

KIM GEBRUIKERS CLUB NEDERLAND

Zevende Jaargang Nr. 27

Augustus 1983

De 6502 KENNER is een uitgave van de KIM Gebruikers Club Nederland.

Adres voor het inzenden van en reacties op artikelen voor DE 6502 KENNER:

Willem L. van Pelt,  
Jacob Jordaensstraat 15,  
2923 CK KRIMPEN AAN DEN  
IJSSSEL.

Redactie De 6502 KENNER:

Vaste medewerkers:

Willem van Pelt  
Gerard van Roekel  
Frans Smeehuijzen  
Jaap van Toledo

Freelance medewerkers:

Adri Hankel  
Renee de Hoop  
Fridus Jonkman  
Herman Kuyvenhoven  
Anton Muller  
Koen van Nieuwenhove  
Erwin Rosseeuw

Gehele of gedeeltelijke overname van de inhoud van DE 6502 KENNER zonder toestemming van het bestuur is verboden. Toepassing van gepubliceerde programma's, hardware etc. is alleen toegestaan voor persoonlijk gebruik.

DE 6502 KENNER verschijnt 5 x per jaar en heeft een oplage van 500 exemplaren.

Copyright (C) 1982 KIM Gebruikers Club Nederland.

De voorpagina is een aquareel van een KIM, geschilderd door:

Rinus Vleesch Dubois.

## INHOUDSOPGAVE DE 6502 KENNER NR. 27 AUGUSTUS 1983

1.	Van het bestuur	2.
2.	UITNODIGING Kim Club Bijeenkomst Heemstede	14.
3.	Cassettebibliotheek	3.
4.	Van de redactie	5.
5.	Programmeren met FOCAL ... door J. Jansen	6.
6.	COMMODORE 64 Print Routine ... door Anton Mueller	9.
7.	Nogmaals: De Basic Compressor	12.
8.	USR - routines ... door Ruud Uphoff	13.
9.	Suggesties t.a.v. Basicode ... door Wouter van Goor	18.
10.	Koppeling van JUNIOR aan 8" Floppy Met FD 1771 controller 8" single en 5" double density ... door Koen van Nieuwenhove	19.
11.	BASIC-programma SUPER STAR TREK Nederlandse versie van het bekende spel ... door Adri Hankel	30.
12.	KIM-1 Tekstverwerking met Teletype KSR33 of ASR ... door J.C.J. Beijer	34.
13.	IDSA ID en SA opzoekprogramma ... door Jaap P. van Toledo	37.
14.	Telefoonbeantwoorder ... door Haary Vandendriessche	38.
15.	UART-schakeling voor JUNIOR ... door A.A. Zwart	41.
16.	JAARSTUKKEN	middenpagina's M1-M4
17.	Howcan ... door R.A.F. Bens	M4.
18.	Akoestisch Telefoonmodem Elektuur	2.
19.	Proton's JUNIOR DOS ... door Piet Fransen	3.
20.	Spiraal (KIM) nu voor JUNIOR	3.
21.	Bestellingen bij Proton	3.
22.	HCC-dagen 21 en 22 oktober 1983	5.
23.	Boekenlijst voor 6502-gebruikers	29.
24.	Automatische Register Uitlezing II ... door Paul van Niekerk	48.
25.	Marktinfo	4, 18, 36, 40.

Door mijn werkzaamheden ben ik in de gelegenheid de laatste trend te volgen op o.a. het gebied van hard- en floppy-disk-drives.

Onlangs was ik op bezoek bij de firma Rodim, een jonge fabrikant op het gebied van 15 1/4 inch harddisk-drives. In capaciteiten varieërend van 5 t/m 40 Mb.

Na een bezoek aan de montagehal, welke geheel voldoet aan de eisen van een 'clean-room' waarin een overdruk heerst om alle mogelijke stofdeeltjes buiten te houden en waar de temperatuur zeer constant is, ben ik geconfronteerd met het feit dat dergelijke fabrikanten een zeer hoogwaardig technisch produkt produceren, en kosten noch moeite sparen om het eindprodukt aan de hoogste eisen te laten voldoen.

