

# SINCLAIR

# MPULS

hcr  
**SINCLAIR**  
Gebruikers  
Groep

**NUMMER 12**

4e KWARTAL 1986



-----  
COLOFONCOLOFON  
-----

SINCLAIR IMPULS Nummer 12

4e kwartaal 1986 jaargang 4

## REDACTIEADRES:

Wilhelminalaan 42  
2625 KH Delft

## ABONNEMENTEN:

Deze lopen altijd tot het eind van een kalenderjaar.  
De kosten bedragen f 25,-- per jaar, maar tijdens een lopend jaar f 5,-- minder voor ieder dan reeds verschenen nummer.  
Het bedrag overmaken naar:

postgiro 5693776 of  
bankgiro 4540.87.446  
van STICHTING IMPULS  
te 's-Gravenhage  
ovv "donatie" & jaar

## ADVERTENTIES:

f 150,-- per pagina per nummer  
f 500,-- per pagina per jaar  
(bedragen inclusief 6 % btw)

## ADVERTENTIEADRES:

Postbus 76  
2260 AB Leidschendam

## SINCLAIR IMPULSOFT:

Zie de pagina's 03, 37 en 38.

## IMPULSOFTADRES:

Postbus 76  
2260 AB Leidschendam

SINCLAIR IMPULS is het blad voor en door de gebruikers van de Sinclaircomputers:  
ZX 80, ZX 81, ZX Spectrum, QL.

Het wordt eens per kwartaal uitgegeven door de  
HCC SINCLAIR GEBRUIKERS GROEP.

## REDACTIEMEDEWERKERS:

Ed Weijgers, redacteur  
Piet Zwager  
Albert Hoekman  
Rob van Staalduinen  
Jack Raats  
Robert Notenboom

## INLEVERING VAN KOPIJ:

De kopij kunt U opsturen naar het redactieadres, of afgeven aan de balie op onze landelijke gebruikersbijeenkomsten in De Bron te Utrecht (en ook op de HCC Microcomputerdagen).

Vergoedingen voor geplaatste artikelen worden vastgesteld door de redactie (doorgaans een tientje per bladzijde).

## INLEVERING VAN SOFTWARE:

Zelfgeschreven programma's kunt U opsturen naar het Impulsoftadres, of afgeven aan de balie (zie hierboven). Ook voor opgenomen software ontvangt U een vergoeding.

-----  
HCC SINCLAIR GEBRUIKERS GROEP:Postbus 76  
2260 AB Leidschendam

Servicetelefoon 01670-66845

## STICHTING IMPULS:

Postbus 212  
1740 AE SchagenTelefoon 070-998791  
-----



RED

VAN DE REDACTIE, HCC SINCLAIR GG EN IMPULSOFT

SGG

Zoals beloofd is deze IMPULS 12 nog voor de HCC-Microcomputer-dagen verschenen. Alle nummers van dit jaar zijn daar nu te koop en we hopen er veel te verkopen, want de kosten over het afgelopen jaar zijn nog niet allemaal betaald. Mede daarom wil ik een vriendelijk verzoek tot U richten: maak liefst zo spoedig mogelijk Uw donatie over, de gegevens kunt U hiernaast in de colofon vinden. Dan bent U niet alleen verzekerd van de IMPULS, ook als U eens niet naar Utrecht kunt, maar dan kan de penningmeester de drukker betalen als de volgende IMPULS uitkomt. Ik vertrouw erop dat U begrip voor dit verzoek hebt en er gevolg aan zult geven.

Dit is alweer de laatste IMPULS van dit jaar, dus hoewel nog wat vroeg, wil ik U namens onze besturen alvast prettige feestdagen en een goed 1987 toewensen. Die wintermaanden zijn wellicht ook de geschiktste perioden om aan mijn vorige verzoeken te voldoen: eindelijk dat artikeltje voor IMPULS of dat programmaatje voor IMPULSOFT schrijven, afmaken en opsturen. Alstublieft, want door dit barstensvolle nummer - zie de inhoudsopgave achterin - zijn we nu echt door onze kopij heen! Wilt U dus ook in februari een IMPULS krijgen, dan moet er wat komen. Voor en door, weet U wel!

Verder kan ik U hier nog melden dat het goed gaat met de reorganisatie van IMPULSOFT. Achterin vindt U het volledige overzicht van alle verkrijgbare cassettes. Dat zijn er nu 21 en dus 9 meer dan de vorige keer. Robert Notenboom heeft voorwaar niet stilgezeten. Hij mag er dunkt mij best eens een pluimpje voor krijgen. Iemand anders, die dat ook verdient voor zijn jarenlange inspanningen voor de Stichting en de SGG, is Rob van Staalduinen. Voor IMPULSOFT heeft hij nu een cartridge klaar met al de tot nu toe in de IMPULS gepubliceerde QL-programma's, aangevuld met nog enkele andere. Die kost slechts f 15,- of f 17,50 bij bestelling. Ook kunt U een zelfgeFORMATte cartridge opsturen met een betaalcheque van f 7,50. Vanwege die lange inhoudsopgave achterin was daar geen plaats meer om U te vertellen hoe U in het bezit kunt komen van de IMPULSOFTcassettes. Daarom hier de 3 mogelijkheden:

- Overmaking van f 12,50 per cassette, o.v.v. gewenst(e) nummer(s), op postgiro 5693775 van "STICHTING IMPULS BESTELGIRO" te Leiden.
- Via de "HCC-BESTELSERVICE" met een bon uit de HCC-Nieuwsbrief.
- Balieverkoop voor f 10,00 per cassette, tijdens de SGG-bijeenkomsten in De Bron en de HCC-Microcomputerdagen in De Jaarbeurs.

Dan krijgt U binnenkort ook weer een acceptgiro van de HCC in Uw bus en U weet wel wat ik U de vorige keer zei: vul in dat U lid van de HCC SINCLAIR GG wilt zijn, want alleen dan houden we onze onmisbare Landelijke Gebruikersdagen. De volgende is op 29 nov, niet 22 nov zoals in de HCC-Nieuwsbrief stond, want 21 en 22 nov zitten we in De Jaarbeurs. U komt toch ook? En daarna in '87 zal

17 jan	21 mrt	23 mei
27 jun	12 sep	28 nov

van 10-16 u in "De Bron", Vader-Rijndreef 7, Utrecht. Tot ziens.

----- rEd -----

-----  
 SP BASICVERSNELLINGSVOORBEELD - LAATSTE VERSNELLING (?) SP  
 -----

BASICVERSNELLINGSTECHNIKEN IN PRAKTIJK uit IMPULS 11-28 bevatte een slotopmerking over de enige mogelijkheid tot verdere verbetering. Nog voor die publicatie toonde Henk Radius de realiseerbaarheid daarvan aan, door op onze Landelijke Gebruikersdag in Utrecht van 26 april met een programma te komen dat aan slechts 1'10'' genoeg had. Ik heb zijn slimme idee nog wat aangepast en daarmee het laatste programma, 7. in mijn artikel, versneld tot:

B. START: LET n=187: LET m=221: LET c=49: GO TO 1 TIJD: 40''

```

0 IF n$(3)=(STR$ m)(LEN n$-2) THEN FOR p=4 TO LEN n$:
  IF n$(p)=(STR$ m)(LEN n$+1-p) THEN NEXT p:
  IF LEN STR$ m<p AND CODE n$=c THEN PRINT n
1 LET n=n+110: LET n$=STR$ n: LET m=m+130:
  IF CODE n$<>c THEN LET c=CODE n$:
    LET n=n-33: LET n$=STR$ n: LET m=m-39:
  IF c=49 THEN LET n=n-33: LET n$=STR$ n: LET m=m-39
2 GO TO n$(2)<>(STR$ m)(LEN n$-1)

```

Een groot aantal 11-vouden valt al direct af, als we slechts die kunnen nemen welke beginnen met de eindcijfers van de 13-vouden, want ook de oplossing voldoet hieraan en wordt zo zeker bereikt.

Dit is mogelijk omdat die rij eindcijfers bestaat uit herhaling van het 10-tal: 0, 3, 6, 9, 2, 5, 8, 1, 4, 7 (alle cijfers).

Dan moeten n en m de 11- en 13-vouden doorlopen met stappen van:

```

10 tegelijk, zonedig gevolgd door
3 terug wanneer het begincijfer van n verandert, en nog
3 terug als dit 1 wordt (0 wordt immers overgeslagen).

```

Hiervoor wordt van het begincijfer van n de CODE in c onthouden.

Eigenlijk zou na elke stap terug getest moeten worden of het begincijfer daardoor niet verandert, maar dat kan net zoals bij de lengtetest beter achteraf, bij een oplossingscandidaat gebeuren.

We kunnen nu niet starten met n=m=0 omdat de gevonden stappenberekening nog niet werkt bij getallen onder de 3 cijfers: het begincijfer van n verandert dan nog te sterk. Dus moet CODE "1" de startwaarde voor c zijn, waarbij de eerstmogelijke startwaarden voor n en m volgens de 10 eindcijfers hierboven 17-vouden zijn.

De besparing op het aantal te doorlopen veelvouden is bijna 90%, hetgeen ruimschoots opweegt tegen de toeneming der opdrachten in de vaakst doorlopen lus, want de tijdwinst bedraagt ruim 5', wat toch ook neerkomt op bijna 89% ten opzichte van ons programma 7.

Nu is het programma zo snel dat het slechts een nacht kostte om het door te laten zoeken naar een volgende oplossing. Die blijkt er tot 1e8 niet te zijn. Na 84615377 functioneert dit programma niet meer omdat de 13-vouden dan meer dan 8 cijfers gaan tellen, waardoor in STR\$ m wetenschappelijke notatie gaat verschijnen.



---

 SP? ERVARINGEN VAN EEN QL-BEGINNER: SNELHEID & SUPERBASIC QL
 

---

Toen ik na mijn vorige stukje "QL?" er een had aangeschaft (QG?) was de snelheid het eerste wat ik wel eens wou bekijken. Dat lag nogal voor de hand na al mijn SP-artikeltjes over dat onderwerp. Het laatste (?) basicversnellingsvoorbeeld - zie in dit nummer - zag er toen, zo weinig mogelijk veranderd, voor de QL aldus uit:

8. START: n#=187: m#=221: c=49: RUN TIJD: 24''

```

1 n#=n#+110: m#=m#+130:
  IF CODE(n#)<>c: c=CODE(n#): n#=n#-33: m#=m#-39:
    IF c=49: n#=n#-33: m#=m#-39
2 IF n#(2)<>m#(LEN(n#)-1): RUN
3 IF n#(3)<>m#(LEN(n#)-2): RUN
4 FOR p=4 TO LEN(n#):
  IF n#(p)<>m#(LEN(n#)+1-p): RUN
5 IF LEN(m#)=p AND CODE(n#)=c: PRINT n#: ELSE RUN
  
```

U ziet het, hierbij is de QL bijna tweemaal zo snel, en dat gaat ook op bij de voorgaande minder snelle versies van dit programma waarmee ik U niet lastig wil vallen. Ik had een grotere snelheid verwacht, maar dit is natuurlijk geen uitputtende snelheidstest. Wel blijkt hetgeen ik over snelheidsverbetering voor de Spectrum heb geschreven in grote lijnen ook voor de QL te gelden.

Voor QL-ers nu zeggen dat dit toch geen SuperBASICprogramma was, wil ik voor SP-ers eerst de verschillen tot dusver toelichten.

Die truc met rnr 0 wist ik niet toe te passen. Daardoor kwam die regel 0 hier achteraan terug, gesplitst in de regels 3, 4 en 5, na een eveneens licht gewijzigde regel 2. Wel gebruikte ik hier RUN, wat bij de QL hetzelfde is als GO TO 1, en dus geen variabelen wist zoals bij de SP.

LET en THEN zijn FACULTATIEF, als er nog wat volgt moet THEN wel door : vervangen worden. Na : is de gebruikte spatie niet nodig.

HAAKJES zijn VERPLICHT bij FUNCTOREN (CODE, LEN enz).

Dan dat vreemde n#=187. Dit QL-mechanisme heet COERCITIE, omdat de QL dan gedwongen is van 187 eerst "187" te maken, voordat aan de stringvariabele n# een waarde gegeven kan worden. Pas op voor de consequentie: n#+m#=408 als ook nog m#=221 uitgevoerd werd! Daarom wordt & voor CONCATENATIE gebruikt: n# & m# = "187221".

En dan nu nogmaals hetzelfde programma, maar in echt SuperBASIC:

```

100 LET n#=187: m#=221: c=49
110 REPEAT zoek
120   LET n#=n#+110: m#=m#+130
130   IF CODE(n#)<>c THEN
140     LET c=CODE(n#): n#=n#-33: m#=m#-39
150     IF c=49 THEN LET n#=n#-33: m#=m#-39
160   END IF
170   IF n#(2)<>m#(LEN(n#)-1) THEN NEXT zoek
180   IF n#(3)<>m#(LEN(n#)-2) THEN NEXT zoek
190   FOR p=4 TO LEN(n#)
200     IF n#(p)<>m#(LEN(n#)+1-p) THEN NEXT zoek
210   END FOR p
220   IF LEN(m#)=p AND CODE(n#)=c THEN EXIT
230 END REPEAT zoek
240 PRINT n#
  
```



Hoewel langer is het niet trager geworden! Wel doorzichtiger met die door inspringen benadrukte REPEAT-, FOR- en IF-blokken. Lange namen, die precies aangeven waarvoor de variabelen dienen, en REM-toelichtingen kunnen de leesbaarheid nog verder verhogen. Dat heb ik hier niet gedaan, omdat de namen en de werking al uit de vorige programma's bekend zijn.

Denk nu echter niet dat dit kwistiger gebruik van geheugenruimte nooit consequenties voor de executiesnelheid zou kunnen hebben. Wanneer dit programma achter een lang ander programma geplaatst wordt, dan gaat de executie ervan meer tijd kosten, waarschijnlijk omdat allerlei opzoekprocessen in dat geval trager werken. Wie wil er in IMPULS eens beschrijven hoe die precies verlopen?

Zoals U hebt kunnen zien behoeft er op FOR en IF niet altijd een afsluitend END te volgen. Er bestaan korte, eenregelige vormen. KORTE IF's vindt U in de regels 1 t/m 5, 150, 170, 180, 200 en 220, en een KORTE FOR, waarmee U wel moet uitkijken, in regel 4. Die moest namelijk lang gemaakt worden (190 t/m 210), toen bleek dat zo'n korte vorm binnen een REPEAT-lus niet goed werkte, hoewel de handleiding daar niet erg duidelijk over is.

Over de waarde van de LOPENDE VARIABLE na het verlaten van een FOR-lus zegt die handleiding helemaal niets. Anders dan bij de SP wordt de grenswaarde NOOIT overschreden en blijft de laatste waarde binnen de lus behouden.

NEXT wordt alleen gebruikt om in FOR- en REPEAT-lussen, nog voor END bereikt is, toch alweer op herhaling te gaan. NEXT dient dus niet, zoals bij de SP, tot afsluiting van een FOR-lus, alhoewel dat soms wel mogelijk is. EXIT is voor "tussentijdse verlating".

Het laatste programma is geheel regelnummernafhankelijk. Daarom heb ik AUTO (zonder meer) gebruikt waardoor het begint met regel 100 en opklimt met stappen van 10. Andere keuzen zijn mogelijk, ook via EDIT waaraan, behalve een rnr, ook een rnr-stap kan worden toegevoegd voor automatische regelnummering. Net als bij de SP verschijnt de EDIT-regel onder in beeld, echter met de cursor achteraan, en met ENTER komt hij in het programma. Dit laatste gaat ook met de verticale-pijltoetsen, waarbij dan de vorige of volgende regel geEDIT wordt. Ik mis de SP-programmacursor, maar vond wel een truc om een bestaande regel makkelijker te EDITten: Wis een niet-bestaande regel ervoor met rnr en pijl-neer-toets. LIST vind ik erg vervelend vanwege het doorrollen tot het einde. Voorts kent de QL nog een RENUMBER- en een DLINE-opdracht om een regel, of een heel regelblok, te henummeren of te verwijderen.

