):
 IF I$<="a" AND I$<>" " OR I$>="z" OR N>=Z THEN
 PAUSE 2: POKE 22495+R*32+K,120: PAUSE 1:
 LET K=K-(I$=CHR$ 8 AND K<>1)+(I$=CHR$ 9 AND K<>B):
 LET R=R+(I$=CHR$ 10 AND R<>H)-(I$=CHR$ 11 AND R<>1):
 GO TO 1
2 FOR F=1 TO F(N):
 PRINT AT R(N,F),K(N,F);I$: NEXT F: GO TO 1

5>CLS #: BRIGHT 1: RESTORE : READ H,B: CLS # zonodig
 DIM N(H,B): DIM T$(121): LET Z=100: vervangen door
 DIM R(99,12): DIM K(99,12): DIM F(99): BRIGHT 0: CLS
 LET T$(Z TO)=" ABCDEFGHIJKLMNOPQRSTU": <- deze string in
 IF PEEK USR "A" THEN GO TO 7 GRAPH invoeren
6 FOR R=USR "A" TO USR "U"+7: na het zwarte
 POKE R,255-PEEK R: NEXT R hokje CHR$ 143
7 FOR R=1 TO H: (GR SYMB SH 8)
8 FOR K=1 TO B: de UDG's A enz
 READ N: LET N(R,K)=N: PRINT T$(N);:
 IF N<Z THEN
 LET F=F(N)+1: LET F(N)=F:
 LET R(N,F)=R-1: LET K(N,F)=K-1
9 NEXT K: PRINT : NEXT R: LET R=1: LET K=2:
 READ O$: PRINT BRIGHT 0;O$: POKE 23658,0: GO TO 1

```



SP

BASICPROGRAMMA VOOR FILIPPIENPUZZELS

SP

## DE DATAREGELS

Hieronder en op de volgende pagina's ziet u de DATA-regels voor een vijftal filippienpuzzels, allemaal afkomstig uit Elsevier.

In regel 10 staat telkens de hoogte en breedte van het diagram.

Dan volgen de regels voor de genummerde hokjes. Voor de zwarte hokjes daartussen dient de letter Z opgenomen te worden en voor de letteraanduidingen A t/m ten hoogste U: Z+1, Z+2, Z+3, enz.

In de laatste regel ziet u een string met de omschrijvingen. Voor de letteraanduidingen hierin moeten UDG's gebruikt worden, dus intoetsen: GRAPH A GRAPH, GRAPH B GRAPH, GRAPH C GRAPH, enz. Komt u schermruimte tekort voor de omschrijvingen, dan kunt u de spaties om die UDG's weglaten, die verschijnen nl toch invers. Ook kunt u die tekst zonder printkomma's achter elkaar zetten en verder kunt u nog de zwarte regel uit een diagram weglaten.

## DE WERKING VAN HET PROGRAMMA

Het programma bestaat uit twee gedeelten: de voorbereiding vanaf startregel 5, waarin onder andere de DATA worden gelezen, en het eigenlijke programma, waarmee de puzzel ingevuld kan worden.

Voor in regel 5 staan CLS # en RESTORE, nodig wanneer u nog een filippien LOADt: ze brengen het scherm en de positie voor DATA-

10 DATA 13,17: REM Hoogte,Breedte: FILIPPIEN 776

```

11 DATA Z+1,3,5,45,11,43,36,31,23,13,6,Z+8,1,59,44,37,29
12 DATA Z+2,19,33,49,4,39,2,42,54,36,Z+9,40,7,36,20,62,27
13 DATA Z+3,47,32,6,64,31,21,10,45,Z+10,22,51,4,65,31,5,44
14 DATA Z+4,56,63,24,60,46,12,34,Z+11,1,25,42,17,11,31,57,49
15 DATA Z+5,38,13,8,28,33,55,Z+12,3,35,23,42,9,61,5,23,67
16 DATA Z+6,63,41,53,15,7,23,26,Z+13,25,13,9,28,58,6,43,36
17 DATA Z+7,8,48,18,6,52,65,Z+14,16,50,41,30,14,66,60,35,49
18 DATA Z,Z
19 DATA 1,2,3,4,5,6,7,Z,8,9,10,11,12,13,14,15,Z
20 DATA 16,17,18,Z,19,20,Z,21,22,23,24,25,26,27,28,29,30
21 DATA 31,32,33,34,Z,35,36,37,38,39,Z,40,41,Z,42,43,44
22 DATA 45,46,Z,47,48,49,50,51,52,53,54,55,Z,56,57,58,Z
23 DATA Z,Z,59,60,61,62,Z,63,64,65,66,67,Z,Z,Z,Z,Z

```