Het verbaast mij dan ook ten zeerste mensen te ontmoeten, die doodleuk zelf sleutelen aan dergelijke produkten, om vervolgens zeer teleurgesteld te zijn als het niet blijkt te lukken.

Hiermee wil ik diegenen waarschuwen die, vaak uit onwetendheid, hun technisch hoogwaardig produkt ongewild ontregelen.

Verder heb ik tijdens mijn bezoek een zogenaamde 'slim-line' harddisk gezien, waarvan de capaciteit 10 Mb is en de afmetingen: 150 mm lang, 90 mm breed, en 40 mm hoog! U leest het niet verkeerd.

Men kan bij wijze van spreken het hele geval in de zak steken en op stap gaan met 10 Mb informatie.

Ook op het gebied van floppydisk-systemen is de trend aanwezig om steeds meer informatie op een kleiner schijfje te schrijven. De miniaturisering brengt een verfijnder techniek met zich. Ik raad U aan dergelijke technieken zoveel als mogelijk is te volgen.

Ik hoop dan ook de volgende keer dieper op deze materie in te gaan, bijvoorbeeld door middel van een lezing, ondersteund door een demonstratie van dergelijke apparatuur.

Rinus Vleesch Dubois,  
Voorzitter.

---

---

#### AKOESTISCH TELEFOONMODEM ELEKTUUR

In het februari-nummer van Elektuur publiceerden J.J.A. Habets en C.A. Truijens een akoestisch telefoonmodem.  
Onze clubgenoot Adri Hankel deelde mede voor dit modem software te hebben ontwikkeld. Het zit er dus in dat JUNIOR-gebruikers op niet al te lange termijn deze software zullen kunnen toepassen. Er blijkt echter nog een kink in de kabel. Het akoestisch telefoonmodem vertoont een mankement aan de PLL-schakeling van de ontvanger. Men probeert er nu achter te komen wat er precies aan de hand is en op welke wijze het euvel kan worden verholpen. Zodra dit probleem is opgelost wordt een en ander bekend gemaakt.

## PROTON'S JUNIOR D.O.S.

Piet Fransen  
Voordewind 21  
1034 KS Amsterdam  
Tel.: 020 - 319271

Onlangs heeft de Fa. Proton een revisie aan hun monitor uitgevoerd. De veranderingen zijn ook doorgevoerd in de voor JUNIOR beschikbare versie. Deze nieuwe versie heb ik in mijn systeem opgenomen en functioneert uitstekend. De veranderingen zijn alle directe verbeteringen. Als eerste is er een entry-point bijgekomen waarmee het mogelijk is vanuit een gebruikersprogramma direct van de disc-drive gebruik te maken. Nu is het namelijk mogelijk door in de registers X en Y een getal te plaatsen en op de monitor RAM plaats "NAME" een filenaam van zes letters te plaatsen, waarna het programma zelf in de directory deze filenaam interpreteert. Het format voor het lezen en schrijven moet men wel zelf maken. Maar dat gaat met behulp van de bestaande entrypoints zeer gemakkelijk. In mijn Basicroutine heb ik dit reeds gedaan en het functioneert perfect. Met het nieuwe entrypoint is het nu mogelijk geworden om b.v. in de toetsherkenning van de Moser Assenbier het volgende op te nemen: "SAVE PROGRAM 1". Na het intoetsen van return wordt dan de betreffende file op disc gestored, met als filenaam "PROGRAM".

Een tweede uitbreiding is de toets "M", waarmee achter de acht gedisplayde geheugenplaatsen ook de ASCII-waarden van de geheugeninhoud wordt uitgeprint. Als voorbeeld:

```
(M)=2000
53 45 4E 49 4F 52 08 01
SENIOR
```

Deze uitbreiding is handig bij onderzoek in de geheugenplaatsen van een source. Ook is een min-toets ingevoerd, zodat bij geheugenonderzoek niet steeds het adres opnieuw hoeft te worden opgegeven.