Wilt U liever met een "full screen editor" werken, dan kunt U de "QLToolkit II" van Tony Tebby op cartridge of EPROM aanschaffen. U krijgt dan tevens een groot aantal nieuwe, dan wel gewijzigde FUNCTIES en PROCEDURES tot U beschikking, waaronder de mogelijkheid met elke toets, in combinatie met de ALTertoets, een zelfgekozen serie "direct commands" uit te laten voeren. Het intoetsen wordt beperkt door een aantal zelf te veranderen "defaults", verstekwaarden, zoals mdv2\_ dat U bij SAVE en LOAD mag weglaten.

Als SP'er mis ik toch die eentoetsbediening wel een beetje, hoewel daar bij de SP al nadelen aan zaten. Ik had verwacht dat er meer verkorte KEYWORDS zouden zijn dan die paar waarvan slechts de hoofdletters ingetikt behoeven te worden, waarna de rest vanzelf in het programma verschijnt. Was er maar een KEYWORD-toets om met een, twee of drie letters elk KEYWORD te kunnen EDITten!



Wat dit BASIC pas echt Super maakt is een uitbreiding die ons in staat stelt om onze EIGEN OPDRACHT- EN FUNCTIENAMEN te gebruiken door middel van definitie van PROCEDURES en FUNCTIEPROCEDURES. Hiermee kunt U aan Uw programma's een modulaire structuur geven: WAT een programma doet wordt gesplitst in deeltaken, verwoord in eigen opdrachten; HOE die opdrachten worden uitgevoerd staat dan in de bijbehorende procedures, zonodig opnieuw uitgedrukt in eigen opdrachten, tot we tenslotte uitkomen bij de elementaire opdrachten, die in SuperBASIC. Dit heet stapsgewijze verfijning.

Een voorbeeld van zo'n procedure vormt deze programmatoevoeging:

```
250 DEFine PROCedure Stapje_terug
260   LET n%=n%-33: m%=m%-39
270 END DEFine Stapje_terug
```

Hierdoor kunnen we in de regels 140 en 150 de twee waardetoekenningen uit regel 260 vervangen door: Stapje\_terug. Door deze opdracht, AANROEP genoemd, wordt namelijk het blok regels (slechts een hier) tussen DEF en END uitgevoerd. Hoewel dit wat op GO SUB lijkt, is zo'n regelnummerafhankelijke subroutine nu overbodig. Deze procedure leek me eerst niet erg zinvol voor dit programma, tot bleek dat daarmee nog een seconde tijdwinst behaald werd.

Als voorbeeld van een functieprocedure volgt nog een toevoeging:

```
280 DEFine FuNction Cijfer_anders (positie)
290   RETURN n%(positie)<>m%(LEN(n)+1-positie)
300 END DEFine FuNction Cijfer_anders
```

De voorwaarden in de regels 170, 180 en 200 vervangen we daarmee door: Cijfer\_anders (2), Cijfer\_anders (3) en Cijfer\_anders (p). Ook hier werkt de naam als aanroep, maar nu is dat een functor, dus een expressie of een deel daarvan (zonder het FN van de SP).

Procedureblokken, ook bij functieprocedures, kunnen uit meerdere regels bestaan. RETURN dient voor "tussentijdse terugkeer": bij FN moet er een expressie voor de functiewaardeuitvoer op volgen, wat dus eenmaal zonder (echte) voorwaarde zal moeten voorkomen.

Een naam achter DEFine PROCedure of FuNction mag gevolgd worden door een lijst FORMELE PARAMETERS, een door haakjes omsloten rij namen, die worden gescheiden door komma's of andere separatoren. Die namen zijn van een soort variabelen, die alleen maar binnen die procedure kunnen worden gebruikt en die daarom LOKAAL heten. Wanneer er ook nog andere variabelen in voorkomen, dan heten die GLOBAAL, omdat die dan overal in het programma betekenis hebben. Globale variabelen kunnen lokaal gemaakt worden door aan het begin van de procedure een opdracht op te nemen, bestaande uit het woord LOCAL, gevolgd door een lijst namen van al die variabelen, waarbij waardeverandering tot de procedure beperkt moet blijven.

Een naam gebruikt als AANROEP moet gevolgd worden door een lijst ACTUELE PARAMETERS indien de definitie formele parameters bevat. Deze lijst moet evenveel actuele parameters tellen als er formele zijn en hoeft alleen bij een functie tussen haakjes te staan. Voor de INVOER van waarden waarmee de procedure moet werken kunt U voor de actuele parameters constanten of expressies invullen. Aan het begin van de procedure krijgt elke formele parameter dan de actuele waarde van de constante of geëvalueerde expressie. Wanneer U als actuele parameter een variabele gebruikt, vindt er



tijdens de executie van de procedure een doorverwijzing naar die variabele plaats. Daardoor zal die dus van waarde veranderen als de formele parameter dat doet, zodat zo'n parameter dan ook voor UITVOER van waarden kan worden gebruikt. Zo'n waardeverandering van een voor de invoer gebruikte variabele kan voorkomen worden door er bij de aanroep haakjes omheen te plaatsen, waardoor U er een expressie van maakt die alleen maar geevalueerd wordt. Zou positie in de voorbeeldprocedure van waarde veranderen, dan zou ook p veranderen, tenzij de aanroep Cijfer\_anders ((p)) was.

Dit artikel had tot doel een belangrijk deel van SuperBASIC, dat verschilt van SpectrumBASIC, aan QL-beginners voor te stellen. De meegeleverde "QL User Guide" vermeldt expliciet twee functies die in SuperBASIC overbodig zouden zijn, namelijk VAL en STR\$. We hebben gezien dat dit, dankzij coërcitie, voor STR\$ inderdaad geldt, maar hetzelfde is bij VAL beslist niet het geval! VAL evalueert namelijk bij de SP een hele numerieke expressie in een stringexpressie, terwijl coërcitie bij de toekenning van een stringwaarde aan een numerieke variabele slechts lukt als de betreffende string begint met een constante (getal). Zo levert a=a\$: PRINT a bijvoorbeeld 1 op ingeval a\$="1/2" en zelfs een foutmelding ingeval a\$="PI" of a\$="b" met b=8. Bij toekenning met INPUT en READ doet zich hetzelfde voor. Zo accepteert INPUT a alleen maar getallen en geen expressies. Dit valt met INPUT a\$ niet te omzeilen, want juist dan ontberen we VAL om een numerieke expressie in a\$ te kunnen evalueren. Volgens mij is de weglating van de VAL-functie wegens de toegevoegde coërcitie een misrekening van de ontwerpers geweest. Zelf een VAL-functie definiëren in SuperBASIC lokt mij geenszins aan: niet vanwege de ingewikkeldheid van zo'n ongetwijfeld lange procedure, maar wegens de wetenschap dat er in de ROM reeds een expressie-evaluator aanwezig is. Het moet dus mogelijk zijn een betrekkelijk eenvoudige VAL-functieprocedure in MC te schrijven die daar gebruik van maakt. Wie kan en wil dat eens proberen?

Dat ik ook enkele punten van kritiek heb in dit artikel, wil beslist niet zeggen dat ik ontevreden ben over mijn QL, verre van dat! Dus begrijp me goed als ik hier nog twee aanmerkingen maak.

Ten aanzien van die door mij geroemde mogelijkheid om eindelijk echt gestructureerd te kunnen programmeren zonder gebruik van de regelnummers heb ik de ontwerpers op één inconsequentie betrapt. Het mechanisme bij READ-DATA-RESTORE is hetzelfde als bij de SP. Dat houdt dus in dat we toch nog "RESTORE rnr" moeten gebruiken! Waarom er geen blok mogelijk is, dat begint met "DATA naam" en eindigt met "END DATA naam" met regels ertussen die dan alleen maar data behoeven te bevatten zonder "DATA", is me een raadsel! Dan hadden we "RESTORE naam" kunnen gebruiken, zonder dat rnr!

Tot slot nog iets over de handleiding, een op het eerste gezicht fraaie band. Dat grote A4-formaat, waarbij de bladen steeds over vier beugels getild moeten worden, vind ik onhandig naast de QL. Erger is dat er, na aftrek van zowat de helft over de software, vijf hoofdstukken resten, waarover bijeenbehorende onderwerpen verspreid zijn zonder een index van die vijf. In die praktische boekjes voor de ZX-en werden de dingen completer beschreven ook. Gelukkig is er een prima alternatief te koop, zie elders in deze IMPULS: "QL SuperBASIC - The Definitive Handbook" van Jan Jones.



SP

DE WAFADRIE VAN ROTRONICS

SP

Voor de ZX-SPECTRUM zijn enige speciaal op maat gemaakte opslagssystemen te koop. Het bekendst is de MICRODRIVE met INTERFACE 1: een vrij grote kast die onder de SPECTRUM gemonteerd moet worden met daarnaast 1 of meerdere drives (tot een maximum van 8) al of niet vast aan elkaar gekoppeld. De tapecartridges zijn wat klein maar het door SINCLAIR ontwikkeld systeem is behoorlijk betrouwbaar en alle voor de SPECTRUM leverbare programma's zijn op cartridge te koop. Er is alleen een 'RS 232'-interface aanwezig.

Wat minder bekend maar door de telkens dalende prijs steeds meer verkocht is de 'OPUS DISCOVERY': een opslagsysteem met diskettes van het formaat 3.5 inch. Niet bijzonder snel maar wel zeer betrouwbaar en gemakkelijk uit te breiden met een extra drive. De meest gangbare interfaces (CENTRONICS en RS 232) zijn ingebouwd.

Waarschijnlijk het minst bekend is de 'WAFADRIE' van ROTRONICS: een systeem met eindloze bandjes, 'WAFERS', zoals de MICRODRIVE. Alleen zijn deze WAFERS ongeveer tweemaal zo groot, waardoor ze wat handelbaarder zijn. Door de wat lagere omloopsnelheid is het systeem op zich betrouwbaarder dan de MICRODRIVE en dus ook iets trager maar dit verschil is in de praktijk nauwelijks merkbaar. Volgens de handleiding is de bandsnelheid tijdens "LOAD en SAVE" 10 inch per seconde en tijdens het "zoeken" 15 inch per seconde. De overdrachtsnelheid is ongeveer 2 kbytes per seconde en de gemiddelde toegangstijd is 15 sec. De WAFERS zijn in verschillende lengten te koop, van 16 tot 128 kb. Een geformatteerde WAFER van 128 kb blijkt in de praktijk een opslagcapaciteit te hebben van gemiddeld 140 kb. Een nadeel is dat een WAFER (in de handleiding ook STRINGY FLOPPY genoemd) ongeveer tweemaal zo duur is als een cartridge voor een MICRODRIVE, namelijk zo'n 18 gulden.

De gehele WAFADRIE bestaat uit een kastje van 23x11x8 cm. Aan de voorkant zijn twee openingen voor de drives en ook zichtbaar zijn 3 LED's waarvan de middelste brandt wanneer de WAFADRIE aan de SPECTRUM is gekoppeld en deze is ingeschakeld. De linker en de rechter LED zijn voor de linker en rechter drive en branden wanneer deze draaien. Aan de achterzijde bevinden zich drie printkaartconnectors: een 'RS232'-, een 'Centronics'-aansluiting en een SPECTRUM-uitbreidingsconnector. Jammer is dat geen originele interfaceconnectors gebruikt zijn, maar ik heb dit opgelost door een 9- en een 15-polige D-connector in de achterwand van het kastje te maken: ruimte genoeg!

Ik ben na een jaar intensief (bijna dagelijks) gebruik zeer tevreden over de combinatie van de WAFADRIE en een 'EPSON MX 80'-printer. Door de ingebruikneming van de 'ROTRONICS TOOLKIT' onlangs, zijn mijn mogelijkheden nog beduidend uitgebreid.

ZIJN ER LEZERS DIE WAT ERVARING MET DE WAFADRIE HEBBEN OPGEDAAN EN MET HUN KENNIS ANDEREN WILLEN HELPEN? LATEN ZIJ ZICH MELDEN!  
ZIJN ER WAFADRIEBEZITTERS DIE ER JUIST NIET ZOVEEL ERVARING MEE HEBBEN MAAR HUN KENNIS WILLEN VERGROTEN? LATEN ZIJ ZICH MELDEN!

Mijn opzet is het de WAFADRIE wat meer bekendheid te geven, in onze SINCLAIR IMPULS en ook op onze GEBRUIKERSDAGEN in Utrecht. Ik wacht op UW schriftelijke of telefonische reacties. Albert.



SP

DUBBELGROTE SCREENDUMP MET SEIKOSHA-100 VIA IF1

SP

Na het aanschaffen van IF1 en een 'SEIKOSHA GP-100'-printer zo'n twee en een half jaar geleden ben ik tijden op zoek geweest naar een voor deze printer geschikt 'screendumpprogramma'. Het enige wat werkte was het bekende basicprogrammaatje van Pennell. Het probleem is dat de printer zeer beperkt en slechts 7-pins is waardoor de vele programma's voor de EPSON-achtigen niet werken. Daarom schreef ik het onderstaande mc-programma. Het past precies in de printerbuffer en maakt een copy van het scherm in dubbele grootte, overdwars en beginnend bij de rechter zijkant van het scherm. [Het resultaat ziet U hiernaast. rEd.] De IF1-initialisatie is opgenomen evenals het terugzetten van de printer naar de textmode.

Dus LOAD de code, LOAD het plaatje en geef RANDOMIZE USR 23296.

```
100 LET chs=0: FOR z=23296 TO 23551:
    READ p: POKE z,p: LET chs=chs+p: NEXT z
110 IF chs<>31488 THEN PRINT " CHECKSUM ERROR "
120 DATA 33, 8, 91, 34, 237, 92, 207, 50
130 DATA 225, 225, 33, 26, 0, 34, 195, 92
140 DATA 62, 66, 50, 217, 92, 205, 23, 11
150 DATA 42, 79, 92, 43, 235, 167, 237, 82
160 DATA 235, 33, 28, 92, 115, 35, 114, 205
170 DATA 0, 7: REM Einde initialisatie
180 DATA 62, 3, 205, 1, 22, 62, 8, 215
190 DATA 1, 255, 176, 253, 113, 71, 205, 236
200 DATA 91, 205, 249, 91, 239, 160, 56, 253
210 DATA 78, 71, 197, 5, 205, 206, 34, 62
220 DATA 3, 205, 221, 91, 193, 13, 197, 5
230 DATA 205, 206, 34, 62, 12, 205, 221, 91
240 DATA 193, 13, 197, 5, 205, 206, 34, 62
250 DATA 48, 205, 221, 91, 193, 13, 197, 5
260 DATA 205, 206, 34, 62, 64, 205, 221, 91
270 DATA 205, 213, 45, 198, 128, 215, 193, 16
280 DATA 192, 205, 229, 91, 253, 113, 71, 205
290 DATA 236, 91, 6, 176, 205, 249, 91, 253
300 DATA 78, 71, 197, 5, 205, 206, 34, 175
310 DATA 253, 190, 71, 40, 36, 193, 13, 197
320 DATA 5, 205, 206, 34, 62, 6, 205, 221
330 DATA 91, 193, 13, 197, 5, 205, 206, 34
340 DATA 62, 24, 205, 221, 91, 193, 13, 197
350 DATA 5, 205, 206, 34, 62, 96, 205, 221
360 DATA 91, 205, 213, 45, 198, 128, 215, 193
370 DATA 16, 194, 205, 229, 91, 175, 253, 190
380 DATA 71, 40, 9, 13, 253, 113, 71, 6
390 DATA 176, 195, 56, 91, 62, 15, 215, 205
400 DATA 229, 91, 201, 205, 40, 45, 239, 4
410 DATA 15, 56, 201, 62, 13, 215, 62, 10
420 DATA 215, 201, 62, 27, 215, 62, 16, 215
430 DATA 62, 0, 215, 62, 65, 215, 201, 62
440 DATA 28, 215, 62, 2, 215, 201
```

Bezitters van 'IF1 issue #1' moeten in de derde DATA-regel (140) '23' door '19' en in regel 110 '31488' door '31484' vervangen.



-----  
HCC

DE SERVICETELEFOON VAN DE HCC SINCLAIR GG

SGG  
-----

Velen van U hebben reeds kennis gemaakt met deze SERVICETELEFOON van onze HCC-SINCLAIRGEBRUIKERSGROEP. U weet wel, dat nummer in Steenberghe, in de colofon van deze en ook van de vorige IMPULS.

U kunt bellen op maandag en donderdag van 20.00 tot 22.00 uur, met allerlei vragen over en problemen met Uw Sinclaircomputers. Veel kan direct worden opgelost, en als Jack Raats er niet uitkomt verwijst hij naar iemand die meer van dat onderwerp afweet.