```

24 DATA ",,A provincie- H drukte, alle overbodige
 hoofdstad I dweper, spaties weglaten
 B vis J land in Afrika, , intoetsen als
 C zwerver K wapen, EXT MODE 6 DELETE
 D Europeaan L eiland in de, (zg printkomma)
 E zware storm, Indische oceaan, en A t/m N als
 F betweter M muziekinstrument GRAPH A GRAPH
 G kliek N bende" enz (de UDG's)

```

SP

BASICPROGRAMMA VOOR FILIPPIENPUZZELS

SP

lezen in de initiele toestand. (Het handboek beweert dat CLEAR ook een RESTORE uitvoert, wat echter gelogen is naar mij bleek). Dan wordt met de gelezen diagramHoogte en -Breedte een variabele N geDIMd voor de te lezen Nummers. Voorts wordt T\$ geDIMd, een string met de Tekens die bij die nummers moeten worden afgedrukt (nu nog spaties). Die nummers verwijzen naar de posities in T\$. Dat zijn er maximaal 99, want op Z (100) komt nu een Zwart hokje gevolgd door de UDG's A t/m U voor de aanduidingen van maximaal 21 woorden. Door deze truc konden ALLE hokjes genummerd worden. Voor de 99 'echte' nummers worden ook nog coördinatenparen R(ij) en K(olom) geDIMd, voor ten hoogste 12 gelijkgenummerde hokjes, alsmede de F(requenties: aantallen der paren), alles per nummer, om later snel alle gelijkgenummerde hokjes te kunnen invullen. BRIGHT 1: het diagram komt straks contrasterend op het scherm.

Regel 6 invertteert alle UDG's, dwz verwisselt zwart en wit. Dat is de reden dat voor de woordaanduidingen niet de gewone letters gebruikt zijn. Om nu te voorkomen dat dit weer ongedaan gemaakt wordt als u nog een filippien LOADt voert regel 5 een test uit.

De regels 7 en 8 doorlopen nu het diagram Rij- en Kolomsgewijs. Elk nummer N wordt gelezen en in N(R,K) gezet, waarna het bijbehorende Teken wordt afgedrukt. Bij elk 'echt' nummer, dwz onder de Z, wordt de Frequentie verhoogd met 1 (aanvankelijk 0) en het coördinatenpaar R,K opgeslagen (beide verlaagd met 1, omdat deze gaan dienen voor PRINT-posities, die bij 0,0 ipv 1,1 beginnen).

10 DATA 13,17: REM Hoogte,Breedte: CRYPTOFILIPPIEN 778

11 DATA Z+1,12,1,29,75,37,64,23,47,41,8,22,32,57,28,67,38  
 12 DATA Z+2,17,31,62,72,28,68,46,13,64,23,25,38,Z+8,51,30,58  
 13 DATA Z+3,23,17,52,36,4,4,17,39,49,69,70,Z+9,34,7,55,45  
 14 DATA Z+4,35,9,56,15,21,46,53,15,25,17,Z+10,14,59,44,66,76  
 15 DATA Z+5,24,61,71,63,69,65,2,20,60,Z+11,38,33,6,42,18,50  
 16 DATA Z+6,15,16,38,15,32,55,73,8,Z+12,48,5,27,54,11,77,58  
 17 DATA Z+7,57,17,26,64,23,43,3,Z+13,70,31,4,19,56,74,40,10  
 18 DATA 1,2,3,4,5,6,7,8,9,10,Z,11,12,Z,13,14,15  
 19 DATA 16,17,Z,18,19,20,Z,21,22,23,24,25,Z,26,27,28,29  
 20 DATA 30,31,32,33,34,Z,35,36,37,38,39,40,41,Z,42,43,44  
 21 DATA Z,45,46,47,48,49,50,51,Z,52,53,54,55,56,57,Z,58  
 22 DATA 59,60,61,62,63,64,Z,65,66,67,68,69,Z,Z,Z,Z,Z  
 23 DATA Z,Z,Z,Z,70,71,72,73,74,75,76,77,Z,Z,Z,Z,Z

24 DATA "Azij gaat om met NorenBfantastis  
 che wielrennerCDommelen of Soest  
 Darbeidsbureau?Edier aan boordFd  
 ienstingangGeindberekeningHzo kl  
 inkt echter een hondIhoe je het  
 ook draait, 't blijft negatiefJb  
 ogen voor tienKop de fiets of me  
 t de benenwagenLmijn Franse zaak  
 aan't waterMsterke pompbediende"

achter elkaar  
 doortikkend,  
 zonder spaties  
 rond de toch  
 contrasterende  
 UDG's A t/m M,  
 vullen deze 9  
 regels net het  
 hele scherm op



-----  
SP

BASICPROGRAMMA VOOR FILIPPIENPUZZELS

SP  
-----

Regel 9 leest de Omschrijvingenstring en drukt deze af onder het diagram in normale helderheid. De coördinaten R en K krijgen hun beginpositie in het eerste woord en voor de sprong naar het invulprogramma wordt de SP nog in de kleine-letterstand (L) gezet.

Regel 1 onthoudt een ingedrukte toets in I\$ en invertteert hokje R,K door POKEn in het attributengeheugen op  $22528+(R-1)*32+(K-1)$  van  $64*1+8*0+1*7$  voor: helder aan, papier zwart en inkt wit. Dan wordt het nummer N van hokje R,K gelezen. Als I\$ een kleine letter of spatie is en N een echt nummer, dus niet van een zwart hokje of UDG, dan wordt voortgegaan in de volgende regel. Anders wordt hokje R,K hersteld (wel helder) waardoor een knippereffect ontstaat; de flankerende PAUSEs zorgen ervoor dat dit rustig is. Voordat wordt teruggesprongen, worden de coördinaten R of K met 1 verhoogd of verlaagd wanneer er een pijltoets werd ingedrukt (CHR\$ 8 t/m 11), mits R,K daardoor niet buiten het diagram komt.