Al deze veranderingen hebben geen invloed op de bestaande versie. Alle reeds bestaande functies blijven normaal in werking. De enige verande-

ring in de monitorfile is, dat de adresplaatsen van de twee veranderingen die ik in het artikel "Maak van JUNIOR een SENIOR" heb genoemd, nu zijn veranderd. In punt 5 staat onder D, dat op \$E28B-\$E28E ( 20 CF EA ) staat. Dit is nu geworden adres \$E290-\$E292, waar nu staat ( 20 18 EB ). Voor de initialisatie routine was het adres \$E067-\$E069 waar stond ( 20 93 E0 ), dit is nu geworden adres \$E069-\$E06B waar nu staat ( 20 95 E0 ). Wat op deze geheugenplaatsen kan of moet worden gezet is behandeld in het betreffende artikel.

## SPIRAAL (KIM) NU VOOR JUNIOR

Het programma SPIRAAL uit de 5502 KENNER nr.25 kan op simpele wijze worden aangepast voor de JUNIOR. Men moet het programma via de standaardmonitor opstarten (?), anders gaat het mis.

Adres:	was:	wordt:
022A	8D 07 17	8D F7 1A
0230	AD 07 17	AD F7 1A
028C	8D 07 17	8D F7 1A
028F	20 1F 1F	20 9E 1D
0292	AD 07 17	AD F7 1A
02D5	20 40 1F	20 AC 1D
02C9	20 6A 1F	20 F9 1D

De subroutine CARLO moet volledig worden veranderd (02A2 t/m 02BC):

02A2	A9 7F	CARLO
02A4	8D 81 1A	
02A7	A2 08	
02A9	A0 06	
02AB	84 FC	LOEP
02AD	89 E7 00	
02B0	49 7F	
02B2	20 E3 1D	

02B5	A4 FC
02B7	88
02B8	D0 F1
02BA	60
02BB	EA
02BC	EA

## INHOUD JUNIOR CASSETTE NR. 3 KANT 1 SOURCE MICRO-ADE

ID	NAAM	NR. KIM/6502	SA	EA
01	RENUMBER SYM-BASIC V. JUNIOR	--	5000	6D29
01	RENUMBER SYM-BASIC V. JUNIOR	--	5000	6D29
02	APPEND SYM-BASIC V. JUNIOR	--	5000	5573
02	APPEND SYM-BASIC V. JUNIOR	--	5000	5573
03	BASIC COMPRESSOR V. JUNIOR	25	3000	65CF
03	BASIC COMPRESSOR V. JUNIOR	25	5000	65CF
04	NPRCHA (nieuwe PRCHA-routine)	26	5000	6973
04	NPRCHA (nieuwe PRCHA-routine)	26	5000	6973
05	ADRESS (Micro-Ade verplaatsen)	26	5000	5FB3
05	ADRESS (Micro-Ade verplaatsen)	26	5000	5FB3
06	MASTERMIND R.VISSERS	26	5000	6E2F
06	MASTERMIND R.VISSERS	26	5000	6E2F
07	OUTCLK (KLOK/8 SCHAKELPOORTEN)	26	5000	7BF3
07	OUTCLK (KLOK/8 SCHAKELPOORTEN)	26	5000	7BF3

## INHOUD JUNIOR CASSETTE NR. 3 KANT 2 CODE

ID	NAAM	NR. KIM/6502	SA	EA
01	RENUMBER SYM-BASIC V. JUNIOR	--	0200	059F
01	RENUMBER SYM-BASIC V. JUNIOR	--	0200	059F
02	APPEND SYM-BASIC V. JUNIOR	--	0200	0224
02	APPEND SYM-BASIC V. JUNIOR	--	0200	0224
03	BASIC COMPRESSOR V. JUNIOR	25	0300	05F1
03	BASIC COMPRESSOR V. JUNIOR	25	0300	05F1
04	NPRCHA (nieuwe PRCHA-routine)	26	0300	0386
04	NPRCHA (nieuwe PRCHA-routine)	26	0300	0386
05	ADRESS (Micro-Ade verplaatsen)	26	0200	029F
05	ADRESS (Micro-Ade verplaatsen)	26	0200	029F
06	MASTERMIND R.VISSERS	26	0200	0458
06	MASTERMIND R.VISSERS	26	0200	0458
07	OUTCLK (KLOK/8 SCHAKELPOORTEN)	26	0200	03F7
07	OUTCLK (KLOK/8 SCHAKELPOORTEN)	26	0200	03F7