Buiten deze "telefoonspreekuren" krijgt U een antwoordapparaat, waarop U een boodschap kunt inspreken, een verzoek om toezending van informatie over de Sinclair GG bijvoorbeeld, een dankwoord, of opbouwende kritiek. Wat U ook vraagt, U wordt (bijna) nooit teruggebeld, daar is geen beginnen aan.

Voor uitgebreide vragen kunt U trouwens beter schrijven naar ons adres, en let daarbij op het nieuwe postbusnummer:

HCC SINCLAIR GEBRUIKERSGROEP  
Postbus 76  
2260 AB Leidschendam  
Servicetelefoon: 01670-66845





# computercollectief

Amstel 312 (t.o. Carré) / 1017 AP Amsterdam / Giro 4 475 158 / Bank NMB 69.79.15.646

HIEONDER EEN OVERZICHT VAN ACTUELE EN NIEUW  
BINNENGEKOMEN BOEKEN EN SOFTWARE VOOR SINCLAIR COMPUTERS

in BELGIE: IST COMPUTERWINKELTJE pvba, M Sabbestraat 39  
B-2800 MECHELEN, tel 015-206645  
(E10 afrit 9 Mechelen Nrd, 2e Lichten Links)

## - nederlandse ZX Spectrum boeken :

*WERKBOEK MACHINETAAL VOOR DE ZX SPECTRUM .....	f 37,50
inclusief cassette met assembler.	
*VAN BASIC NAAR MACHINETAAL OP DE ZX SPECTRUM ..	f 37,50
ZX SPECTRUM PLUS, tips, programma's, BASIC ....	f 33,75
ZX SPECTRUM HARDWARE BOEK .....	f 25,75
LEREN PROGRAMMEREN ZX SPECTRUM 3 (met foto's) ..	f 27,50
LEREN PROGRAMMEREN ZX SPECTRUM 4 (met foto's) ..	f 27,50
MACHINECODE MET DE ZX SPECTRUM .....	f 30,--
*40 GRAFISCHE PROGRAMMA'S VOOR ZX SPECTRUM .....	f 29,50
FINANCELE PROGRAMMA'S VOOR DE ZX SPECTRUM ....	f 25,--
TOOLKITS en ENKELE SPELEN VOOR DE ZX SPECTRUM ..	f 25,--
*WERKEN MET DE ZX MICRODRIVE .....	f 32,50
ZX SPECTRUM MACHINETAAL ROUTINES .....	f 32,75
PROGRAMMATUUR 1 VOOR DE ZX SPECTRUM .....	f 22,50

## - engelse ZX Spectrum boeken :

*COMPLETE SPECTRUM .....	f 49
SPECTRUM INTERFACING AND PROJECTS .....	f 35
ASSEMBLY LANGUAGE FOR ARCADE GAMES .....	f 42
PRACTICAL ROBOTICS AND INTERFACING FOR ZX SPECTRUM	f 32
MAKING THE MOST OF YOUR ZX MICRODRIVE .....	f 27
ADVANCED GRAPHICS FOR THE ZX SPECTRUM .....	f 52
ADVANCED SPECTRUM MACHINE LANGUAGE (Webb) .....	f 36
ARTIFICIAL INTELLIGENCE ON THE SPECTRUM .....	f 36
MACHINECODE APPLICATIONS FOR THE ZX SPECTRUM .....	f 36
THE COMPLETE SPECTRUM ROM DISASSEMBLY (Logan) .....	f 49
*THE ASTRONOMER'S SOFTWARE HANDBOOK .....	f 45
ZX SPECTRUM ASTRONOMY .....	f 36
INSIDE YOUR SPECTRUM .....	f 36
*SINCLAIR REFERENCE DIARY - 1986 zakagenda .....	f 16
*HANDBOEK COMPUTERKRAKEN EN BEVEILIGEN .....	f 32,50

\*\*\*\*\*  
\* Onze nieuwe WINTER 1986 PRIJSLIJST is nu uit! \*  
\* Stuur ons een kaartje met je naam en adres en de \*  
\* vermelding 'Sinclair Impuls' en we sturen hem \*  
\* gratis aan je toe. \*  
\* COMPUTERCOLLECTIEF - Amstel 312 - 1017 AP AMSTERDAM \*  
\*\*\*\*\*

## - engelse Spectrum boeken :

THE SPECTRUM OPERATING SYSTEM .....	f 32
SPECTRUM MICRODRIVE BOOK (Logan) .....	f 32
*SPECTRUM SHADOW ROM DISASSEMBLY .....	f 45
*THE MICRO CLOAK AND DAGGER BOOK codes, cryptogr. ...	f 36
*THE PURPLE PLANET - MicroPROLOG for the Spectrum ..	f 35
*SPECTRUM + LOGO - complete LOGO listing .....	f 16
*SPECTRUM ASSEMBLY LANGUAGE COURSE .....	f 69
compleet met tape met assembler en monitor programma.	

## - boeken voor de Sinclair QL :

*SINCLAIR QL LEREN PROGRAMMEREN .....	f 24,50
THE SINCLAIR QDOS COMPANION (Pennell) .....	f 36
QL GAMESMASTER (Ewbank) .....	f 39
QL ADVANCED USER GUIDE (Dickens) .....	f 79
*QL SUPERBASIC - The Definitive Handbook (Jones) ...	f 59
INSIDE THE SINCLAIR QL (Naylor) .....	f 36
QL SUPERBASIC - a programmer's guide .....	f 36
THE QL BOOK OF GAMES (Hurley) .....	f 36
*ADVANCED QL MACHINE CODE (Denning) .....	f 45
QL ASSEMBLY LANGUAGE PROGRAMMING (Opie) .....	f 59
QUICK QL MACHINE LANGUAGE (Giles) .....	f 39
ASSEMBLY LANGUAGE PROGRAMMING ON THE SINCLAIR QL ..	f 39

## ACTUELE EN NIEUW BINNENGEKOMEN SOFTWARE VOOR SPECTRUM / QL

### \*\*\*\*\* SPECTRUM 128 software : \*\*\*\*\*

MATCHDAY 128 .....	f 45
*SAMANTHA FOX STRIP POKER 128 ...	f 39
GLADIATOR 128 .....	f 45
*SNEEVO'S WHIRLED .....	f 45

### - arcade games en adventures :

*GHOSTS & GOBLINS .....	f 36
*QUAZATRON .....	f 39
*BIGGLES .....	f 45
*THE PLANETS .....	f 45
CYBERUN .....	f 45
*BATMAN .....	f 36
*THEY SOLD A MILLION II .....	f 45
*SABOTEUR! .....	f 45
GREEN BERET .....	f 36
*HEAVY ON THE MAGICK .....	f 45
*COMMANDO .....	f 39
TAU CETI .....	f 49
V .....	f 36
ZOIDS .....	f 39
SUPERBOWL .....	f 45
RAMBO .....	f 39
*BOMB JACK .....	f 36
*SPITFIRE 40 spectrum .....	f 49
4 CRASH SMASHES .....	f 45
*WEST BANK .....	f 36
*BACK TO SKOOL .....	f 35
*LORD OF THE RINGS .....	f 79

### -- nuttige programma's :

TASWORD III cartridge .....	f 69
*BETA BASIC 3.0 .....	f 69
TASCOPY screen dump .....	f 45
*OCP ART STUDIO cassette versie ..	f 69
OMNICALC II spreadsheet .....	f 69
*ASTRONOMER II planetarium .....	f 49
MINI OFFICE SPECTRUM .....	f 29
*WHAM - THE MUSIC BOX .....	f 49

### -- denkspelen en simulaties :

MONOPOLY .....	f 49
CLUEO .....	f 49
SUPERCHESS 3.5 .....	f 49
*WATERLOO - war game .....	f 49
FRANK BRUNO'S BOXING .....	f 35
HYPERSPORTS (Konami) .....	f 39
*TOMAHAWK .....	f 49
*YIE AR KUNG FU .....	f 39
WAY OF THE TIGER .....	f 45
*SKYFOX .....	f 39
WORLD SERIES BASEBALL .....	f 34
*WORLD SERIES BASKETBALL .....	f 35
MATCHDAY - 3D voetbal .....	f 39
*WINTER GAMES .....	f 39
*ELITE 3D ruimte simulatie .....	f 65

### - utilities :

SUPERCODE III 150 routines .....	f 65
PAINT PLUS (opvolger Paintbox) ..	f 49
TRANSFORMER (microdrive util.) ..	f 45
THE ARTIST .....	f 69
UDG ontwerper .....	f 36

### - programmeertalen :

LASER BASIC COMPILER .....	f 65
LASER BASIC .....	f 65
ABERSOFT FORTH .....	f 69
HISOFT PASCAL compiler .....	f 125
HISOFT C - compiler .....	f 125
HISOFT DEVPC assembler .....	f 69
WHITE LIGHTNING microdrive v... ..	f 95
THE COLT - Hisoft Basic compiler ..	f 65

### - Adventure selectie :

WORM IN PARADISE .....	f 49
HACKER .....	f 39
SHADOW OF THE UNICORN +interface ..	f 79
BORED OF THE RINGS .....	f 39
SORDERONS SHADOW (Beyond) .....	f 49
MARSPORE space trilogy .....	f 45
ARNHEM - wargame .....	f 45
*THE PRICE OF MAGIK .....	f 45

### QL SOFTWARE

*HISOFT DEVPC QL .....	f 185
*TECHNI QL (2D CAD package) .....	f 195
*METACOMCO PASCAL .....	f 399
*QL ROM SUPERBASIC EXTENSIONS ..	f 149
MATCHPOINT .....	f 79
HYPERDRIVE .....	f 79
MONQL ROM .....	f 149
Q.D.OCTOR .....	f 95
BRIDGE II QL .....	f 79
LATTICE C .....	f 140
GRAPHI QL .....	f 110
PSION CHESS .....	f 95

winkel open van woensdag t/m zaterdag tussen 11.00 t/m 17.00 (maandag/dinsdag gesloten) - alle prijzen inclusief BTW  
verzendkosten f 6 per bestelling - onze NIEUWE WINTER 1986 CATALOGUS is nu uit! vraag hem aan! (gratis).

microcomputer tijdschriften boeken en software

dealer aanvragen welkom



QL

## QL SUPERBASIC - THE DEFINITIVE HANDBOOK

QL

Het boek met de bovenstaande titel acht ik voor QL-bezitters van zo grote waarde, niet alleen als alternatief voor de meegeleverde "QL User Guide"(!), dat ik er hier graag aandacht voor vraag.

Het werd geschreven door Jan Jones, zoals op de kapt vermeld: de "designer and writer of Sinclair's QL SuperBASIC language". Als zodanig, zegt zij in haar "Introduction", is zij ontevreden over andere geschriften die (vaak op onjuiste wijze) slechts een kale opsomming geven van wat in SuperBASIC kan en niet kan. Volgens haar behoort de gebruiker te weten waarom bepaalde dingen werken zoals ze dat doen en hoe de informatie intern opgeslagen wordt. In mijn boek zullen de zaken tenminste juist zijn, verzekert ze. Na oplossing met de "Beginner's Guide" van Uw aanloopproblemen zal het van onschatbare waarde zijn: meer zegt zij niet over die "QL User Guide". Gezien haar dankwoord aan Sinclair Research Ltd is dat echter niet zo verwonderlijk. Verder bedankt zij nog haar collega Tony Tebby, die tevens de maker van de "QL Toolkit" is.

De "CONTENTS" kon ik hier niet integraal overnemen, maar ik heb de hoofdstukken van cursiefjes voorzien, deels daaruit gehaald:

CONTENTS overzicht van 4 pagina's met onderwerpen en keywords

1. Introduction *or, why we did it*
2. Notes on definitions, examples and internal storage units
3. Procedures, Methods and Functions *parsing v. execution*
4. The Nametable *namelist, variable values, de spil vh systeem*
5. The Expression Evaluator *stacks, coercion, FPR, operators*
6. Control Structures *IF-THEN-ELSE, SELECT ON, (ON)-GO TO/SUB*
7. Loops *REPEAT/FOR-NEXT-EXIT-END REPEAT/FOR*
8. Data Handling, Input, Output and the File System *Network*
9. Creating and Running Programs *EDIT, AUTO, RENUM, LIST, SAVE*
10. Screen Control *MODE, WINDOW, BLOCK, CURSOR, CSIZE, UNDER*
11. Graphics *scaling, drawing, filling, turtle gr., animation*
12. Machine Code and Memory Access *loading, saving, executing*
13. The Calendar and Clock *SDATE, ADATE, DATES, DATE, DAYS*
14. The QL Sound *range of sounds, BEEP parameters, BEEPING*
15. The Syntax Graphs *'railway lines' definition of SuperBASIC*
16. The Keywords *rapid guide to names: type, action and syntax*

## APPENDICES

- A. The character set, ASCII codes, hex conversion
- B. Errors and what they mean
- C. The SuperBASIC Tokens
- D. The SuperBASIC Storage Area
- E. The QL Memory Map

INDEX A comprehensive guide to all topics *liefst 50 pagina's!*

De opzet is logisch, het hoe en waarom wordt helder beschreven. Ook als naslagwerk zeer aanbevolen. Minpuntjes moest ik zoeken: Die paperback heeft wel een handzaam formaat, maar blijft helaas nooit netjes openliggen, tenzij omgekeerd, of met iets erop. Voor sommigen zal de taal een nadeel zijn. Zelf prefereer ik ook mijn moedertaal. Dat lijkt me echter onhaalbaar voor zo'n boek. Tenslotte vind ik de prijs niet gering, hoewel f 59,- zeker niet ongebruikelijk is voor een buitenlands boek van 264 bladzijden.

Het besproken werk verscheen in 1985 bij McGraw-Hill Book Company (UK) Ltd, ISBN 0-07-084784-3, en werd voor IMPULS beschikbaar gesteld door COMPUTERCOLLECTIEF in A'dam, waar U het kunt kopen of bestellen: zie hun advertentie en stand 111 in De Jaarbeurs.



-----  
SP

TEKENTIPJES

SP  
-----

Wanneer U een tekenprogramma hebt gemaakt en U probeert daarmee een cirkel te tekenen die buiten het tekenveld komt, dan stopt de computer. Dit gebeurt bijvoorbeeld bij "CIRCLE 100,75,100". Om dit tegen te gaan vervangt U in Uw tekenprogramma

CIRCLE x,y,r

door

POKE 23613,PEEK 23613-2:

CIRCLE x,y,r: POKE 23613,PEEK 23613+2

Bij het voorbeeld worden x, y en z opeenvolgend 100, 75 en 100. Deze cirkel wordt getrokken voorzover dat gaat, waarna het programma niet meer stopt, maar voortgaat met de volgende opdracht.

-----  
U kunt de wachttijd voordat een toets gaat repeteren verkorten door een kleinere waarde te POKEN op adres 23561 (minimaal 0). Deze systeemvariabele REPDEL heeft de waarde 35 na inschakeling. Ook een grotere waarde is natuurlijk mogelijk (maximaal 255). Dit is gemakkelijk voor mensen die niet willen dat bij eenmaal een toets indrukken er meerdere letters snel achterelkaar komen.

-----  
Om bijvoorbeeld voor een fruitautomaat draaiende kersen en dergelijke te krijgen kunt U het principe van dit programma nemen:

FOR a=0 TO 255: PRINT AT 10,0: "IMPULS":

PAUSE 5: POKE 23606,a: NEXT a: POKE 23606,0

Voor vruchten moet U zelf een "character set" boven RAMTOP maken en de systeemvariabele CHARS, op de adressen 23606 (L) en 23607 (H), de adressen van die "set" minus 256 laten doorlopen via a.

-----  
Met dit ene simpele regeltje verkrijgt U een draaiende driehoek:

PLOT 43,36: DRAW 122,122,10000

Maar ook een cirkel is mogelijk:

PLOT 70,40: DRAW 100,100,100000

En wat dacht U van een draaiende ster?:

PLOT 50,137: DRAW 22,22,1000000

U kunt ook een leuk plaatje krijgen door eerst de driehoek op Uw scherm te zetten en daarna de ster zo, dat die er overheen komt:

PLOT 43,36: DRAW 122,122,10000:

PLOT 50,128: DRAW 20,20,1000000



SP

DIE EIGENZINNIGE HCC-HOND FIDO EN DE SPECTRUM

SP

De HCC-leden weten al (zo'n beetje) wie FIDO is: het symbool van een netwerk van "bulletinboards", waarmee men berichten kan uitwisselen, ook over de oceaan heen. "Wat moet dat nou, wat is een bulletinboard nou weer, wat heb ik daaraan met mijn Spectrum en wat komt daar weer voor kijken" zou je je nu kunnen afvragen. Op elk van deze vragen zal ik een antwoord proberen te geven.