Regel 2 drukt voor alle F(N) hokjes met hetzelfde nummer N op de daarbij opgeslagen coördinaten vliegensvlug de kleine letter of spatie uit I\$ af, alvorens terug te springen naar het begin.

PS: Als u in regel 2 na NEXT F : LET T\$(N)=I\$ opneemt, dan wordt een oplossing opgeslagen. Voegt u voorts een nieuwe startregel 4 toe waarheen u de opdracht DIM T\$(121) uit regel 5 verhuist, dan kunt u na BREAK herstarten met GO TO 5, of het programma met T\$ erin SAVEN om de oplossing te kunnen tonen dmv MERGE en GO TO 5.

10 DATA 12,15: REM Hoogte,Breedte: FILIPPIEN 780

11 DATA Z+1,15,32,37,7,2,29,16,12,45,Z+8,38,49,8,6  
 12 DATA Z+2,1,44,6,33,29,22,48,30,Z+9,31,16,13,26,23  
 13 DATA Z+3,41,22,27,3,29,10,19,Z+10,9,32,26,7,47,49  
 14 DATA Z+4,43,32,14,41,28,42,23,Z+11,11,26,28,29,36,5  
 15 DATA Z+5,18,37,40,30,2,34,Z+12,41,12,5,35,21,34,39  
 16 DATA Z+6,46,3,24,42,23,7,Z+13,14,25,20,50,29,8,43  
 17 DATA Z+7,1,16,36,27,4,Z+14,27,19,49,15,28,43,44,17  
 18 DATA Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z  
 19 DATA 1,2,Z,3,4,5,Z,6,7,8,9,10,Z,11,12  
 20 DATA 13,14,Z,15,16,17,18,19,Z,20,21,Z,22,23,24  
 21 DATA 25,26,Z,27,28,29,30,Z,31,32,33,34,Z,35,36  
 22 DATA 37,38,39,Z,40,41,42,43,44,45,46,47,48,49,50

23 DATA ",,A laf, H kleur,  
 B troosteloos, I plaats in,  
 C plaats in, Utrecht,  
 Drente, J Nederlandse,  
 D vogel, dichter,  
 E kledingstuk, K talmen,  
 F laatdunkend, L getal,  
 G Nederlandse, M billijk,  
 schilder, N verloochenen": REM als bij 776

SP

BASICPROGRAMMA VOOR FILIPPIENPUZZELS

SP

10 DATA 12,17: REM Hoogte,Breedte: CRYPTOFILIPPIEN 782

11 DATA Z+1,34,29,9,61,17,22,1,66,44,58,52,4,30,21,42,55

12 DATA Z+2,43,23,7,37,11,48,20,66,14,6,36,41,61,Z+8,50,65

13 DATA Z+3,66,36,24,30,64,51,14,29,36,40,3,Z+9,58,27,39,20

14 DATA Z+4,60,23,57,41,9,16,5,12,25,42,Z+10,27,47,59,11,32

15 DATA Z+5,53,39,32,30,15,42,67,56,28,55,Z+11,3,8,49,46,37

16 DATA Z+6,2,54,38,43,59,19,52,31,14,Z+12,64,36,18,7,23,63

17 DATA Z+7,49,62,30,21,3,10,26,44,Z+13,49,35,13,48,45,62,33

18 DATA Z,1,2,Z,3,4,5,6,Z,7,8,Z,9,10,11,12,Z

19 DATA 13,14,15,16,17,18,19,20,21,Z,22,23,24,Z,25,26,Z

20 DATA 27,28,29,30,31,32,Z,33,34,35,36,37,Z,38,39,40,Z

21 DATA 41,42,43,44,Z,45,46,47,48,Z,49,50,51,52,Z,53,54

22 DATA 55,Z,56,57,58,59,Z,Z,60,61,62,63,Z,64,65,66,67

23 DATA "AtelefoonrekeningBSinterklaas  
 Czonder drukte het cafe inDbe-  
 jaarden maken niet zo'n rotzooi  
 Eoverdreven benoemenFsein om aan  
 tafel te gaanGzeker niet slecht  
 voor een makelaarHkoninklijk  
 waterIgeluid van een zoenJafme-  
 tingen van vriendenKsnedig beet-  
 genomenLstafmedewerkersMis die  
 leider anti-vrouw?": REM 10 schermregels

10 DATA 13,17: Hoogte,Breedte: FILIPPIEN 784

11 DATA Z+1,8,1,38,31,60,10,Z+8,34,49,7,42,13,28,39,17,27

12 DATA Z+2,58,43,6,19,31,24,62,Z+9,62,43,27,20,24,59,13,57

13 DATA Z+3,20,18,34,1,48,60,52,25,Z+10,51,41,13,11,49,55,34

14 DATA Z+4,3,54,12,30,58,17,32,46,2,Z+11,44,54,15,37,10,5

15 DATA Z+5,35,21,50,60,20,9,38,Z+12,5,33,13,61,22,57,53,14

16 DATA Z+6,16,1,26,6,30,61,56,Z+13,14,60,8,30,29,36,10,45

17 DATA Z+7,29,46,38,60,20,4,24,59,13,23,53,Z+14,47,12,53,40

18 DATA Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z,Z

19 DATA 1,2,Z,3,4,5,Z,6,7,8,9,10,Z,11,12,13,14

20 DATA 15,Z,16,17,18,19,Z,20,21,22,23,24,Z,25,26,27,28

21 DATA 29,30,31,Z,32,33,34,35,36,37,38,39,Z,40,41,42,Z

22 DATA 43,44,Z,45,46,Z,47,48,49,50,51,52,53,Z,Z,Z,Z

23 DATA Z,Z,Z,54,55,56,57,Z,58,59,60,61,62,Z,Z,Z,Z

24 DATA ",,A boosaardig, H Nederlandse,  
 B Nederlandse, schilder,  
 schrijver, I hoofddekse,  
 C kerkgebouw, J zweep,  
 D insect, K vermogend,  
 E hachelijk, L meteen,  
 F musicus, M winstuitkering,  
 G beraadslagen, N vreemde munt": REM als bij.776



SP

## ERRATUM TOOLKIT UIT IMPULS 82-18

SP

In regel 12 werd k gebruikt voor 2, de gekozen optiewaarde. Vlak daaronder kreeg K een andere waarde. Omdat dit echter in een lus gebeurt bezit k bij de volgende doorgang de waarde 2 niet meer. U kunt dit herstellen door (A+k) te wijzigen in (A+SQR PI). In regel 12 hieronder is niet alleen deze fout gecorrigeerd, hij is bovendien sneller geworden zonder dat zijn lengte veranderde.