Programma's in de cassettebibliotheek zijn of origineel van onze clubleden of bewerkingen van programma's uit het First Book of Kim, of anderszins. Alle rechten van de programma's blijven aan de auteurs voorbehouden. De kosten van de cassette zijn kostendekkend en niet bedoeld om de programma's te kopen!  
DE CLUB IS EN WORDT GEEN HANDEL !!  
Kosten per cassette: fl. 12,50.  
Bestellen van de cassette onder opgave van naam, adres, postcode, etc. en nummer van de cassette + naam van het systeem (JUNIOR/KIM) vergezeld van een girobetaalkaart of door overschrijving op postrekening 841433 t.n.v. W.L. van Peit te Krimpen a.d. IJssel. Andere wijzen van bestellen zijn niet mogelijk.

## BESTELLINGEN BIJ PROTON

LDA #7F	zet de poorten
STA PA0D	goed
LDX #08	begin met di6
LDY #06	
STY #FC	
LDA COUNT,Y	haal display op
EOR #7F	initieer
JSR \$1DE3	display en wacht

Voor het bestellen van artikelen van Proton kunnen clubleden voortaan direct bij Proton terecht. De sekretaris heeft aldaar een ledenlijst neergelegd, zodat Proton zelf de controle kan doen. Proton geeft aan onze clubleden niet onbelangrijke kortingen op haar artikelen. Proton wil voor haar PC-computers komen tot de oprichting van een gebruikersgroep. Wie zich daarvoor wil aanmelden kan dat bij de redactie.

## INFORMATRONICA

Er is bij uitgeverij NANTON PRESS een nieuw electronica-maandblad verschenen, met de naam INFORMATRONICA (v/n ETI). Hierin wordt een 6502 zelfbouw-computer beschreven. Voortopis wordt behandeld: een basisprint, een RAM-kaart, een epronkaart, een I/O-kaart, een epronprogramma en een geluidskaart.

## 8-BIT CMOS MICROPROCESSORS dissipate 40mW at 2MHz

### R65C00 CMOS Microcomputer

The 8-bit R65C00 microcomputer system is produced with CMOS Silicon Gate technology. Advanced system architecture enhances its performance speeds; a family of software compatible microprocessor (CPU) devices enhances system cost-effectiveness. Rockwell also provides memory and microcomputer systems, as well as low-cost design aids and documentation.