Een bulletinboard is letterlijk een prikbord, waarop (bijna) ieder een berichtje kan prikken, zodat ieder ander dat kan lezen, erop kan reageren, persoonlijk of algemeen. De micromarktadvertenties in de HCC-Nieuwsbrieven vervullen maar een deel van die functie, je brievenbus een ander deel daarvan. Maar! als je een vraag, oproep of advertentie leest, hoe vaak komt het er dan van dat je ook reageert? Hier kun je de advertentie lezen en direct via je toetsenbord reageren. Een persoonlijk bericht blijft persoonlijk, als je dat wilt, maar als je iets te koop wilt aanbieden kunnen kopers zich direct melden via hun toetsenbord. Ook berichten naar andere knooppunten in het FIDO-net zijn mogelijk en die kosten veel minder dan wanneer je rechtstreeks met iemand in het buitenland wilt corresponderen. Je berichten worden (binnenkort) collectief verzonden naar een centraal punt in dat andere land, zodat het maken van de verbinding maar eenmaal hoeft te geschieden. Via FIDO kun je dan voordelig antwoord krijgen.

Ook kun je ("free-domain") software van het bulletinboard overnemen ("down-loaden") of invoeren ten behoeve van anderen ("up-loaden"). Dat zijn soms heel aardige dingen, voor niks of voor "geen geld".

Zo'n landelijk knooppunt voor een bepaald merk computers heeft ook nog een ander voordeel. Door via het lokaal bereikbare FIDO-knooppunt een vraag te zenden aan het FIDO-knooppunt dat er voor de Spectrum is, bereik je meer Spectrumklanten dan als je alleen maar een papiertje bij de supermarkt opprikt. FIDO werkt gericht. Maar je bereikt ook Spectrumgeïnteresseerden uit het hele land en daarbuiten. Daarbij kun je allerlei specialistische vragen stellen: "Een Spectrumprogramma om familiestambomen mee te maken" of "Games waarin een Stier voorkomt" of "wie weet de oplossing voor ...". Daarom is het ook voor jou leuk, interessant en wellicht lonend.

"Lonend", wat is er dan zoal voor nodig? Niet zo heel veel. Een "modem" is nodig, een toestel om de computer op de telefoonlijn te kunnen aansluiten. Die zijn er in soorten. Het gaat al als je er over een beschikt waarmee met 1200/75 baud gewerkt kan worden (de Viditelnorm) of als je er een hebt met 300 baud full-duplex. Die komen steeds voordeliger op de markt: al vanaf zo'n f 200,-- of goedkoper. Modems die van alles hebben, zoals meer snelheden (baud-rates), die automatisch "een telefoonnummer kunnen kiezen" (auto-dial) of automatisch "de telefoon aan de computer kunnen koppelen wanneer je gebeld wordt" (auto-answer) zijn natuurlijk duurder, maar niet per se nodig voor dit doel.

Verder heb je natuurlijk een koppeling tussen je Spectrum en dat modem nodig. Een "RS232-interface" is daarvoor uitstekend. Velen hebben die al in hun "ZX INTERFACE I" of hun printerinterface. Tenslotte: deze communicatie vereist ook nog speciale software.



-----  
SP

INTERFACE-1-COMPATIBEL MODEMINTERFACE

SP  
-----

In IMPULS 10 beschreef Rik Koevoets een modeminterface om met de VIDITEL- en BULLETIN-BOARD-systemen te kunnen werken.

Het enige nadeel van dat interface is dat het niet compatibel is met ZX-INTERFACE-1. Er is dan ook speciale software voor nodig.

Bij het schema dat U nu voorgeschoteld krijgt hebt U dit nadeel niet: dit interface is in zekere mate compatibel met IF-1.

IF-1, en dus ook dit interface, gebruikt poort #P7 voor in- en output. Met BIT 0 schrijft hij naar buiten en met BIT 7 kijkt hij de buitenwereld in.

Dit interface is het beste te bouwen op VEROBOARD. Let goed op dat U de juiste aansluitingen gebruikt en controleer alles twee keer voordat U het gaat gebruiken. Bij verkeerde aansluitingen kunt U de ULA en/of VIDEORAMS van Uw Spectrum kapot maken!

Als het interface goed gebouwd is, kunt U het aansluiten op Uw Spectrum, nadat U daar eerst de spanning van verwijderd hebt. Sluit daarna de spanning weer aan.

Daar U geen ROM heeft op dit interface, kunt U de EXTENDED BASIC van IF-1 niet gebruiken. Wel kunt U dit interface gebruiken met de VIDITELprogramma's van Rik Koevoets en Jan Breedenbeek.

Het interface kunt U op de volgende manier testen.

Sluit op de uitgangen een voltmeter aan met een meetbereik van 0 tot 10 volt en RUN daarna het volgende programmaatje.

```
10 OUT 247,1: PAUSE 100
20 OUT 247,0: PAUSE 100
30 GOTO 10
```

Wanneer het interface goed is ziet U de voltmeter om de twee seconden tussen de 0 en ongeveer 5 volt slingeren.

-----  
Jack Raats - Noorddonk 107 - 4651 ZD Steenbergen  
-----

#### vervolg van FIDO

Voor het FIDO-net moet je eigenlijk HCC-lid zijn, maar je kunt er al wat rondsnuffelen zonder dat te zijn. Ook zijn er vele andere "bulletin-boards", waar je vaak een gasttoegangsnummer kunt krijgen voordat je echt als lid bekend bent en tot in alle uit- hoeken van het BBS (Bulletin-Board-System) mag snuffelen.

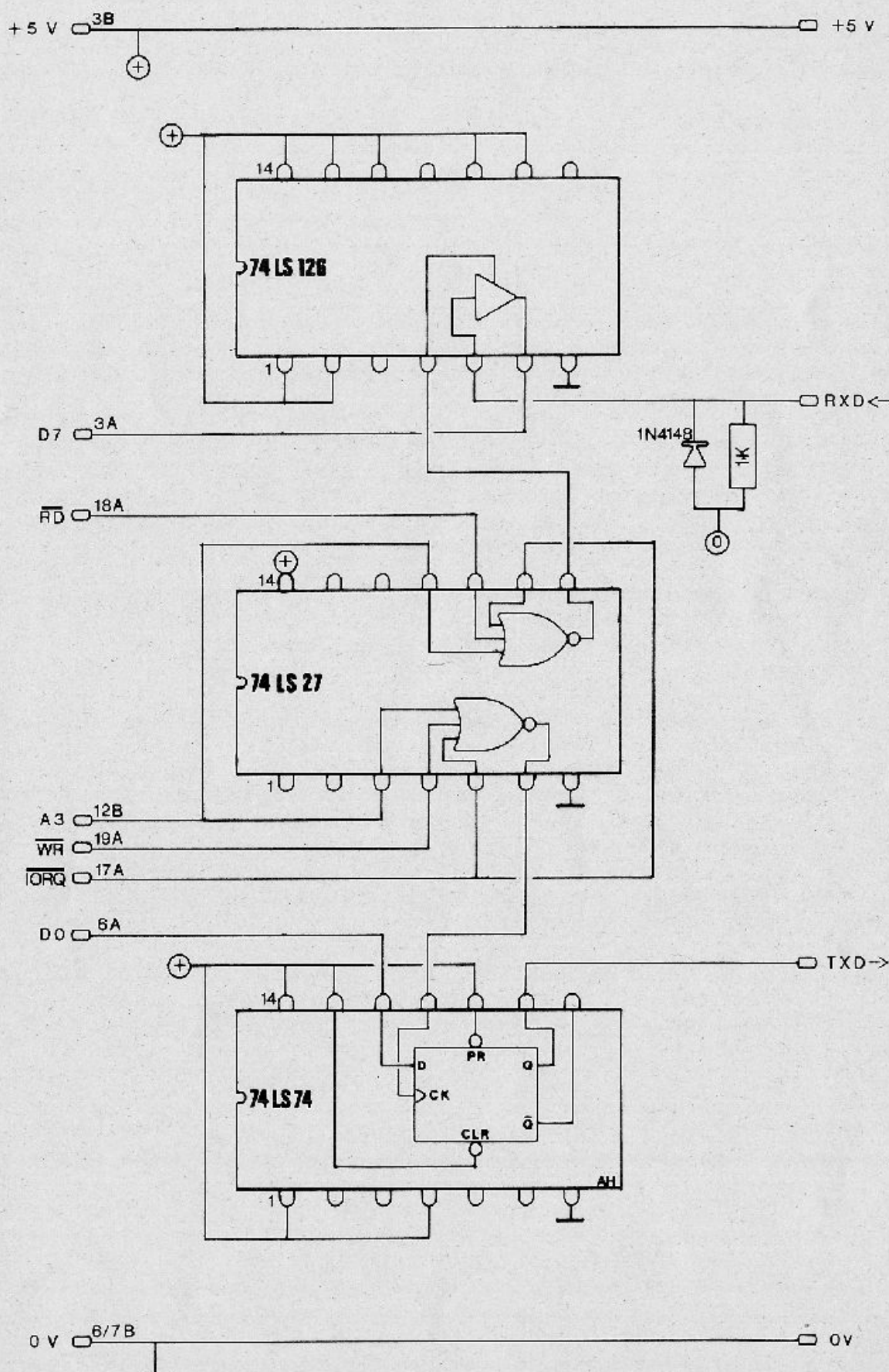
Een erg bekend BBS is de VIDIBUS van Viditel. MEMOCOM is ook zo- iets, maar allebei vragen ze weer geld om er lid van te worden. FIDO vraagt echter dan pas geld als je berichten wilt verzenden.

Voor een aantal computermerken zijn er bepaalde FIDO-knooppunten aangewezen. Voor onze Spectrum is dat: FIDO-ZWOLLE (in Kampen). Ik heb eind juni geconstateerd dat men er een rubriek heeft ingericht voor ons! ik was echter de eerste die daar wat intikte. De beheerder van FIDO-Zwolle heeft nu A gezegd en wij zullen nu met z'n allen de rest van het alfabet moeten invullen, ja, zelfs alle 256 CHR#'s! De SINCLAIR-GG is een van de grootste en meest actieve binnen de HCC. Laten we dus zien dat we dit BBS vullen!

-----  
Pieter Schim van der Loeff

's-Gravenhage  
-----





SCHEMA VAN HET INTERFACE-1-COMPATIBELE MODEMINTERFACE



-----  
SP

EEN MASTERFILEPROBLEEM

SP  
-----

In een poging om nette adresstickers door MASTERFILE af te laten drukken stuitte ik op een probleem. Wie heeft er een oplossing?

Niet iedereen heeft hetzelfde aantal voorletters. Als je ook nog met titulatuur rekening wilt houden dan moet "Prof. jhr. mr. dr. ir. E.G.P.H.M. van der Pott" een heel ander formaat hebben dan "dhr. W. Quarles van Ufford" (om mijn eigen naam nu eens niet ten voorbeeld te stellen. Voor alfabetisch ordenen is de achternaam van de eerste ("Pott") kort, maar voorletters en titulatuur zijn met 41 bytes lang. Als het nodig is, en in één geval is dat zo bij mij, komen er na "dhr. W. " 33 spaties voor de (lange) achternaam, wat niet mooi staat. Het alternatief is dat Van der Pott enkele van zijn ijverig bestudeerde titels kwijt raakt, in mijn adressering tenminste, hetgeen me een vriend, cliënt of wat dies meer zij kan kosten.

Maar als je naar de display van MASTERFILE kijkt en je hebt een veld BRIGHT gedefinieerd, dan zie je dat MASTERFILE wél overweg kan met de veldlengte. Waarom dan niet in de printout? Overigens staan op het scherm de velden wél weer op vaste plaatsen. Derhalve probeerde ik het volgende:

```
5000 DIM Z$(3,40): GO TO USR R
6000 IF C$(1)="V" THEN LET q=LEN C$: LET Z$(1)=C$(2 TO )
6010 IF C$(1)="N" THEN LET Z$(1,q+1 TO )=C$(2 TO )
6020 enz.
```

Maar dat werkt dus niet! De gedachtengang hierbij was als volgt: Eerst zet ik van die C\$ die met de "V" van Voorletterveld begint de lengte in q, waarna ik de inhoud van dit veld in Z\$(1) zet. Vervolgens neem ik de inhoud van die C\$ die met de "N" van Naamveld begint, en laat ik die achternaam een positie verderop in Z\$(1) beginnen, waardoor daar precies een spatie aan voorafgaat. Maar kennelijk komt deze routine het Naamveld eerder tegen dan het Voorletterveld, want ik krijg steevast de foutmelding: "2 Variable not found 6010:1".

Wie kan mij helpen om MASTERFILE de adressen te laten afdrukken zoals ik vind dat het hoort. Ik weet dat er meer mensen zijn die dit zouden willen, maar denken dat het onmogelijk is. Hopelijk kunnen we in het volgende nummer lezen hoe het wel kan.

-----  
Pieter Schim van der Loeff's-Gravenhage  
----------  
ERRATUM:

MONTAGETEKENING BIJ BANKSWITCHING

IMPULS 11-33  
-----

Er is een verbinding weggevallen tussen: IC1-lip 6 en IC2-lip 3. Gelukkig is die wel getekend in het bijbehorende schema eronder.

Met excuses, Albert.

  
-----



SP

PRINTEN ALSOF HET GEDRUKT STAAT

SP

In het vierde deel van deze serie ga ik de grafische functies behandelen. Helaas wijken de diverse printers op dit punt nogal wat van elkaar af. Mocht een programmaatje dat als voorbeeld is gebruikt niet direkt het gewenste resultaat opleveren, ga dan rustig wat experimenteren. Uw printer of computer kan er niet door kapot gaan en u leert er uw printer wel beter door kennen. Komt u er niet uit, dan zijn wij gaarne bereidt u op een van de volgende gebruikersdagen in De Bron verder te helpen.

Succes met uw duik in het grafische en nog een kleine tip toe:

Screendumpprogramma's kunnen vele malen worden versneld als u ze

"door een compiler haalt".

De onderstaande programma's kunnen zonder meer worden gecompileerd met de MCODER 2 en de FP-COMPILER.

#### 4. GRAFISCHE FUNKTIES

\* ESC \*+m+n1+n2 \*  
oooooooooooooooooooo

Naam: 8-pin bit image mode selection

Code: CHR\$ 27;"\*";CHR\$ m;CHR\$ n1;CHR\$ n2;

Functie: met deze functie bent u in staat uit het onderstaand schema een regelbreedte, punt dichtheid en printersnelheid te kiezen. Die kunt u kunt gebruiken om b.v. een plaatje uit een programma op papier te zetten. De m staat voor de gewenste keuze uit de lijst en n1 & n2 geven het aantal te printen bits weer volgens de berekening:  
 $n2 = \text{int} (\text{aantal bits} / 256)$   
 $n1 = \text{aantal bits} - 256 * n2$

Schema:

m	dots over 8 inch	max dots	inch/sec
0	480 single density	480	18
1	960 dual density	960	9
2	960 double speed dual d	480	18
3	1920 quadruple density	960	9
4	640 CRT graphics	640	9
5	576 plotter graphics	576	13.5
6	720 CRT graphics	720	9

Opmerk.: U dient er wel voor te zorgen dat uw printerinterface sommige codes, die u voor het printen in de grafische mode gaat gebruiken, niet in "keywords" omzet!