```
12 FOR k=k TO I+I:
```

```
 LET B$=CHR$ PEEK (B-I)+CHR$ PEEK B: INPUT " TO ";R:
```

```
 FOR C=I TO R:
```

```
 LET A=VAL A$: LET C=VAL R$:
```

```
 LET B$=B$+CHR$ PEEK A+CHR$ PEEK (A+I)+
```

```
 CHR$ PEEK (A+k)+CHR$ PEEK (A+PI):
```

```
 POKE A-I, CODE " ": POKE A, CODE " ": POKE A+I, CODE " ":
```

```
 POKE A+k, CODE " ": POKE A+PI, CODE " ": NEXT C:
```

```
 RANDOMIZE VAL A$-B-I: POKE B-I, PEEK L: POKE B, PEEK H:
```

```
STOP : NEXT k
```

Een met optie 2 verkregen regel kan over regels heen getild worden door hem te EDITten en voorts het regelnummer te veranderen. Met optie 4 kunnen dan de oorspronkelijke vervolgregels weer afgesplitst worden. Dit mislukt echter ingeval er door het EDITten FPR's zijn verdwenen uit regels na een regel waarin REM staat.

PS

Elders in dit nummer refereert Rob Willig aan een wedstrijd, indertijd uitgeschreven door de "Sinclair Gids", wie de mooiste zg "one-liners" kon maken. Over wat "mooi" is valt te twisten, maar deze "toolkit" slaat wat lengte betreft met zijn ruim 900 bytes vast, hoewel ik de uitslag niet ken, het winnende programma. Wat het aantal opdrachten in een regel betreft, daaraan zit een maximum. Probeer maar eens een regel te EDITten met meer dan 126 dubbele punten erin. Wanneer u alleen maar dubbele punten in die regel gebruikt - lege opdrachten zijn ook opdrachten - dan zijn die heel gemakkelijk te tellen ( $4 \times 32 - 2$ ). Er zijn dus ten hoogste 127 opdrachten in een regel mogelijk (voor en achter een :). Overigens haalde "toolkit" met 100 opdrachten dit maximum niet.

---

LAATSTE IMPULS/-STUIPTREKKING

---

Sinclair-computers zijn heel geschikt om uit te breiden en om op te programmeren. Ten opzichte van de concurrenten in de beginjaren '80 was er een flink RAM-geheugen, een goede BASIC met opdrachten voor een behoorlijke hoge resolutie en diverse kleuren. Alleen het toetsenbord kon de toets der kritiek nauwelijks doorstaan. Gedurende de hoogtijdagen werden ze met tienduizenden verkocht. Omdat er aan de hardware nog wel wat mankeerde, maar er een edge-connector aanwezig is, ontstond er een enorme toeleveringsindustrie. Wie niet tevreden was met de cassette recorder kon kiezen uit verschillende disk-opslagsystemen. Ontevreden met het toetsenbord? Keuze genoeg. Wilt u in plaats van een ZX-type printer een seriele of een parallelle 80-koloms printer? Kiest u maar. Wilt u echt geluid in plaats van een armetierig piepje? Alstublieft, mogelijkheden zat. Jaap Kuiper heeft aan al die interfaces vele pagina's besteed in RAM (oude stijl, onder Willem Bos). Schrijft u uw programma's liever zelf? Met Beta BASIC, Hisoft BASIC, Pascal of C, Prolog, Logo of MC kunt u alle kanten op. En wie de SP wilde gebruiken als werkpaard kon VU-File/VU-Calc, Tasword, Masterfile, Omnicalc en Komin Grootboek nemen.

## GEBRUIK

Of, en zo ja, waarvoor al die Spectrums werden gebruikt is de vraag, want zelden of nooit schreef iemand een artikeltje in een tijdschrift over een onderwerp als "hoe stop ik mijn grammofooncollectie in Masterfile" of "hoe spaar ik een dure accountant uit door met Omnicalc zelf mijn hypotheek te berekenen". De meeste bladen stonden vol spelletjesprogramma's en een blad als het bovengenoemde RAM zette zelfs een RAM-soft programmaservice op voor wie een hekel had aan eindeloos overtypen. Toch heeft volgens mij Jaap Kuiper in RAM zijn best gedaan, want hij was de enige die ooit tips gaf om commerciële en vaak kopieerbeveiligde software op schijf te zetten. De rest liet het grandioos afweten en zo moest ieder voor zich steeds opnieuw het wiel uitvinden. Toch is het aannemelijk, dat in ieder geval een deel van die vele duizenden kopers ooit weleens een programma heeft geschreven waarmee niet slechts hun eigen probleem was opgelost, maar dat, als het was gepubliceerd, ook voor een ander de oplossing of ten minste een stap in de goede richting had kunnen zijn. Zelden of nooit ben ik een dergelijk artikeltje tegengekomen en dus moest ieder voor zich steeds opnieuw het wiel uitvinden.