### Instruction Set

- |  |         |  |
|--|---------|--|
| (1) new instruction                        | (2) JSR | Jump to New Location Saving Return Address   |
| (2) previous instruction                   | (2) LDA | Load Accumulator with Memory                 |
| with additional address-                   | LDX     | Load Index X with Memory                     |
| ing mode(s)                                | LDY     | Load Index Y with Memory                     |
|  | LSR     | Shift One Bit Right (Memory or Accumulator)  |
| (2) ADD                                    | NOP     | No Operation                                 |
| Add Memory to Accumulator with Carry       | (2) ORA | "OR" Memory with Accumulator                 |
| (2) AND                                    | PHA     | Push Accumulator on Stack                    |
| "AND" Memory with Accumulator              | PHP     | Push Processor Status on Stack               |
| ASL  | PHX     | Push X Register on Stack                     |
| Shift left One Bit (Memory or Accumulator) | (1) PHY | Push Y Register on Stack                     |
| (1) BBR                                    | PLA     | Pull Accumulator from Stack                  |
| Branch on Bit Reset                        | PLP     | Pull Processor Status from Stack             |
| (1) BBS                                    | PLX     | Pull X Register from Stack                   |
| Branch on Bit Set                          | (1) PLY | Pull Y Register from Stack                   |
| BCC  | (1) RMB | Reset Memory Bit                             |
| Branch on Carry Clear                      | ROL     | Rotate One Bit Left (Memory or Accumulator)  |
| BCS  | ROR     | Rotate One Bit Right (Memory or Accumulator) |
| Branch on Carry Set                        | RTI     | Return from Interrupt                        |
| BEG  | RTS     | Return from Subroutine                       |
| Branch on Result Zero                      | SBC     | Subtract Memory from Accumulator with Borrow |
| (2) BIT                                    | SEC     | Set Carry Flag                               |
| Test Bits in Memory with Accumulator       | SED     | Set Decimal Mode                             |
| BMI  | SEI     | Set Interrupt Disable Status                 |
| Branch on Result Minus                     | (1) SMB | Set Memory Bit                               |
| BNE  | (2) STA | Store Accumulator in Memory                  |
| Branch on Result not Zero                  | STX     | Store Index X in Memory                      |
| BPL  | STY     | Store Index Y in Memory                      |
| Branch on Result Plus                      | (1) STZ | Store Zero                                   |
| (1) BRA                                    | TAX     | Transfer Accumulator to Index X              |
| Branch Always                              | TRY     | Transfer Accumulator to Index Y              |
| BRK  | (1) TRB | Test and Reset Bits                          |
| Force Break                                | (1) TSB | Test and Set Bits                            |
| BVC  | TSX     | Transfer Stack Pointer to Index X            |
| Branch on Overflow Clear                   | TXA     | Transfer Index X to Accumulator              |
| BVS  | TXS     | Transfer Index X to Stack Register           |
| Branch on Overflow Set                     | TYA     | Transfer Index Y to Accumulator              |
| CLC  |         |  |
| Clear Carry Flag                           |         |  |
| CLD  |         |  |
| Clear Decimal Mode                         |         |  |
| CLI  |         |  |
| Clear Interrupt Disable Bit                |         |  |
| CLV  |         |  |
| Clear Overflow Flag                        |         |  |
| (2) CMP                                    |         |  |
| Compare Memory and Accumulator             |         |  |
| CPX  |         |  |
| Compare Memory and Index X                 |         |  |
| CPY  |         |  |
| Compare Memory and Index Y                 |         |  |
| (2) DEC                                    |         |  |
| Decrement Memory by One                    |         |  |
| DEX  |         |  |
| Decrement Index X by One                   |         |  |
| DEY  |         |  |
| Decrement Index Y by One                   |         |  |
| (2) EOR                                    |         |  |
| "Exclusive-OR" Memory with Accumulator     |         |  |
| (2) INC                                    |         |  |
| Increment Memory by One                    |         |  |
| INX  |         |  |
| Increment Index X by One                   |         |  |
| INY  |         |  |
| Increment Index Y by One                   |         |  |
| (2) JMP                                    |         |  |
| Jump to New Location                       |         |  |

## TINY (5.5K) BASIC

De heer J. Jansen, Gerardsweg 30, 6525 RT Nijmegen, telefoon: 088 - 562882, ontwikkelde een eigen BASIC. De Basic neemt de ruimte vanaf \$2000 tot \$3608 in, is 10 x sneller dan de Basic van de ZX 81 en 2 x sneller dan de Basic van de VIC-20. Statements: LET, PRINT, REM, WAIT INPUT, GET, BEEP, CLS, RND, ABS, GOSUB, RETURN, GOTO, POKE, PEEK, LINK, FOR NEXT, IF THEN, ON GOTO/GOSUB, en de commando's VARIABELE, DECIMAAL, HEXADECIMAAL, ESCAPE, RUN, LIST, TOP, FREE, CLEAR, NEW, CSAVE, CLOAD. Beginadres: \$2000 Startadres: \$3190 Startadres: \$306F (warm) Eindadr.+1: \$3608. Er is een 18 pagina's tellende manual bij geschreven. Prijs: Fl. 25,- incl. cassette, manual en portii. Voor informatie en bestellingen wende men zich tot de heer Jansen op bovenstaand adres.