Bij interface 1 kiest u daartoe voor het B-kanaal en bij het MS-tronicsinterface verandert u de poke C,182 in C,140. Raadpleeg voor uw aanpassing uw leverancier.



```

10 REM esc * test/screencopy vo
or AVT 120 & AVT 180
20 FOR i=5 TO 80 STEP 5
30 CIRCLE 85,85,i
40 NEXT i
50 LPRINT CHR$ 27;"3";CHR$ 22;
60 FOR i=0 TO 6
70 FOR y=0 TO 43
80 LET a=(43-y)*4: LET b=a+1: L
ET c=a+2: LET d=a+3
90 LPRINT " ";CHR$ 27;"*";CHR
$ i;CHR$ 0;CHR$ 2;
100 FOR x=0 TO 255
110 LET e=192*(POINT (x,d))+48*(
POINT (x,c))+12*(POINT (x,b))+3*(
POINT (x,a))
120 LPRINT CHR$ e;CHR$ e;; NEXT
x
130 LPRINT "" : NEXT y: NEXT i

```

\* ESC K+n1+n2 \*

Naam: Normal density bitimage  
Code: CHR\$ 27;"K";CHR\$ n1;CHR\$ n2;

Funktie: Identiek aan ESC \* in de m=0 mode. De data die u in binair horizontaal schrijft, worden dan vertikaal afgedrukt met het hoogste bit boven. Stel u wilt een recht-hoek tekenen van 8 punten hoog en 4 punten breed, dan moeten de gegevens er als volgt uitzien:

Resultaat:	128	X X X X
	64	X . . X
	32	X . . X
	16	X . . X
	8	X . . X
	4	X . . X
	2	X . . X
	1	X X X X
		1 2 3 4

1e printgegevens: 1+2+4+8+16+32+  
64+128 = 255  
2e printgegevens: 1+128 = 129  
3e printgegevens: 1+128 = 129  
4e printgegevens: 1+2+4+8+16+32+  
64+128 = 255

De te printen code ziet er dan als volgt uit:  
LPRINT CHR\$ 27;"K";CHR\$ 4; CHR\$ 0;CHR\$ 255;CHR\$ 129;CHR\$ 129;CHR\$ 255

Opmerk: Als voorbeeld weer het plaatje uit het ESC \* programma, maar nu met wat verschillende print-formaten voor de AVT 100.

```

10 REM esc K test/screencopie v
oor de AVT 100
20 FOR i=5 TO 80 STEP 5
30 CIRCLE 85,85,i
40 NEXT i
50 INPUT "1=A4 2=A5 3=label";a
60 IF a>0 AND a<4 THEN GO TO a*
100
70 GO TO 50
100 LPRINT CHR$ 27;"3";CHR$ 18;
110 FOR x=0 TO 127
120 LET a=x*2: LET b=a+1
130 LPRINT " ";CHR$ 27;"K";CHR
$ 16;CHR$ 2;
140 FOR y=0 TO 175
150 LET c=56*(POINT (a,y))+7*(PO
INT (b,y))
160 LPRINT CHR$ c;CHR$ c;CHR$ c;
170 NEXT y
180 LPRINT ""
190 NEXT x
199 STOP
200 LPRINT CHR$ 27;"3";CHR$ 22;
210 FOR y=0 TO 43
220 LET a=(43-y)*4: LET b=a+1: L
ET c=a+2: LET d=a+3
230 LPRINT " ";CHR$ 27;"K";CHR
$ 0;CHR$ 2;
240 FOR x=0 TO 255
250 LET e=192*(POINT (x,d))+48*(
POINT (x,c))+12*(POINT (x,b))+3*(
POINT (x,a))
260 LPRINT CHR$ e;CHR$ e;
270 NEXT x
280 LPRINT ""
290 NEXT y
299 STOP
300 LPRINT CHR$ 27;"3";CHR$ 22;
310 FOR y=0 TO 21
320 LET a=(21-y)*8: LET b=a+1: L
ET c=a+2: LET d=a+3: LET e=a+4: L
ET f=a+5: LET g=a+6: LET h=a+7

```



```

330 LPRINT " ";CHR$ 27;"K";CHR
$ 0;CHR$ 1;
340 FOR x=0 TO 255
350 LET i=128*(POINT (x,h))+64*(
POINT (x,g))+32*(POINT (x,f))+16*
(PPOINT (x,e))+8*(POINT (x,d))+4*(
POINT (x,c))+2*(POINT (x,b))+(POI
NT (x,a))
360 LPRINT CHR$ i;
370 NEXT x
380 LPRINT ""
390 NEXT y

```

\* ESC L+n1+n2 \*  
 00000000000000000000

Naam: dual density bit image  
 Code: CHR\$ 27;"L";CHR\$ n1;CHR\$  
 n2;  
 Functie: als ESC \* in de m=1 mode  
 Opmerk.: hier een voorbeeld van  
 een screencopyprogram-  
 ma voor de AVT 80, waar-  
 bij de gegevens direkt  
 uit het videogeheugen  
 worden gelezen en eerst  
 tijdelijk worden opge-  
 slagen achter in het ge-  
 heugen.

```

10 POKE 23658,8
20 IF INKEY$<>"" THEN GO TO 20
30 INPUT "Zwart/wit Inverse Cop
y ";a$
40 IF a$(1)="C" THEN GO TO 100
50 IF a$(1)="Z" THEN FOR a=2252
8 TO 23295: POKE a,56: NEXT a: GO
TO 10
60 IF a$(1)="I" THEN FOR a=1638
4 TO 22527: POKE a,255-PEEK a: NE
XT a: GO TO 10
70 GO TO 10
100 LET a=42000
110 FOR b=0 TO 2
120 FOR c=0 TO 7
130 FOR d=0 TO 1
140 FOR e=0 TO 31
150 LET f=16384+b*2048+c*32+d*10
24+e
160 LET g=128
170 LET h=PEEK (f)
180 LET i=PEEK (f+256)

```

```

190 LET j=PEEK (f+512)
200 LET k=PEEK (f+768)
210 FOR l=0 TO 7
220 LET m=0
230 IF h>=g THEN LET m=m+192: L
ET h=h-g
240 IF i>=g THEN LET m=m+48: LE
T i=i-g
250 IF j>=g THEN LET m=m+12: LE
T j=j-g
260 IF k>=g THEN LET m=m+3: LET
k=k-g
270 POKE a,m: LET g=g/2: LET a=
a+1: NEXT l
280 IF d=1 THEN PRINT AT c+b*8,
e;" "
290 NEXT e: NEXT d: NEXT c: NEX
T b
300 INPUT "STEL PRINTER IN, DRU
K <enter>";a$
310 LPRINT CHR$ 27;"e"
320 LPRINT CHR$ 27;"3";CHR$ 24;
330 FOR a=0 TO 31
340 LPRINT CHR$ 27;"L";CHR$ 0;C
HR$ 5;
350 FOR b=0 TO 31
360 LET c=PEEK (42000+a*32+b)
370 LPRINT CHR$ c;CHR$ c;CHR$ c
;CHR$ c;CHR$ c;
380 NEXT b: LPRINT "": NEXT a
390 LPRINT CHR$ 27;"2";CHR$ 7"
400 STOP

```

\* ESC Y+n1+n2 \*  
 00000000000000000000

Naam: Double speed, dual dens.  
 Code: CHR\$ 27;"Y";CHR\$ n1;CHR\$  
 n2;  
 Functie: zie ESC \* mode m=2

\* ESC Z+n1+n2 \*  
 00000000000000000000

Naam: Quadruple-density bit  
 Code: CHR\$ 27;"Z";CHR\$ n1;CHR\$  
 n2;  
 Functie: zie ESC \* mode m=3

De beschikbare ruimte is vol, dus  
 u moet de ESC -beschrijving tot  
 de volgende keer tegoe houden.





## KOMIN COMPUTERSYSTEMEN

NEDERLANDS GROOTSTE

sinclair SPECIALIST

HCC-dagen  
stand 4046

## T060 TRANSFORM KEYBOARD



In Engeland als beste aangemerkt door de vele extra mogelijkheden. Nu nieuwste uitvoering met 65 toetsen, w.o. 15 funktietoetsen en nummeriek toetsenblok. Gegraveerde toetsen. Aan/uit switch. Geschikt voor inbouw van Interface 1 en voedingsunit.

f 225,00

SAGA 3

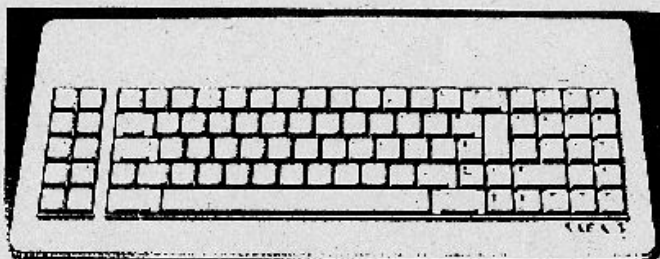
ELITE

Het superieure keyboard voor de Spectrum.  
Dit IBM look like keyboard is de top-per voor de Spectrum of Spectrums.

## SPECTRUM HARDWARE

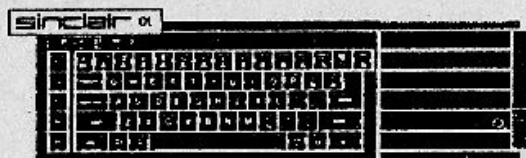
A090 Multiface One (copieert elk programma)	f195,00
T225 Saga Elite keyboard (IBM-look)	f295,00
C002 DK Tronics keyboard (nieuwe uitvoering)	f129,00

S4070 Seikosha SP 1000 printer	f899,00
T070 Beta diskinterface (nieuw met magic button)	f449,00



f295,00

VRAAG ONZE GRATIS CATALOGUS!



## SINCLAIR QL

Zeer snelle en krachtige home-computer met de 68008 van Motorola als processor, 128 KB (RAM) geheugen, super basic programmeertaal, professioneel toetsenbord, twee ingebouwde disk drives. Wordt compleet geleverd inclusief 4 software pakketten.

495

QL  
OMEGA

Q101 Omega QL 1 (Grootboek)	
Q102 Omega Q1 2 (Inkoop)	f 259,00
Q103 Omega QL 3 (Verkoop)	f 159,00
Q104 Omega QL 4 (Voorraad)	f 159,00

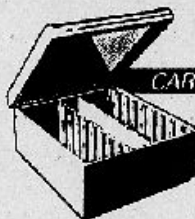
## Q100 Omega QL 1-2-3-4

Geïntegreerd Nederlands Administratiepakket voor de QL bestaande uit Grootboek, Inkoop, Verkoop en Voorraad. De modules zijn ook afzonderlijk te gebruiken. Bij gebruik van Diskdrives wordt een topmanu bijgeleverd die snelle wisseling van programma's mogelijk maakt.

BINNENKORT LEVERBAAR: CST/THOR DE QL COMPATIBLE EN IBM LOOK, OPVOLGER VAN DE SINCLAIR QL. INFORMATIE BESCHIKBAAR.

CST THOR

## CARTRIDGE BOX 1065



Een 'must' voor elke microdrivegebruiker.  
Opslag van 20 cartridges.  
Meerdere units koppelbaar.

f24,00

## QL HARDWARE

Q720 Delta Disk Interface (met parallelprinterpoort)	f599,00
Q722 Delta Disk Interface + 128K RAM	f799,00
Q725 Geheugenuitbreiding 128K RAM Delta	f200,00
Q726 Geheugenuitbreiding 256K RAM	f499,00
Q727 Geheugenuitbreiding 512K RAM	f599,00
Q728 Geheugenuitbreiding 256K RAM en Toolkit ROM	f535,00
Q729 PCML Diskinterface 256K RAM en Toolkit ROM	f1098,00
Q800 Stofhoes QL	f 24,00
Q810 Tweeweg uitbreidingspoort QL	f185,00
S4075 Seikosha SP 1000 QL Printer	f749,00

## WIJZIGINGEN VOORBEHOUDEN

Onze showroom is geopend  
ma/vrij van 9 tot 5 uur.

INFORMATIE EN BESTELLEN  
KAN OOK TELEFONISCH TUSSEN  
10.00 EN 16.00 UUR  
040-455660

Postbus 1805  
5602 CA Eindhoven  
De Greefstraat 15a  
5622 GJ Eindhoven  
Telefoon 040 - 456660  
Telex 59032 Nodel NL  
Bank: Rabo 15.90.05.701  
ABN 52.82.75.615  
Giro: 46.43.401

VERZENDINGEN ONDER REMBOURS OF BIJ VOORUITBETALING



QL

ONTBREKENDE KEYWORDS TOCH OP DE QL

QL

Dat SuperBASIC op de QL een echte "super"-basic is, zal door iedereen die ook maar 10 minuten op deze machine heeft geprogrammeerd worden beaamd, maar dat er nog niet alles op mogelijk is merk je pas na een wat diepgaander studie van het lijvige handboek.

Op een gegeven moment merk je, dat er b.v. in de Spectrum en in het daarvoor geschreven programma Beta Basic een aantal handige keywords zitten, waarvoor je op de QL nogal wat moet "aanklooien" om het zelfde effect te bereiken.

Nu is het wel zo, dat de QL steengoed is in het werken met procedures en met die kennis ben je in staat, de betreffende keywords aardig te benaderen.

Hieronder heb ik wat van deze routines opgeschreven en ook wat andere die mij bij het programmeren in basic wel handig lijken.

Helaas ben ik bij het maken van deze procedures nog op een probleem gestuit, waar ik op dit moment nog niet uit ben:

Binnen een procedure kan ik de dimensie van een bestaand array of array\$ niet veranderen, vandaar dat in sommige procedures een nieuwe array(\$) of variabele wordt aangemaakt om de uitkomst van de betreffende procedure door te spelen naar het basicprogramma.

Als afsluiting kom ik dan weer met het gebruikelijke bedelpraatje om uw routines naar de redactie van dit blad te sturen, want ik denk echt, dat een heleboel lezers hun voordeel kunnen doen met hetgene dat u hebt bedacht. Dat is toch eigenlijk de grondslag van onze vereniging en van dit blad: HET ZOVEEL MOGELIJK VAN ELKAAR OP KUNNEN STEKEN, ZODAT WE MET ELKAAR ONZE MACHINES ZO GOED MOGELIJK LEREN BEGRIJPEN.

Na deze preek wens ik u nog veel programmeerplezier met de onderstaande programmadeeltjes.

Rob van Staalduinen.

**SORTEER\$:** Deze routine is te gebruiken om 2-dimensionale tekstarrays te sorteren. Het is mogelijk om de tekstarray van A tot z te sorteren (oplopend) of van z naar A (aflopend).

Gebruiken: **SORTEER\$** arraynaam\$, (0 voor oplopend/1 voor aflopend)

Resultaat: arraynaam\$

```

10000 DEFine PROCedure SORTEER$ (sorter$,richting$)
10001   LOCAL teller,schuiver,len_s,sort$(DIMN(sorter$,1),DIMN(sorter$,2))
10002   len_s=DIMN(sorter$)
10003   FOR teller=1 TO len_s
10004     CLS#0:PRINT#0\\"Nog"!len_s-teller!"items te sorteren!"
10005     schuiver=teller
10006     REPEAT schuif
10007       IF sorter$(teller)>=sort$(schuiver-(schuiver>0))OR NOT schuive
10008         r:EXIT schuif
10009       sort$(schuiver)=sort$(schuiver-1):schuiver=schuiver-1
10010     END REPEAT schuif
10011     sort$(schuiver)=sorter$(teller)
10012   END FOR teller
10012   IF richting$<>0
10013     FOR teller=0 TO len_s
10014       sorter$(len_s-teller)=sorter$(teller)
10015     END FOR teller

```



---

```

10016 ELSE
10017 FOR teller=0 TO len_s
10018     sorter$(teller)=sort$(teller)
10019 END FOR teller
10020 END IF
10021 END DEFine

```

Voorbeeld: U wilt uw adressenbestand namen\$(500,25) oplopend sorteren,  
dan toetst u in: `SORTEER$ namen$,0`  
Na deze opdracht staat namen\$ in alfabetische volgorde.

\*\*\*\*\*

**SORTEER:** Deze routine is te gebruiken om getallenarrays te sorteren.  
Het is mogelijk om de getallenarray van 0 tot 9 te sorteren  
(oplopend) of van 9 naar 0 (aflopend)..