## BLADEN

De Sinclair GebruikersGroep is onderdeel van de landelijke Hobby Computer Club HCC. Deze HCC kent een eigen periodiek, clubblad en computervakblad. Vroeger de HCC Nieuwsbrief, nu Computer! Totaal genaamd. In het colofon van die Nieuwsbrief stond dat bijdragen van leden zeer op prijs werden gesteld. Praktische bruikbaarheid en nut stond niet voorop, als het maar creatief was en iets met computers te maken had. De vrijwel enige bijdrage van de Sinclair GG was steeds een uitgebreid verslag van de gebruikersbijeenkomsten in De Bron. Ik heb in mijn eentje meer



-----  
-- LAATSTE IMPULS/-STUIPTREKKING --  
-----

bijdragen over de Spectrum verzorgd voor de HCC Nieuwsbrief dan alle anderen in al die jaren bij elkaar. Ik wil beslist niet beweren dat de kwaliteit van die bijdragen steeds boven iedere kritiek verheven was, maar het was tenminste iets.

De HCC Sinclair GG kent een eigen periodiek, Sinclair Impuls, het blad dat u nu zit te lezen. Het wordt al sinds jaar en dag volgeschreven door een heel (veel te) klein aantal mensen.

Dat geldt ook voor het Bulletin van de Sinclair GG Groningen/ Assen. De snelst gevulde pagina is de redactionele. Die wordt steeds gevuld met een smeekbede aan de lezers/abonnees: stuurt u alstublieft eens wat in, want zonder kopij kunnen we geen blad maken. Maar waarom moeten er in een toch klein land als Nederland dan ook twee clubs met ieder een eigen periodiek en met hetzelfde doel bestaan? Als de redacties hadden samengewerkt, had dat verschillende voordelen gehad:

- De gebruikers hoeven zich slechts bij een club met een blad aan te sluiten om alle informatie te verkrijgen;
- Ipv twee bladen die allebei slechts met veel moeite gevuld kunnen worden verschijnt er regelmatig een tijdschrift;
- Er is slechts een gemeenschappelijke softwarebibliotheek nodig en dus ook geen concurrentiestrijd wie wat krijgt.

Na het uitkomen en populair worden van het Opus Discovery interface, richtten de gebruikers een aparte club op, de Discovery Users Club DUC. Wie niets van het Nederlandse Sinclair-gebeuren wilde missen, moest dus bij drie loketten wezen om zeker te zijn van alle bladen. Pas na verloop van tijd besloten Impuls en DUC te gaan samenwerken door te fuseren ipv elkaar te beconcurreren.

Het verhaal herhaalde zich toen de Disciple-interface -te laat- uitkwam, gevolgd door de Plus-D. De gebroeders Faas begonnen over de ins en outs van dit kastje te publiceren in hun eigen blad, de Disciple Nieuwsbrief. Zijzelf staken er veel tijd in en met hen (te) weinig anderen. Wederom moest iedereen die niets wilde missen zich opsplitsen om alle bladen te bemachtigen. Ook voor de Disciple Nieuwsbrief werd de spoeling tenslotte te dun en uiteindelijk hield het blad op te bestaan.

## TECHNEUTENTIJDSCHRIFT

Er zijn klachten geweest, dat het niveau van Impuls langzaamaan onbegrijpelijk hoog en erg technisch is geworden. Dat schrikt enerzijds wellicht af, anderzijds is het de schuld van de leden/abonnees, die behalve (hopelijk) de contributie nooit enige inhoudelijke bijdrage aan hun Impuls leverden. De weinige mensen die nog wel actief bezig zijn, gaan langzamerhand steeds dieper graven en spitten en halen technisch de onderste steen boven. Bij gebrek aan iets anders worden de resultaten van dat speurwerk omgezet in kopij. Daarmee wordt het niveau van Impuls inderdaad, vanzelfsprekend, vrij hoog. Doch niet de actieve redactie treft hier blaam, maar al die uitsluitend passieve lezers.

-----  
-- LAATSTE IMPULS/-STUIPTREKKING --  
-----