## ROUTINE TO SAVE AND LOAD JUNIOR SYM-BASIC DATA TABLES.

Het schrijven naar en ophalen van cassetteband van BASIC programma's is voor bezitters van een uitgebreide JUNIOR of andere machine geen enkel probleem. De monitor I/O-routines verzorgen deze opdrachten. Het wordt anders als we Basic-data naar de band willen schrijven en er weer vanaf willen halen. Stel: U heeft een Basic-programma waarmee U een adresbestand kunt samenstellen. CLARE op de Basic-cassette nr. 2 is zo'n voorbeeld. Als U het programma laadt, is de lijst weer lees en moet alles weer opnieuw ingevoerd. Met behulp van Save and Load kunt U de ingevoerde adressen op de band bewaren. John M. Blalock publiceerde in Micro The 6502 Journal van april '80 een routine om het save en loaden van Basic Data Tables te verzorgen. We hebben deze machinetaalroutine aangepast voor JUNIOR met SYM-Basic en is, met twee demonstratieprogramma's in Basic te verkrijgen door Fl. 5,- over te schrijven op postrekening 841433 t.n.v. W.L. van Peit te Krampen a.d. IJssel.

Op het punt kwaliteit van DE 6502 KENNER wil ik nog iets aan U kwijt. De laatste tijd is daarover al het een en ander gezegd, en dat mag naar U toe een indicatie zijn dat het de waarneembare kentekens zijn van een bestuur en een redactie die dit onderwerp zeer ter harte gaat. Het gaat erbij niet meer over de vraag hoe een inzending zo snel mogelijk kan worden geplaatst, maar meer nog hoe maken we een zo verantwoord mogelijke keuze uit de totale voorraad artikelen, met behoud van het criterium om een voor publikatie in aanmerking komend artikel zo snel mogelijk op te nemen. Bedoelde keus wordt beïnvloed door behoeften aan software-artikelen voor een in personals zo gedifferentieerd lezerspubliek, bij voorkeur geardeerd met hardware-informatie en enige standaards voor bestuur, redactie, marktinformatie, e.d. Aan die kwaliteit kunnen ook de leden/lezers een niet geringe, om niet te zeggen een belangrijke bijdrage leveren door te zorgen dat in elk geval van alle in onze club gebruikte computers materiaal wordt ingezonden naar de redactie. Ik zeg met opzet "alle ... computer", omdat we daarmee een poging kunnen wagen de gedifferentieerdheid te laten weerspiegelen in Uw clubblad. Om het simpel te zeggen: laat geen zelf ontwikkelde software voor Uw OSI, PET, AIM-65, de PROTON-computers, APPLE, ACORN, KIM, BASIS 108, JUNIOR, etc. in Uw lade liggen. Het kan in Basic, Pascal, Forth, machinetaal, Micro-Ade zijn; het kan een hardware artikel zijn; de redactie blijft een onverzadigbare behoefte hebben aan Uw inzendingen, en de daarmee te ontwikkelen contacten. De eis dat een en ander behoorlijk becommentarieerd moet worden is nog steeds van kracht, dat wel. Er moet dus wel enige moeite worden gedaan om het een etiket "kwaliteit" toe te kennen.

Intussen hebben we op de redactie, door de grote stroom van enthousiaste inzendingen, waarbij zeer omvangrijke, wel de grootste moeite om een concept te vinden voor de combinatie om de informatie zo snel mogelijk door te laten stromen met

de leesbaarheids-kwaliteitseis. U hebt al gemerkt dat we al in zogenaamde "compressed"-mode, d.w.z. heel kleine letters, zijn gegaan. We zullen binnenkort zelfs over moeten gaan tot het in compressed-mode "op z'n kant" zetten van de pagina's. Voor sommigen onder ons kan dat enig bezwaar opleveren. Wij moeten echter begrip vragen voor dit kleine ongemak ten voordele van de doorstromingsnelheid en in onze poging elke inzending zo snel mogelijk te publiceren. We beschikken immers over slechts 48 pagina's per editie.