Gebruiken: `SORTEER arraynaam,(0 voor oplopend/1 voor aflopend)`

Resultaat: arraynaam

```

10100 DEFine PROCedure SORTEER (sorter,richting$)
10101 LOCAL teller,schuiver,len_s,sort(DIMN(sorter))
10102 len_s=DIMN(sorter)
10103 FOR teller=0 TO len_s
10104     CLS#0:PRINT#0\\"Nog"!len_s-teller!"items te sorteren!"
10105     schuiver=teller
10106     REPEAT schuif
10107         IF sorter(teller)>=sort(schuiver-(schuiver>0))OR NOT schuiver:
10108             EXIT schuif
10109         sort(schuiver)=sort(schuiver-1):schuiver=schuiver-1
10110     END REPEAT schuif
10111     sort(schuiver)=sorter(teller)
10112 END FOR teller
10112 IF richting$
10113     FOR teller=0 TO len_s
10114         sorter(len_s-teller)=sort(teller)
10115     END FOR teller
10116 ELSE
10117     FOR teller=0 TO len_s
10118         sorter(teller)=sort(teller)
10119     END FOR teller
10120 END IF
10121 END DEFine

```

Voorbeeld: U wilt uw debiteurengegevens debi(500) aflopend sorteren,  
dan toetst u in: `SORTEER namen,1`  
Na deze opdracht staat uw grootste debiteur boven in de lijst.

\*\*\*\*\*

**INARRAY\$:** Met deze routine gaat u in een gesorteerde 2-dimensionale  
tekstarray op zoek naar een bepaalt gegeven om te weten te  
komen op welke plaats in de tekstarray dat gegeven staat.

---



---

Gebruiken: INARRAY\$ arraynaam\$,zoekvariabele\$,plaatsvariabele  
 Resultaat: plaatsvariabele

```

11000 DEFine PROCedure INARRAY$ (array$,zoek$,plaats)
11001   LOCAL deler,teller,zoeker
11002   plaats=INT(DIMN(array$)/2+.5):deler=plaats:teller=0
11003   REPEAT zoeker:deler=INT(deler/2+.5):IF deler=1:teller=teller+1
11004     IF zoek$=array$(plaats):EXIT zoeker
11005     IF zoek$>array$(plaats):plaats=plaats+deler*(plaats+deler<=DIMN(
      array$))
11006     IF zoek$<array$(plaats):plaats=plaats-deler*(plaats-deler>0)
11007     IF teller>1:plaats=0:EXIT zoeker
11008   END REPEAT zoeker
11009 END DEFine

```

Voorbeeld: U wilt in uw adressenbestand namen\$() het plaatsnummer weten van de heer Pieterse (staat op namen\$(37) en u zet dit gegeven in zoek\$ (b.v. INPUT "Wie zoeken? ";zoek\$).  
 U tikt dan in: INARRAY\$ namen\$,zoek\$,plaats  
 Na deze opdracht vindt u in de variabele plaats het getal 37, hetgeen dus de plaats van de heer Pieterse in namen\$ is.

\*\*\*\*\*

ONARRAY\$: Met deze routine gaat u in een ON-gesorteerde 2-dimensionale tekstarray op zoek naar een bepaald gegeven om te weten te komen op welke plaats in de tekstarray het gegeven staat.

Gebruiken: ONARRAY\$ arraynaam\$,zoekvariabele\$,plaatsvariabele  
 Resultaat: plaatsvariabele

```

11100 DEFine PROCedure ONARRAY$ (array$,zoek$,plaats)
11101   FOR plaats=1 TO DIMN(array$)
11102     IF zoek$=array$(plaats):EXIT plaats
11103   END FOR plaats
11104   IF zoek$<>array$(plaats):plaats=0
11105 END DEFine

```

Voorbeeld: U wilt uit uw adressenbestand namen\$() het plaatsnummer weten van de heer Verheul (staat op namen\$(413) en u zet dit gegeven in zoek\$ (b.v. INPUT "Wie zoeken? ";zoek\$).  
 U geeft dan in: INARRAY\$ namen\$,zoek\$,plaats  
 Na deze opdracht vindt u in de variabele plaats het getal 413, hetgeen dus de plaats van de heer Verheul in namen\$ is.

\*\*\*\*\*

INARRAY: Deze procedure zoekt in een gesorteerd getallenarray naar een bepaalde waarde en zet het plaatsnummer van deze waarde in plaatsvariabele.

Gebruiken: INARRAY arraynaam,zoekvariabele,plaatsvariabele  
 Resultaat: plaatsvariabele

---

```

11200 DEFine PROCEDURE INARRAY (array,zoek,plaats)
11201   LOCAL deler,teller,zoeker
11202   plaats=INT(DIMN(array)/2+.5):deler=plaats:teller=0
11203   REPEAT zoeker:deler=INT(deler/2+.5):IF deler=1:teller=teller+1
11204     IF zoek=array(plaats):EXIT zoeker
11205     IF zoek>array(plaats):plaats=plaats+deler*(plaats+deler<=DIMN(ar
      ray))
11206     IF zoek<array(plaats):plaats=plaats-deler*(plaats-deler>0)
11207     IF teller>1:plaats=0:EXIT zoeker
11208   END REPEAT zoeker
11209 END DEFine

```

Voorbeeld: U bent op zoek naar die ene debiteur die u f 317.35 schuldig is, maar u weet niet meer wie dat was. Aan de hand van de plaats in de getallenarray kunt u dit alsnog achterhalen. U toetst in: INARRAY debi,317.35,plaats  
In de variabele plaats vindt u nu het antwoord.

\*\*\*\*\*

ONARRAY: Deze procedure zoekt in een Ongesorteerd getallenarray naar een bepaalde waarde en zet het plaatsnummer van deze waarde in plaatsvariabele.

Gebruiken: ONARRAY ongesorteerde arraynaam,zoekvariabele,plaatsvariabele

Resultaat: plaatsvariabele

```

11300 DEFine PROCEDURE ONARRAY (array,zoek,plaats)
11301   FOR plaats=1 TO DIMN(array)
11302     IF zoek=array(plaats):EXIT plaats
11303   END FOR plaats
11304   IF zoek<>array(plaats):plaats=0
11305 END DEFine

```

Voorbeeld: U bent op zoek naar die ene donateur die u f 85.00 heeft betaald, maar u weet niet meer wie dat was. Aan de hand van de plaats in de getallenarray kunt u dit alsnog achterhalen. Het bedrag van f 85.00 staat in de variabele sponsor. U toetst in: INARRAY donaties,sponsor,plaats  
In de variabele plaats vindt u nu het antwoord.

Om duidelijk te kunnen maken wat de verschillen in tijd zijn tussen de INARRAY(\$) en de ONARRAY(\$) routines hieronder een tabelletje van gemiddelde zoektijden in seconden bij verschillende arraylengten.

Lengte	10	25	50	100	250	500	1000	2500	5000	10000
INARRAY	0.11	0.15	0.18	0.21	0.26	0.30	0.34	0.39	0.43	0.47
ONARRAY	0.06	0.12	0.18	0.40	0.84	1.98	3.87	9.21	18.63	41.30

Hieruit blijkt de INARRAY-routine vanaf een arraylengte van 50 veel bruikbaar te zijn, maar bedenk dat de INARRAY-routines alleen maar kunnen worden gebruikt als de array geheel gesorteerd is.



81

DE MELEDA

SP

De Meleda is een puzzel waarbij een samenstel van ringen van een soort lange schalm aan een handvat bevrijd moet zien te worden. Volgens het "Boek van de maand" - april '78 - "Spelen met Puzzels" (De Bezige Bij) dateren deze "Chinese Ringen" uit ca. 200. Het volgende programma is een korte, snelle simulatie hiervan. IMPULSOFTCASSETTE SC 03 bevat een versie met kleuren en geluid.

----- ZX81-UITVOERING -----

```

1 LET R=CODE INKEY$ -28
2 IF R<1 OR R>7 THEN GOTO 1
3 IF M$( TO R-1)<>V$(8-R TO ) THEN GOTO 9
4 LET M$(R)="C" AND NOT CODE M$(R)
5 PRINT M$:Z:AT 9,10:
6 LET Z=Z+1
7 GOTO 1
8 SAVE "MELEDA"
9 PRINT AT 8,9:"****MELEDA****";
  TAB 9:"* * *";
  TAB 9:"*1234567*****":AT 9,10:
10 LET M$=" * "
11 LET V$=" O"
12 LET Z=1
13 GOTO 1

```

SAVE dus met RUN 8  
Deze tekens in wit  
op zwart (GRAPHICS)  
maar de  
spaties wit laten.

- OPDRACHT -

- LAAT ALLE RINGEN VERSCHIJNEN EN DAARNA WEER VERDWIJNEN MET -  
- ZO WEINIG MOGELIJK AANSLAGEN OP DE TOETSEN 1 TOT EN MET 7. -

Bij elke fout keert de begintoestand terug: vindt U dat evenwel veel te moeilijk, dan kunt U de regels 2, 3 en 4 vervangen door:

```

2 IF R<1 OR R>7 THEN GOTO 9 OR R
3 IF M$( TO R-1)=V$(8-R TO )
  THEN LET M$(R)="O" AND NOT CODE M$(R)

```

Bij deze eenvoudigere versie gaat opnieuw beginnen alleen met 0.

----- SPECTRUMUITVOERING -----

```

1 PRINT AT 8,9:"****MELEDA****";
  TAB 9:"* * *";
  TAB 9:"*1234567*****";
  LET m$=" * "; LET v$=" O"; LET z=1
2 LET r=CODE INKEY$-CODE"0": IF r<1 OR r>7 THEN GO TO 2
3 IF m$( TO r-1)<>v$(8-r TO ) THEN GO TO 1
4 LET m$(r)="O" AND m$(r)=" "; PRINT AT 9,10:m$:z:
  LET z=z+1: GO TO LN EXP LN EXP LN EXP 2

```

SAVE "MELEDA" LINE 1  
De letters, cijfers  
en de \* in INV VIDEO  
O niet

Voor de eenvoudiger versie plaatst U achter "GO TO 2" in regel 2 "AND r", waarna U de daaropvolgende regels moet vervangen door:

```

3 IF m$( TO r-1)=v$(8-r TO )
  THEN LET m$(r)="O" AND m$(r)=" "
4 PRINT AT 9,10:m$:z: LET z=z+1:
  GO TO LN EXP LN EXP LN EXP 2 [ vertraging nodig ]

```

----- EdW -----

FDE8 2A4F5C	INIT	LD	HL,(CHANS)	FE41 18EF	JR	SPACES
FDEB 110F00		LD	DE,15	FE43 FEA5	NO_TAB CP	#A5
FDEE 19		ADD	HL,DE	FE45 3805	JR	C,NO_TOK
FDEF 110BFE		LD	DE,START	FE47 D6A5	SUB	#A5
FDF2 73		LD	(HL),E	FE49 C3100C	JP	PR_TOK
FDF3 23		INC	HL	FE4C FE80	NO_TOK CP	#80
FDF4 72		LD	(HL),D	FE4E 3804	JR	C,NO_GRA
FDF5 3E3F		LD	A,63	FE50 0601	LD	B,#01
FDF7 D35F		OUT	(PAC),A	FE52 18DE	JR	SPACES
FDF9 3EFF		LD	A,255	FE54 FE0D	NO_GRA CP	#0D
FDFB D37F		OUT	(PBC),A	FE56 2007	JR	NZ,NO_CR
FDFD 3E02		LD	A,2	FE58 E5	PUSH	HL
FDFE D37F		OUT	(PBC),A	FE59 219FFE	LD	HL,TELLER
FE01 3E01		LD	A,1	FE5C 3600	LD	(HL),#00
FE03 D33F		OUT	(PBD),A	FE5E E1	POP	HL
FE05 3E40		LD	A,64	FE5F FE20	NO_CR CP	#20
FE07 329EFE		LD	(AANTAL),A	FE61 3811	JR	C,LPRINT
FE0A C9		RET		FE63 F5	PUSH	AF
FE0B FE06	START	CP	#06	FE64 E5	PUSH	HL
FE0D 2814		JR	Z,PR_COM	FE65 219FFE	LD	HL,TELLER
FE0F FE16		CP	#16	FE68 7E	LD	A,(HL)
FE11 2804		JR	Z,PR_AT	FE69 2B	DEC	HL
FE13 FE17		CP	#17	FE6A BE	CP	(HL)
FE15 202C		JR	NZ,NO_TAB	FE6B 3E0D	LD	A,#0D
FE17 E1	PR_AT	POP	HL	FE6D D454FE	CALL	NC,NO_GRA
FE18 E1		POP	HL	FE70 23	INC	HL
FE19 D9		EXX		FE71 34	INC	(HL)
FE1A E1		POP	HL	FE72 E1	POP	HL
FE1B F5		PUSH	AF	FE73 F1	POP	AF
FE1C 1F		RRA		FE74 000000	LPRINT	DEFB 00,00,00
FE1D 3001		JR	NC,UPDATE	FE77 000000	DEFB	00,00,00
FE1F 41		LD	B,C	FE7A 000000	DEFB	00,00,00
FE20 78	UPDATE	LD	A,B	FE7D F5	PUSH	AF
FE21 1806		JR	COUNT	FE7E F5	PUSH	AF
FE23 F5	PR_COM	PUSH	AF	FE7F CD541F	BUSY	CALL BREEK
FE24 3A9EFE		LD	A,(AANTAL)	FE82 D2000D	JP	NC,#0D00
FE27 CB3F		SRL	A	FE85 DB3F	IN	A,(PBD)
FE29 219FFE	COUNT	LD	HL,TELLER	FE87 CB4F	BIT	1,A
FE2C 96		SUB	(HL)	FE89 20F4	JR	NZ,BUSY
FE2D 380B		JR	C,PRINCR	FE8B F1	POP	AF
FE2F E1		POP	HL	FE8C D31F	OUT	(PAD),A
FE30 C8		RET	Z	FE8E AF	XOR	A
FE31 47		LD	B,A	FE8F D33F	OUT	(PBD),A
FE32 3E20	SPACES	LD	A,#20	FE91 3C	INC	A
FE34 CD54FE		CALL	NO_GRA	FE92 D33F	OUT	(PBD),A
FE37 10F9		DJNZ	SPACES	FE94 F1	POP	AF
FE39 C9		RET		FE95 FE0D	CP	#0D
FE3A CDF81F	PRINCR	CALL	PRI_CR	FE97 C0	RET	NZ
FE3D F1		POP	AF	FE98 3E0A	LD	A,#0A
FE3E FE06		CP	#06	FE9A CD74FE	CALL	LPRINT
FE40 CB		RET	Z	FE9D C9	RET	

Bij gebruik van een assembler moeten deze "equations" hieraan voorafgaan:

PAC EQU 95  
 PAD EQU 31  
 PBC EQU 127  
 PBD EQU 63  
 BREEK EQU #1F54  
 PRI\_CR EQU #1FF8  
 PR\_TOK EQU #0C10  
 CHANS EQU 23631

Voeg voorts twee "declarations" toe:

AANTAL DEFB #00      TELLER DEFB #00

Een andere geschikte plaats voor deze routine is de printerbuffer. Als startadres kunt U dan 23296 nemen.



SP

EEN CENTRONICSINTERFACE VOOR ZELFBOUW

SP

De meeste vragen die we bij onze servicetelefoon en vraagbaak in De Bron gesteld krijgen gaan over printers, printerinterfaces en hoe je de programma's moet aanpassen om die te kunnen gebruiken. Dure interfaces ook, als je weet welke onderdelen daarin zitten.

Voor deze problemen hebben we de volgende oplossingen gevonden. Een interface dat U ongeveer f 55,- aan onderdelen gaat kosten, waarvoor we de benodigde aanpassingen van de diverse programma's in de IMPULS gaan publiceren: aan het slot van dit artikel kunt U alvast die voor "SPECTRAL WRITER" en "TASWORD TWO" aantreffen. Tevens komt er een IMPULSOFTcassette waarop, behalve de hierna volgende routine om LPRINT & LLIST bij dit interface te kunnen gebruiken, ook routines voor SCREENCOPY & A4-COPY voorkomen.

Het interface bestaat uit een '2 80A PIO' en een weerstandje van 10 kilo-ohm, meer niet: dat kunt U in het schema verderop zien. Natuurlijk hebt U verder nog een edgeconnector voor Uw Spectrum en een kabel met een centronicsplug voor Uw printer nodig. Voor dit interface hebben we een printje ontworpen, dat U bij de schrijver kunt bestellen, of aan de balie in De Bron kunt kopen.

#### PRINTERROUTINE IN MC VOOR HET GEBRUIK VAN LPRINT EN LLIST

Nevenstaande routine laat U op adres 65000 beginnen met een hex-loader, die uit IMPULS 09-40 bijvoorbeeld, of met een assembler, waarvoor de aanvullende gegevens onder de routine bestemd zijn. Voor hexloaders zijn er 182 bytes: 2A 4F 5C 11 0F ... 74 FE C9. Na de laatste op adres 65181 moeten er nog 2 bytes vrij blijven. Verlaag vooraf RAMTOP met CLEAR 64999. Initialiseer deze routine met RANDOMIZE 65000, waarna U LPRINT en LLIST kunt gebruiken.