## DISCIPLEZAKEN

Toen de Disciple Nieuwsbrief ophield te verschijnen, bereikten de HCC Sinclair GG klachten van Disciple- en Plus-D-gebruikers dat er niets meer gebeurde. Er verscheen een oproep in Impuls om weer voor artikelen en programma's te zorgen. Ondertussen had na het verdwijnen van de Disciple Nieuwsbrief de Sinclair Gebruikers-Groep groningen/Assen alle abonnees van de Nieuwsbrief een kennismakings- en proefnummer gezonden. Alle oude en nieuwe lid-abonnees werden door twee Disciple-experts voorzien van artikel-tjes en programma's. De mensen die vroeger voor de Disciple Nieuwsbrief regelmatig iets inzonden, lieten echter niets meer van zich horen. Van hen is verder taal noch teken vernomen.

De oproep in Impuls leidde er na de ledenvergadering (wat heet) in 1990 toe, dat ik het initiatief nam om de gebroeders Faas te benaderen. Ik kreeg voor de Sinclair GG de hele voorraad Nieuwsbrieven en alle diskettes los, herordende en catalogiseerde de boel en sinds de HCC-Dagen 1990 zijn de boekjes en schijffjes weer te koop.

Gezien de klachten der gebruikers toen er niets meer gebeurde op het Disciple-front zou men verwachten dat het storm zou lopen als er weer iets op gang zou komen. Na de HCC-dagen van 1990 en een hoopvol begin van 1991 werd het echter snel stiller. De HCC-Dagen 1991 betekenden nog een opleving, maar in 1992 stagneerde de verkoop van Disciple Nieuwsbrieven en DDiSCs totaal. Er is eigenlijk niets nieuws bijgekomen. Geen DDiSCs meer met programmatuur, mondjesmaat een artikel-tje in Impuls. Voor de enkeling die mij nog wel van software voorzag is het een teleurstelling geworden. Omdat een programma-tje van een paar Kb te weinig is om een schijf te vullen, maar er gewoon niet meer binnenkwam, zijn de programma's van deze mensen nooit uitgebracht. Voor deze enthousiastelingen spijt mij dat, ik had het graag anders gezien en gedaan. Maar er is geen vraag en te weinig aanbod. Ook ik ben teleurgesteld, want die klagende Disciple-gebruikers van weleer hebben het naderhand massaal laten afweten. Zo voel ik het tenminste.

De nog overgebleven Sinclairgebruikers gaan een moeilijke tijd tegemoet. De Spectrum wordt al jaren niet meer geproduceerd. Dat geldt ook voor de randapparatuur en toebehoren. De onderdelen worden schaars, de databank bestaat niet meer en de hulplijn is opgeheven. Als u dit leest, "is" ook Impuls niet meer. Een oprecht woord van dank aan en bewondering voor de weinige mensen die zich jarenlang hebben ingespannen en met heel veel inzet die Impulsen hebben volgeschreven, lijkt mij zeer verdiend en op zijn plaats!

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087



-----  
DD

NIEUWS VAN DE DD-PROGRAMMABANK

+D  
-----

Wegens de al vermelde reorganisatie van onze programmabank, maar ook vanwege het verschijnen van DDiSC-21 t/m DDiSC-27 krijgt u:

## EEN OVERZICHT VAN DE THANS VERKRIJGBARE DDiSCS

|           |                                                                                                  |              |
|-----------|--------------------------------------------------------------------------------------------------|--------------|
| M1        | inhoud van 01, 02, 03 en 04                                                                      | zie voor de  |
| M2        | inhoud van 05, 06, 07 en 08                                                                      | inhoud het   |
| 09        | ongewijzigd                                                                                      | overzicht    |
| M3        | inhoud van 10, 11, 12 en 13                                                                      | in           |
| 14 t/m 20 | ongewijzigd                                                                                      | IMPULS 91-18 |
| 21 nieuw  | (datacommunicatie)software uit IMPULS in P-files met uitpakprogramma en uitgebreide documentatie |              |
| 22 met    | pijldiskprogramma's van Megasoft                                                                 |              |