Ik heb uiteraard ook al snel bedacht dat je dan het aantal edities maar moet opvoeren. Dat is heel eenvoudig te realiseren door de contributie drastisch te verhogen. Echter, is het wel reeel in deze tijd een verdubbelde bijdrage te vragen. Immers, de allergrootste besrotingspost is - logisch - DE 6502 KENNER. Wil je het aantal edities verdubbelen, dan zul je evenredige contributieverhoging moeten vragen. Het probleem is echter veel groter dan het zich op het eerste gezicht presenteert. Het aantal inzendingen zal ook moeten verdubbelen. Maar hoe doe je dat met een gelijk blijvend of traag omhoog lopend aantal? Ik denk eerder dat de realiteit zit in de omgekeerde redenering: verdubbel het aantal leden en probeer daaruit het aantal inzendingen te halen die een toename van het aantal edities kunnen garanderen.

Mijn filosofie kan nog zo goed zijn, hij is sedoemd even snel als de andere sedachte te mislukken, als de leden niet bereid zijn mee te werken. Ik bedoel: van de leden zou ook gevraagd moeten worden meer werfkracht dan nu al het geval is op te brengen naast hun hobby-activiteiten. Ik heb eerder al eens gezegd: de club, dat bent U!

Het betekent ook dat we een krachtig bestuur met elkaar behoren op te bouwen. Dat is de medeverantwoordelijkheid van elk lid. Naar mijn smaak moet het uitbreiden van het ledenaantal met Uw hulp kunnen lukken. Geef gevolg aan de oproepen die bestuur en redactie doen, en de kans wordt groter dat U meer edities gaat ontvangen!

W.L. van Peit

## HCC - DAGEN

```

** De voorzitter van de Hobby Computer Club schreef onlangs dat hij **
** reeds lang met genoegen DE 6502 KENNER leest. Vandaar ook dat hem **
** in editie 26 de aankondiging van de HCC-Dagen op verkeerde dagen op- **
** gevallen is. Wij rectificeren uiteraard op zijn verzoek! **
**
** De gebruikelijke HCC-Dagen worden op vrijdag 21 en zaterdag 22 okto- **
** ber 1983 gehouden in de Jaarbeurs te Utrecht. **
** Onze club hoopt daar aanwezig te kunnen zijn met veel enthousiaste **
** leden, het liefst met hun systeem of andere te demonstreren 6502 - **
** zaken. **
**
** Heeft U belangstelling de club behulpzaam te zijn bij het opbouwen **
** van de stand en/of het representeren van de club aldaar, neem even **
** contact op met een van de bestuursleden of de redactie. **

```

PROGRAMMEREN MET FOCAL

FOCAL is een programmeertaal die in veel opzichten hetzelfde is als BASIC, maar in een aantal gevallen is FOCAL krachtiger. Dit wordt geen volledige handleiding, statements als LET, PRINT INPUT, IF THEN, FOR NEXT enz in BASIC zijn in FOCAL ook terug te vinden. Daarom behandel ik alleen een paar interessante(re) statements.

Werken we in BASIC met regelnummers, in FOCAL werken we met groepen (max. 99). Elk verschillend onderdeel van een programma komt in een eigen groep. Binnen zo'n groep kan men 99 regels gebruiken. Zo betekent 10.30 groep 10 30ste regel.

FOCAL bezit 'gewone' statements en 'speciale' statements.

De gewone statements (IF, GOTO, FOR, RETURN, DO, SET enz) mogen tot hun eerste letter worden afgekort en werken (ongeveer) op dezelfde manier als in BASIC.

De speciale statements zijn FUNCTIES. Elke naam v/e FUNCTIE begint met een 'F' (bv. FCAL, FIDV enz). Het zijn deze FUNCTIES die interessant zijn.

FOCAL werkt (tenminste mijn uitvoering) met 8 getallen nauwkeurigheid, maar kent geen wetenschappelijke notatie.