Soms zult U deze printerroutine moeten aanpassen aan Uw printer. Daartoe is het mogelijk om maximaal drie CHR\$'s tegen te houden. Als voorbeeld neem ik de "SEIKOSHA GP-500A"-printer. Volgens de "print control function table" geeft CHR\$ 20 een Carriage-Return zonder een Line-Feed. Voor de Spectrum betekent CHR\$ 20 echter: ga invers printen op het beeldscherm. Wanneer U dus een LISTing afdruckt waarin CHR\$ 20 zit, dan zal Uw printer twee regels over elkaar drukken. Vandaar dat we dan CHR\$.20 moeten onderdrukken. Voor elke CHR\$ die U wilt onderscheppen voert U drie POKE's uit:

```
: POKE 65140,254 : POKE 65143,254 : POKE 65146,254 :
: POKE 65141,... : POKE 65144,... : POKE 65147,... :
: POKE 65142,200 : POKE 65145,200 : POKE 65148,200 :
```

Op de puntjes vult U van ieder te onderscheppen CHR\$ de CODE in, dus 20 als U CHR\$ 20 uit het voorbeeld wilt tegenhouden. In de routine ziet U dat er op deze adressen (FE74 t/m FE7C hex) nullen staan, hetgeen betekent dat er niets wordt onderschept.

Tijdens het initialiseren wordt ook het aantal der tekens dat er per regel wordt afgedrukt ingesteld op 64. U kunt dit veranderen door een ander aantal, bijvoorbeeld 80, op adres 65030 te POKE'n.

Deze printerroutine geeft na elke Carriage-Return een Line-Feed. Wanneer U deze Line-Feed wilt onderdrukken, omdat Uw printer een automatische Line-Feed heeft, dan toetst U POKE 65173,201 in. Wilt U dit dan weer ongedaan maken dan POKEt U op dat adres 254.



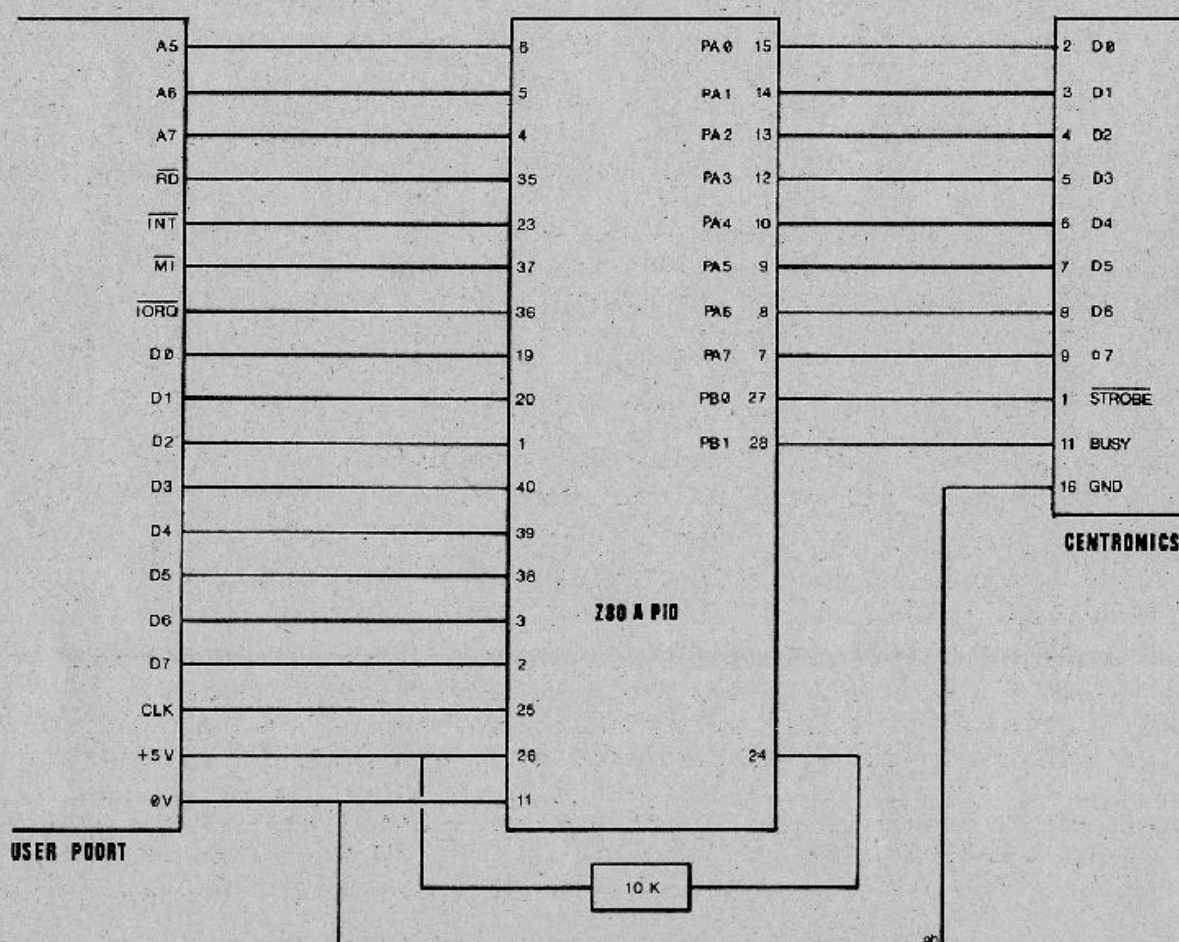
# Donateurs:

Op de komende ECC-beurs zullen wij de nieuw in te schrijven donateurs een cassette aanbieden. Omdat wij vinden dat u als vaste donateur ook van deze aanbieding moet kunnen profiteren, doen wij u het volgende voorstel:

A. u stort voor 15 december uw donatie 1986 à f25,- en komt de cassette op 17 jan. in de bron halen

B. u maakt f27,50 over en vermeld daarbij welke cassette u wenst. U krijgt deze dan spoedig toegestuurd.

Rob v. Staalduinen  
Tenn. St. Impuls





## AANPASSING VAN "SPECTRAL WRITER" AAN DIT CENTRONICSINTERFACE

Dit kunt U erg gemakkelijk gedaan krijgen op de volgende manier:

1. Laad "Spectral Writer" en ga naar het MENU.
2. Kies optie D: "Door naar BASIC".
3. Voer de onderstaande POKE's uit.
4. Geef RUN en ga wederom naar het MENU.
5. Kies optie A: "Wijzig de parameterwaarden".
6. Verander de INTERFACECODE in: 2.
7. Kies optie B: "Save dit programma".

POKE 60957,247	POKE 60970, 62	POKE 60983, 79
POKE 60958, 62	POKE 60971, 1	POKE 60984, 32
POKE 60959, 63	POKE 60972,211	POKE 60985,244
POKE 60960,211	POKE 60973, 63	POKE 60986,241
POKE 60961, 95	POKE 60974,205	POKE 60987,211
POKE 60962, 62	POKE 60975, 84	POKE 60988, 31
POKE 60963,255	POKE 60976, 31	POKE 60989,175
POKE 60964,211	POKE 60977,210	POKE 60990,211
POKE 60965,127	POKE 60978, 0	POKE 60991, 63
POKE 60966, 62	POKE 60979, 13	POKE 60992, 60
POKE 60967, 2	POKE 60980,219	POKE 60993,211
POKE 60968,211	POKE 60981, 63	POKE 60994, 63
POKE 60969,127	POKE 60982,203	POKE 60995,201

## AANPASSING VAN "TASWORD TWO" AAN DIT CENTRONICSINTERFACE

Dit kunt U erg gemakkelijk gedaan krijgen op de volgende manier:

1. Laad "Tasword Two" en ga naar het MENU.
2. Kies optie B om naar BASIC te gaan.
3. Geef RANDOMIZE USR 59806.
4. Voer de onderstaande POKE's uit.
5. Geef opnieuw RANDOMIZE USR 59806.
6. Geef RUN en ga wederom naar het MENU.
7. Kies optie G en geef bij "DEFINE GRAPHICS" een ENTER.
8. Verander de "INTERFACE CONTROL CODES" code 1: 0  
zoals dat hiernaast staat aangegeven. code 2: 0
9. Kies optie T om "Tasword Two" te save. code 3: 0  
code 4: 64836

POKE 64914,205	POKE 65038, 62	POKE 65050, 32
POKE 64915, 5	POKE 65039, 2	POKE 65051,250
POKE 64916,254	POKE 65040,211	POKE 65052,241
POKE 65029,245	POKE 65041,127	POKE 65053,211
POKE 65030, 62	POKE 65042, 62	POKE 65054, 31
POKE 65031, 63	POKE 65043, 1	POKE 65055,175
POKE 65032,211	POKE 65044,211	POKE 65056,211
POKE 65033, 95	POKE 65045, 63	POKE 65057, 63
POKE 65034, 62	POKE 65046,219	POKE 65058, 60
POKE 65035,255	POKE 65047, 63	POKE 65059,211
POKE 65036,211	POKE 65048,203	POKE 65060, 63
POKE 65037,127	POKE 65049, 79	POKE 65061,201

Veel succes met het centronicsinterface en de routines daarvoor.

SP

DE ZX-MICRODRIVES DEEL 3

SP

In de "SINCLAIR GEBRUIKER" van maart 1985 heeft Rob Baas al het een en ander verteld over technische opbouw en tapeformat van de Microdrive (Md). Dit zal ik hier dus niet herhalen, maar ik wil nog wel de belangrijkste details geven:

- In een cartridge zit circa 5 meter eindloze band, die met een snelheid van 76 cm per seconde ronddraait, dus 16 keer zo snel als in een cassette recorder!
- Een "file" staat op zo'n band altijd in stukjes die "records" genoemd worden en elk 512 bytes aan "data" bevatten.

Over die FILES gaan we het nu hebben, want daarvan bestaan vijf verschillende vormen bij de Md. Bijeenbehorende gegevens worden een "file" genoemd: Engels voor "bestand". Dus geen verwantschap met de beruchte rij langzaamrijdende en stilstaande automobielen hoewel u als geheugensteun een file op een Md-cartridge best wel hiermee mag vergelijken. Zo'n file (om verwarring te voorkomen zal ik verder dit woord gebruiken ipv bestand) is namelijk niets anders dan een soms lange rij gegevens, die een programma of gegevens daarvoor bevat. Er zijn dus vijf soorten, maar eigenlijk zijn er slechts twee hoofdsorten: PROGRAMMAFILES en DATAFILES.

PROGRAMMAFILES zijn files waarmee u eigenlijk al vertrouwd bent, het gaat namelijk om files die u maakt en terugleest met behulp van de SAVE-, LOAD-, VERIFY- en MERGE-commando's. De Md werkt hiermee op precies dezelfde manier als uw cassette recorder, alleen moet u nu \*"m":1: inlassen om aan te geven dat u met de Md (1) werkt. Er zijn vier varianten mogelijk bij programmafiles door toevoeging van LINE, CODE, DATA of SCREEN\$, net zoals u dat bij uw cassette recorder gewend bent. SAVE \*"m":1:"naam" LINE 10 bijvoorbeeld, zet een programma zodanig op de cartridge in Md 1, dat het vanaf regel 10 begint te RUNnen als u het teruglaadt. Veel meer valt er over programmafiles eigenlijk niet te vertellen, alleen zijn er nog wel een aantal verfijningen aangebracht:

- Als u een programma met de naam "run" op Md SAVet, zal dit automatisch worden geladen na RUN & ENTER, mits uw Spectrum "leeg" was, dus direkt na inschakelen of NEW. Dit is bij veelgebruikte programma's makkelijk. Alleen Md 1 kan hiervoor gebruikt worden.
- Een programma dat u met een LINE-rnr geSAVED hebt kan niet geMERGED worden, zoals dat bij het cassettesysteem wel kan. Probeer u dat toch, dan krijgt u de melding "MERGE error".
- Als u tijdens het laden van een BASIC-programma op BREAK drukt wordt de Spectrum ge-reset, zodat u dan alles kwijt bent. Deze (en de vorige) nieuwigheid is aangebracht om BASIC-programma's beter te beveiligen tegen inbreken.

Dan nu de andere soort die u op de Md kunt zetten: de DATAFILES. Voor de duidelijkheid: dit zijn geen arrays die u met SAVE..DATA bewaart, maar gewoon gegevens van een bepaalde soort, die u met PRINT-statements opslaat en met INPUT weer terugleest. Dat opslaan en teruglezen gaat echter altijd in dezelfde volgorde: van het begin naar het einde van de file. Daarom worden deze files SEQUENTIEEL genoemd: de gegevens leest u altijd terug in dezelfde volgorde als waarin ze werden opgeslagen.

Als het mogelijk zou zijn geweest de file in willekeurige volgorde terug te lezen, dan zouden we te maken hebben gehad met een "RANDOM ACCESS FILE", ofwel een willekeurig toegankelijke file.



Zoals al vermeld kunt u met PRINT de gegevens (getallen, strings enz) opslaan. Dat kan echter niet zomaar, eerst moet u de file OPENEN voor SCHRIJVEN. Dat doet u met OPEN #. In deel 1 heb ik u al het een en ander verteld over dit statement, toen we bezig waren met de in- en uitvoerorganen van de Spectrum. Dit OPEN # moet gevolgd worden door een stroomnummer en een filenaam.

Nu is het zo dat we apparaten als beeldschermen en toetsenborden in principe ook tot de files kunnen rekenen omdat daar ook gegevens naartoe kunnen gaan of vandaan kunnen komen. Deze apparaten hebben - zoals we al in deel 1 zagen - de stroomnummers 0 t/m 3: 0 en 1 voor het toetsenbord, 2 het scherm en 3 de ZX-printer.

De stroomnummers 4 t/m 15 zijn dus nog vrij. Hieruit kunnen we er een kiezen om aan een Md-file te koppelen, bijv 4. We kunnen dan met PRINT #4:... gegevens naar die file sturen. Nadat we dat gedaan hebben moeten we de Spectrum vertellen dat we geen gegevens meer naar die file sturen: we dienen die file te SLUITEN.

Dat doen we met het CLOSE #-statement. In het volgende voorbeeld zal ik het gebruik van zo'n datafile demonstreren. De getallen 1 tot en met 100 worden daarbij opgeslagen in een datafile.

```
10 OPEN #4:"m":1:"getallen"
20 FOR n=1 TO 100: PRINT #4:n: NEXT n
30 CLOSE #4
```

In regel 10 openen we een file met de naam "getallen" op Md 1. Met de FOR-NEXT-lus van regel 20 PRINTen we vervolgens de getallen naar die file, en met regel 30 sluiten we die weer.

Nu hebben we die getallen dus in die Md-file gezet, en willen we ze nu weer teruglezen; dan gaat dat als volgt:

```
10 OPEN #4:"m":1:"getallen"
20 FOR n=1 TO 100: INPUT #4:a: PRINT a: NEXT n
30 CLOSE #4
```

Ook hier op regel 10 een OPEN-statement, waarmee we de file weer OPENEN, nu echter voor LEZEN, terwijl er in het vorige voorbeeld voor schrijven werd geopend.

Maar hoe weet de Spectrum nu of we een file voor lezen openen of voor schrijven, als we voor beide hetzelfde statement gebruiken? Het antwoord op die vraag vindt de Spectrum door de Md-cartridge af te zoeken naar de filenaam. Als hij de opgegeven naam daarbij tegenkomt wordt er geopend voor lezen, en anders voor schrijven. Bij reeds bestaande files kunt u dus alleen lezen, daarom is het onmogelijk om een bestaande file te wijzigen.

Terug nu naar ons laatste voorbeeld. Met de FOR-NEXT-lus van regel 20 lezen we de getallen die we naar de file hebben gePRINT weer terug en daarna sluiten we haar weer. Maar stel nu dat we niet weten hoeveel getallen er in die Md-file opgeslagen zijn. Een FOR-NEXT-lus is dan niet mogelijk. We zouden dan net zolang door moeten lezen tot we aan het eind van de file waren gekomen, waarna we de melding "End of file" zouden krijgen:

```
10 OPEN #4:"m":1:"getallen"
20 INPUT #4:a: PRINT a: GO TO 20
```

Deze "End of file"-melding is meestal niet zo wenselijk. Om deze te voorkomen moet de Spectrum dus van tevoren weten hoeveel gegevens er komen. Dit kan heel eenvoudig door dat aantal, voorafgaande aan de eigenlijke gegevens, naar de file te PRINTen.