| 23 nieuw  | bijbehorende handleidingen in S-files (TW2)                                                      |              |
| 24 met    | disassembly van de DISCiPLE-ROM                                                                  |              |
| 25 nieuw  | door Otto van der Velden                                                                         |              |
| 26 met    | disassembly van de DISCiPLE-RAM                                                                  |              |
| 27 nieuw  | door Otto van der Velden                                                                         |              |

Prijs f 10 (HCC-SGG-leden en IMPULS-abonnees) of f 12.50 / disk, plus verzendkosten: f 2.50 (1 of 2 disks) of f 4.50 (3 of meer). Bij balieverkoop zijn uiteraard geen verzendkosten verschuldigd.

## DDiSCS BESTELLEN

is slechts mogelijk door overmaking van het juiste bedrag naar

postgirorekening 5374525  
tnv HCC SINCLAIR GG te Bunnik

ovm DDiSC (nodig ter onderscheiding van DUCDISKS)  
met gewenste nummer(s) en gewenst formaat, tw:

3.5 inch 80 tracks  
5.25 inch 80 tracks  
5.25 inch 40 tracks.

Elke DDiSC is enkelzijdig beschreven en bevat hoogstens 180 Kb. Gireer wel het JUISTE BEDRAG en vermeld uw HCC-nummer wanneer u de ledenprijs hanteert, anders ontvangt u minder dan u bestelde.

Rob Willig - J v Stolberglaan 74 - 1412 BJ Naarden - 02159-43087

| IMPULS 93-94 |              |
|--------------|--------------|
| 1            | IMPULS 93-94 |
| 2            | IMPULS 93-94 |
| 3            | IMPULS 93-94 |
| 4            | IMPULS 93-94 |
| 5            | IMPULS 93-94 |
| 6            | IMPULS 93-94 |
| 7            | IMPULS 93-94 |
| 8            | IMPULS 93-94 |
| 9            | IMPULS 93-94 |
| 10           | IMPULS 93-94 |
| 11           | IMPULS 93-94 |
| 12           | IMPULS 93-94 |
| 13           | IMPULS 93-94 |
| 14           | IMPULS 93-94 |
| 15           | IMPULS 93-94 |
| 16           | IMPULS 93-94 |
| 17           | IMPULS 93-94 |
| 18           | IMPULS 93-94 |
| 19           | IMPULS 93-94 |
| 20           | IMPULS 93-94 |
| 21           | IMPULS 93-94 |
| 22           | IMPULS 93-94 |
| 23           | IMPULS 93-94 |
| 24           | IMPULS 93-94 |
| 25           | IMPULS 93-94 |
| 26           | IMPULS 93-94 |
| 27           | IMPULS 93-94 |
| 28           | IMPULS 93-94 |
| 29           | IMPULS 93-94 |
| 30           | IMPULS 93-94 |
| 31           | IMPULS 93-94 |
| 32           | IMPULS 93-94 |
| 33           | IMPULS 93-94 |
| 34           | IMPULS 93-94 |
| 35           | IMPULS 93-94 |
| 36           | IMPULS 93-94 |
| 37           | IMPULS 93-94 |
| 38           | IMPULS 93-94 |
| 39           | IMPULS 93-94 |
| 40           | IMPULS 93-94 |
| 41           | IMPULS 93-94 |
| 42           | IMPULS 93-94 |
| 43           | IMPULS 93-94 |
| 44           | IMPULS 93-94 |
| 45           | IMPULS 93-94 |
| 46           | IMPULS 93-94 |
| 47           | IMPULS 93-94 |
| 48           | IMPULS 93-94 |
| 49           | IMPULS 93-94 |
| 50           | IMPULS 93-94 |
| 51           | IMPULS 93-94 |
| 52           | IMPULS 93-94 |
| 53           | IMPULS 93-94 |
| 54           | IMPULS 93-94 |
| 55           | IMPULS 93-94 |
| 56           | IMPULS 93-94 |
| 57           | IMPULS 93-94 |
| 58           | IMPULS 93-94 |
| 59           | IMPULS 93-94 |
| 60           | IMPULS 93-94 |
| 61           | IMPULS 93-94 |
| 62           | IMPULS 93-94 |
| 63           | IMPULS 93-94 |
| 64           | IMPULS 93-94 |
| 65           | IMPULS 93-94 |
| 66           | IMPULS 93-94 |
| 67           | IMPULS 93-94 |
| 68           | IMPULS 93-94 |
| 69           | IMPULS 93-94 |
| 70           | IMPULS 93-94 |
| 71           | IMPULS 93-94 |
| 72           | IMPULS 93-94 |
| 73           | IMPULS 93-94 |
| 74           | IMPULS 93-94 |
| 75           | IMPULS 93-94 |