Stel dat u een stringarray A\$ bestaand uit N strings in een file wilt opslaan. Dit zou dan als volgt kunnen gebeuren:

```
10 OPEN #4:"m":1:"string"
20 PRINT #4:N
```

```
30 FOR I=1 TO N: PRINT #4:A$(I): NEXT I
40 CLOSE #4
```

Het teruglezen kan dan zo:

```
10 OPEN #4:"m":1:"string"
20 INPUT #4:N
30 FOR I=1 TO N: INPUT #4:A$(I): NEXT I
40 CLOSE #4
```

Wanneer u wat met deze datafiles experimenteert zult u zien dat de uitvoering van een PRINT- of INPUT-statement niet altijd tot het starten van de Md leidt, in de meeste gevallen zelfs niet. Dit komt doordat de gegevens die u opslaat of terugleest niet meteen in de file worden gezet, maar eerst worden opgeslagen in een buffer van 512 bytes. Zodra deze buffer vol is (bij schrijven) of leeg is (bij lezen) wordt zijn inhoud pas weggeschreven naar de Md of wordt er pas een nieuw record ingelezen uit de Md. In verband hiermee is het CLOSE-statement belangrijk. Dit ruimt de in de "Channel information area" voor de file gereserveerde buffer op, waardoor er weer zo'n 600 bytes vrijkomen. Als de file geopend was voor schrijven, moeten de gegevens die nog in de buffer zitten eerst weggeschreven worden en dient dat record de markering "End of file" te krijgen en daar zorgt CLOSE ook voor.

Verder moet ik nog wat kwijt over het gebruik van PRINT en INPUT bij files. U weet dat u bij een INPUT van het toetsenbord altijd ENTER moet intikken na de invoer van gegevens. U geeft ermee aan dat alle gegevens ingevoerd zijn, "het einde van de regel" dus. Als de Spectrum nu een INPUT doet van een Md-file moet hij ook weten wanneer "het einde van de regel" bereikt is.

Zou u bijvoorbeeld PRINT #4:"abc": "def" doen en dit dan weer teruglezen met INPUT #4:A\$ (na sluiten en heropenen van de file!), dan zou A\$ de waarde "abcdef" krijgen terwijl dat misschien niet de bedoeling was. U moet dus bij PRINT altijd voor een nieuwe regel zorgen door een apostrof of zo. Dus PRINT #4:"abc" "def" of PRINT #4:"abc": PRINT #4:"def". Verder leveren aanhalingstekens in een string problemen op omdat de Spectrum dan denkt dat hij aan het einde van de string is terwijl dat nog niet zo is. Dit is echter te omzeilen door INPUT #4: LINE A\$ te gebruiken. Verder dient u te bedenken dat INPUT-statements ook als PRINT-statements kunnen werken, zoals in INPUT "Wat is je naam?":A\$. Dit gaat echter fout bij een INPUT uit een Md-file, want die is immers alleen voor lezen geopend! U krijgt dan een foutmelding: "Writing to a 'read' file". Gebruik dus na INPUT uit files nooit "promptstrings" en ook nooit komma's, AT, TAB etc.

Er bestaat nog een manier om een datafile op de Md te lezen. Dat gaat met de INKEY\$-functie. Hiermee kunt u teken voor teken uit een Md-file inlezen. U kunt dit als volgt proberen met de file "getallen" uit het voorbeeld eerder in dit artikel:

```
10 OPEN #4:"m":1:"getallen"
20 PRINT INKEY$#4:GO TO 20
```

Echt nuttig zal de INKEY\$-functie hierbij niet zijn: slechts in bepaalde gevallen is die te prefereren boven INPUT.

Tot zover de Md-files. Ik hoop dat ik nu het mysterieuze van de datafiles voor u heb ontrafeld. Volgende keer hoop ik wat verder in te kunnen gaan op de mogelijkheden die Interface-1 biedt voor de machinecodeprogrammeur (en dat zijn er heel wat!). Ook kijken we dan nog naar het MOVE-statement dat soms erg nuttig kan zijn.

----- JAN BREDENBEEK -----





-----  
ZX

DE BUGS BIJ DELING EN -65536

ZX  
-----

Op het interessante artikel "SPECTRUMBUGS" van Jan Bredenbeek in IMPULS 10-44 wilde ik graag reageren.

Zijn uitleg bij de "deling"-fout overtuigt me echter niet zo. Volgens Jan zou deze fout aan te tonen zijn met "PRINT 1/2-.5", dat ca "2.3E-10" geeft. Maar deze uitkomst is positief, dus volgens de Spectrum, en ook volgens de ZX81, is 1/2 groter dan 0.5. Nu wordt 1/2 in "Floating Point Representation": "128 0 0 0 0". Berekening:  $2^7(128-128)*0.5=2^7*0.5=1*0.5=0.5$ , dus geheel exact. Maar 0.5 is in FPR: "127 127 255 255 255". Nu zijn alle bits van de mantisse 1. Berekening:  $2^7(127-128)*(2^7-1+2^6-2+2^5-3+...+2^1-32)=(2^7-1)*(2^7-1+...2^1-32)=0.5*(1-2^7-32)=0.5-2^7-33$ . Tekort:  $2^7-33$ . Die  $1-2^7-32$  kan berekend worden met de formule voor de som van de eerste n termen van een meetkundige rij met aanvangsterm a en reden r :  $a*(1-r^n)/(1-r)$ .

Je zou nu verwachten dat dit ene beetje tekort bij de Spectrum in FPR "96 0 0 0 0" werd, want  $2^7(96-128)*0.5=2^7*(-32)*0.5=2^7-33$ . Maar dat gebeurt niet: 1/2-.5 wordt "97 0 0 0 0" ( $=2^7-32$ )! En het wordt nog vreemder: .5-1/2 blijkt wel 0 op te leveren! Met het volgende programmaatje is alles te bestuderen!

Voor de Spectrum

```
10 DIM a(1)
20 LET var=PEEK 23627
  +256*PEEK 23628
30 INPUT a(1)
40 PRINT a(1)
50 FOR x=var TO var+10
60 PRINT PEEK x:" ":
70 NEXT x
80 PRINT
90 GO TO 30
```

Voor de ZX81-1k

```
10 DIM A(1)
20 INPUT A(1)
30 PRINT A(1)
40 FOR X=0 TO 10
50 LET V=PEEK 16400
  +256*PEEK 16401
60 PRINT PEEK (V+X):" ":
70 NEXT X
80 PRINT
90 GOTO 30
```

Probeer nu eens de volgende getallen of expressies:

-.5, 1/2=.5, .5=1/2,  $2^7-32$ ,  $2^7-33$ ,  $.5+2^7-32$ ,  $.5+2^7-33$ , enz.

Ontdek de vreemde afwijkingen, ook bij de negatieve getallen.

Bij de Spectrum is ook de "-65536"-fout en de "Small Number Representation", dus de "integer"-vorm van getallen, te zien.

Meestal worden getallen die als breuk zijn ingetoetst, bijvoorbeeld 1/4, 3/8 en 5/16, goed in RPF opgeslagen, terwijl dezelfde getallen als .25, .375 en .3125 ingevoerd, net te klein in RPF komen te staan: dus dan gaan zowel Spectrum als ZX81 in de fout.

Ten aanzien van de "-65536"-fout wil ik nog even een lans breken voor de opvatting van het handboek, dat zegt dat gehele getallen van -65535 t/m -1 in de vorm van "131072+(een negatief getal)" opgeslagen worden. Die moeten dus wel als 17-bits lange getallen gezien worden: zie de overeenkomst met 1-byte lange MC-sprongen, zoals "Jump Relative -3", dat we als "JR 253" (256-3) noteren. Bij de ZX80, de Commodore- en de MSX-computers worden "integers" van -32768 t/m -1 opgeslagen als positieve getallen van de vorm "65536+(een negatief getal van -32768 t/m -1)" in twee bytes. Aangezien het bereik van de Spectrum tweemaal zo groot is, moeten de "integers" dan ook als 17-bits-getallen worden beschouwd.



SP	OVERZICHT VAN DE THANS LEVERBARE IMPULSOFTCASSETTES		SP
SC01	KOLENMIJNSCHAT KLAS UITSLAG	Avonturenspeel in de duisternis van Uw SP. Gegevensbestand: leerlingen en rapporten. Berekent en tekent uitslagen in plaatwerk.	
SC02	STADHOUDER FINANCIEEL BASICODE	Regeer een aantal jaren over flipflop-land. Administreer Uw vaste lasten over een jaar. Vertaalprogramma voor NOS-BASICODE-2.	
SC03	INFO/LOAD MASTERMIND IDEM IN ZW/WT DE MELEDA	Aanwijzingen- en laadprogramma. Het overbekende spel in kleuren, maar ... deze versies spelen ook tegen U, en hoe!! Een verrassende oudchinese ringenpuzzel.	
SC04	ISOLATIE SALARIS SLOWLOADER	Spel voor twee spelers. Sluit elkaar in. Berekening van maandsalarissen. Laadt een ZX81-programma in Uw Spectrum.	
SC05	SOLITAIR BOEKHOUDING TAAL	Sla ook die laatste pion in je eentje! Uw grootboek, v&w en balans op saldbasis. Leer vreemde talen mbv Uw Spectrum.	
SC06	BLACK JACK HYPOTHEEK PIANOLA	Eenentwintigen met inzet, 1 t/m 7 spelers. Bereken zelf de gunstigste hypotheekvorm. Laat Uw geliefde pianoconcert weerklinken.	
SC07	TOUWTREKKEN ENERGIE SYNTHSIZER	Reactiespel voor twee sterke spelers. Maakt overzichten van Uw energieverbruik. Uw Spectrum als elektronisch orgel.	
SC08	VIER OP RIJ TOTOMAAT SINCISIZER	Dat rest U, als er eentje op de loop is. Speel in de toto op basis van statistiek. Haal meer uit Uw synthsizer.	
SC09	DUPPIE RPC AUTOMENU	Let op Uw spaarcenten! Rekenprogramma voor schoolkinderen. Laden en starten vanaf Microdrives.	
SC10	PUZZLE TAFELS VARMEN	Laat ze maar schuiven met deze puzzel. Leert Uw kinderen vermenigvuldigen. Toont de gebruikte variabelen en arrays.	
SC11	ZEESLAG DELING FILES	Breng Uw Spectrum eens tot zinken. Maak de getoonde staartdelingen af kinders. File-handling zonder Microdrives.	
SC12	EXCAT NEDERLAND PIXELSCROLL	Menuprogramma voor Microdrives. Leer Uw eigen land kennen per helicopter. Schuiven met schermen.	
SC13	PIRE1 TIMING GGD	Mensen redden met behulp van een vangzeil. Stopwatchprogramma voor de juiste "timing". Berekent de grootste gemene deler.	
SC14	WRIGGLY TOONSTEM CASSETTES	Hoe je ook draait of keert, je gaat eraan. Om U een toontje lager te laten zingen. Index van muziekbandjes, voor Microdrives.	

ART	INHOUDSOPGAVE VAN DEZE IMPULS 12	PAG
Colofon .....		02
Van de redactie, HCC SINCLAIR GG en IMPULSOFT .....		03
Basicversnellingsvoorbeeld - laatste versnelling (?) .....		04
Ervaringen van een QL-beginner: snelheid & SuperBASIC .....		05
De Wafadrive van Rotronics .....		09
Dubbelgrote screendump met de Seikosha-100 via IF1 .....		10
De servicetelefoon van de HCC Sinclair gg .....		11
Computercollectief, advertentie .....		12
QL SuperBASIC - The Definitive Handbook, boekbespreking .....		13
Tekentipjes .....		14
Die eigenzinnige HCC-hond FIDO en de Spectrum .....		15
Interface-1-compatibel modeminterface voor zelfbouw .....		16
Een Masterfileprobleem .....		18
Erratum montagetekening bij bankswitching Impuls 11-33 .....		18
Printen alsof het gedrukt staat .....		19
Komin, advertentie .....		22
Ontbrekende keywords toch op de QL .....		23
De Meleda, een oudchinese ringenpuzzel .....		27
Een Centronicsinterface voor zelfbouw .....		28
De ZX-Microdrives, deel 3 .....		32
CLS # en CLEAR #, onvermelde opdrachten bij IF1 .....		35
De bugs bij deling en -65536 .....		36
Overzicht van de thans leverbare IMPULSOFTcassettes .....		37
Inhoudsopgave van deze IMPULS 12 .....		38
Data-skip, advertentie .....		39

SC15 TORENHANOI LETTERS 3eGRAADS	Verplaats de toren, maar niet uitstekend. Vergroot maximaal 16 tekens per regel. Lost vergelijkingen op van de derde graad.
SC16 VERHUIZEN EMMERS ROTATIE	Kan die kast wel door de deur? Zo ontdekt. Lekker klieren met water. Twee handige routines in machinetaal.
SC17 BO-KA-EI REKENEN AARDMETER	Boter, kaas en eieren in superuitvoering. Rekenprogramma voor schoolkinderen. Geografische afstandsberekening.
SC18 QUEENS BLOKGOLF SPOOLER	Het koninginneschaakprobleem. Blok er niet op, Uw Spectrum kan golfen. Printerspouler.
SC19 PLAATJES TAAL TALOMZET	Bruikbaar in diverse programma's. Taalprogramma voor schoolkinderen. Vertaalt getallen naar andere talstelsels.
SC20 SLURPER BERPI DALONA	Slurp snel water op, Uw Spectrum zinkt. Berekening van het getal PI. Ontwerp "multiple choice"-vragen.
SC21 MOTOR DEFCAR LETTERSO TASMAN	Probeer Uw Spectrum draaiende te houden. Gebruik Uw eigen "character set". UDG's naar de Seikosha-250. Aanvulling voor "Tasword Two".



Bezoek ons op de  
HCC-dagen stand 4031  
in de Margriet hal



## Opus Discovery

nu voor

fl. 495,-

Wij ruilen uw micro-  
drive + interface in  
tot max. f 175,-.  
Zo wordt uw nieuwe  
Opus nog goedkoper !

### A FEATURE-PACKED SYSTEM FOR JUST £149.95 INCLUDING:

- 3.5" 250 K disc drive
- Double density disc interface
- Parallel printer interface
- Joystick interface
- Video monitor interface
- Peripheral through connector
- Built-in power supply
- Utilities on ROM including format and verify
- RAM disc facility
- Random access files fully supported
- Connections for second drive upgrade
- Comprehensive user manual

## AMX MOUSE

= vernieuwd ! =

FL. 275,-

De AMX-Muis is nu nog beter! Met nog fraaiere  
styling en betere grip.  
Ook de bijgeleverde programma's zijn aangepast  
en verbeterd.

Voor al de "extended Basic" is zeer mooi: het  
stelt U in staat Mouse-Control in te bouwen in  
Uw eigen of andere programma's.

Incl. Demo's, Icon-Designer en uitgebreide  
handleiding. Ook te gebruiken met Art Studio!

Data-Skip levert ook programma's voor AMX-Muis, o.a. Dbase AMX.

Topograaf AMX

f 19,-

f 29,-

Videoface digitizer.....f 275,-  
Multiface One, versie 86.2.....f 175,-  
VIX-5000, modem+interface.....f 175,-  
Brother M1109 printer.....f 695,-  
Sinclair QL.....f 495,-  
ZX Spectrum 128.....f 349,-  
ZX Spectrum 128+2.....f 649,-  
Saga 3 Elite, een pracht keyboard...f 295,-  
10 st. 3 1/2 inch diskettes SSDD...f 50,-



BEL VOOR WICHTINGEN, PRIJZEN  
EN ADVIES

07810-20581

# DATA-SKIP, GOUDA

L. Willemsteeg 10-12  
2801 WC Gouda

Bestellingen d.m.v. ingesloten betaalkaart of vooruitbetaling op giro 47.27.958 t.n.v. Data-Skip, Gouda



**hii**  
**SINCLAIR**  
Gebruikers  
Groep

Redactie  
SINCLAIR IMPULS  
Postbus 76  
2260 AB Leidschendam

PORT BETAALD  
DEN HAAG