| 76           | IMPULS 93-94 |
| 77           | IMPULS 93-94 |
| 78           | IMPULS 93-94 |
| 79           | IMPULS 93-94 |
| 80           | IMPULS 93-94 |
| 81           | IMPULS 93-94 |
| 82           | IMPULS 93-94 |
| 83           | IMPULS 93-94 |
| 84           | IMPULS 93-94 |
| 85           | IMPULS 93-94 |
| 86           | IMPULS 93-94 |
| 87           | IMPULS 93-94 |
| 88           | IMPULS 93-94 |
| 89           | IMPULS 93-94 |
| 90           | IMPULS 93-94 |
| 91           | IMPULS 93-94 |
| 92           | IMPULS 93-94 |
| 93           | IMPULS 93-94 |
| 94           | IMPULS 93-94 |
| 95           | IMPULS 93-94 |
| 96           | IMPULS 93-94 |
| 97           | IMPULS 93-94 |
| 98           | IMPULS 93-94 |
| 99           | IMPULS 93-94 |
| 100          | IMPULS 93-94 |



---

 DE INHOUD VAN DEZE LAATSTE DUBBELE IMPULS 93-94
 

---

|    |                                                        |       |
|----|--------------------------------------------------------|-------|
| 02 | COLOFON                                                | --    |
| 03 | VAN DE REDACTIE                                        | --    |
| 04 | SIR CLIVE SINCLAIR                                     | --    |
| 06 | DE OPBOUW VAN DE CATALOGUS                             | OD    |
| 07 | CLEAR CAT IPV FORMAT                                   | OD    |
| 08 | PRINTEN EN POKEN MET DE MULTIPRINT                     | SP    |
| 10 | HET INLEZEN VAN TEKENINGEN OP DE SPECTRUM              | SP    |
| 12 | MC BEWERKEN MET TASWORD TWEE                           | SP    |
| 13 | HI-SOFT-C OP DE DISCIPEL                               | DD    |
| 16 | ONTHOUD UW PINCODE MET EEN ZELFGEKOZEN WOORD           | SP    |
| 18 | ONE-LINERTJES                                          | SP    |
| 19 | RS232-INTERFACES VOOR DE ZX SPECTRUM                   | SP    |
| 25 | DRIVERS VOOR ZOWEL IN- ALS OUTPUT BIJ RS232-INTERFACES | SP    |
| 34 | LOADEN EN SAVEN VAN ZEER LANGE CODEFILES               | OD    |
| 36 | DE ZX-SPECTRUM-EMULATORS "Z80" EN "SPECTATOR"          | PC/QL |
| 41 | ACCENTEN AFDrukKEN MET TASWORD TWEE                    | SP    |
| 43 | INTERRUPT BIJ DE Z80                                   | --    |
| 49 | P-OMZETTERS - DE KIP EN HET EI                         | SP    |
| 50 | SNEL TEKSTPASSAGES OP EEN DISK ZOEKEN MET BETA-BASIC   | DD    |
| 52 | TOOLKITS VOOR DE SPECTRUM                              | SP    |
| 65 | DE FUNCTIE VAL\$ - TOEPASSING 2                        | SP    |
| 66 | DE ZX-PRINTER ZOEMT WEER                               | SP    |
| 70 | FILE NOT FOUND - REPARATIE                             | MD    |
| 71 | DE BLAST SPECTRUMBASIC-COMPIER                         | SP    |
| 75 | HET SAVEN EN HERLOADEN VAN VARIABELEN (2)              | SP    |
| 76 | BASICPROGRAMMA VOOR FILIPPIENPUZZELS                   | SP    |
| 81 | ERRATUM TOOLKIT UIT IMPULS 82-18                       | SP    |
| 82 | LAATSTE IMPULS/-STUIPTREKKING                          | --    |
| 85 | NIEUWS VAN DE DD-PROGRAMMABANK                         | DD    |
| 87 | DE INHOUD VAN DEZE LAATSTE DUBBELE IMPULS 93-94        | --    |

---

|                     |                   |                 |
|---------------------|-------------------|-----------------|
| -- ALGEMEEN         | SP ZXSPECTRUM     | DD DISCIPLE     |
| 80 ZX80             | MD MICRODRIVE     | QL QUANTUM LEAP |
| 81 ZX81             | OD OPUS DISCOVERY | 88 Z88          |
| CR CASSETTERECORDER | BD BETA DISK      | PC SINCLAIR PC  |

---

## DE SINCLAIRDAGEN IN BUNNIK IN 1994

|              |             |            |
|--------------|-------------|------------|
| ZA 10-16 UUR | 05 FEBRUARI | 09 APRIL   |
| 18 JUNI      | 27 AUGUSTUS | 01 OKTOBER |

---

ONDER VOORBEHOUD - BEKIJK STEEDS DE AGENDA IN "COMPUTER! TOTAAL"