33333.34	Bij het uitprinten van getallen kan men zelf
1.00	bepalen hoeveel getallen voor en achter de
-102.51	decimale punt moeten worden afgedrukt. Door
0.01	deze mogelijkheid komt een reeks van getallen
22210.00	altijd mooi met de decimale punt onder elkaar.

In FOCAL is het mogelijk om op meerdere manieren een subroutine aan te roepen. Enkele voorbeelden:

- dmv DO 3.1 wordt een regel (hier 3.1) als subr. uitgevoerd
- dmv DO 3 wordt een groep (hier 3) als subr. uitgevoerd
- dmv DO A\$ wordt een string als subr. uitgevoerd
- dmv FCAL wordt een machinetaal subr. uitgevoerd
- dmv FSBR wordt een regel of groep als subr. uitgevoerd

FSBR heeft als eigenschap dat men een variabele mag meegeven naar een subr. met die waarde wordt alles berekend en de uitkomst komt weer in die variabele terug. Zeer handig als men met verschillende variabelen dezelfde berekeningen wilt uitvoeren.

HET GEBRUIK VAN STRINGS:

Normaal wordt een toetsenbord gebruikt als inputorgaan (input device) en het beeldscherm als outputorgaan (output device). Dit is in FOCAL softwarematig te veranderen zodat men bv. naar een printer, cassetterecorder, floppydisc enz. kan overschakelen. Het krachtige is echter dat men deze hardware zaken ook kan vervangen door software (nl. dmv. STRINGS).

Het is mogelijk om het toetsenbord te vervangen door een string alles wat in die string staat accepteert FOCAL als of het door de programmeur is ingetypt. Zo kan men kommando's, statements en zelfs gehele regels, die in een STRING staan, laten uitvoeren. Ook is het mogelijk het outputorgaan (beeldscherm) te vervangen door een string (oid) nu worden alle terugmeldingen in die string (oid) opgeborgen.

Het vullen v/e string kan dmv:

- a. een String Input (FSTI(x,A\$(y),z)) waarbij x het max. aantal karakters aangeeft, y de plaats waarin A\$ wordt begonnen en z met welke karakter geëindigd kan worden (indien nog geen x karakters zijn ingevoerd).
- b. A\$(10)='Q' nu wordt de 11ste plaats van A\$ met een 'Q' gevult.
- c. A\$ als output device te verklaren en de data die in de string moet komen gewoon te laten uitprinten.

Ik zal een voorbeeld geven van c:

Dmv SET FODV(A\$) wordt het output device (was beeldscherm) nu A\$ TYPE (of T) doet hetzelfde als PRINT in BASIC en hiermee kun je nu A\$ vullen (A\$ vervangt immers het beeldscherm, waar het anders op zou komen). RESTORE OUTPUT (of R O) herstelt het outputorgaan. SET FIDV(A\$) zal het inputorgaan (is meestal het toetsenbord) op A\$ zetten. Dus alles wat in A\$ staat lijkt voor FOCAL als of het gewoon is ingetypt. Een '!' bij een TYPE statement print een carriage return (= 'cr').

QUIT is hetzelfde als END in BASIC, er wordt gestopt met het programma.

Nu volgt er een programma dat eerst A\$ met een regel FOCAL vult en daarna die regel bij het programma toevoegt.

- 1.1 SET FODV(A\$)                    maak A\$ output voor het vullen van A\$  
1.2 TYPE "2.1 TYPE X\*3",!,"R I",!  
1.3 RESTORE OUTPUT                herstel het outputorgaan (evt. R O)  
1.4 SET FIDV(A\$)                    vervang het toetsenbord door A\$  
1.5 QUIT                            stop en kijk naar het input device

wat er gedaan moet worden.

Dmv 1.1, 1.2 en 1.3 wordt A\$ geladen met:

2.1 TYPE X\*3

R I

Regel 1.4 zet het input device op A\$ en 1.5 stopt het programma. Nu wordt er naar het input device (is nu dus A\$) gekeken, daarin staat een regel FOCAL (+ een 'cr') deze wordt er dus bij geschreven. Nu wordt er weer verder gekeken in A\$ en daarin staat R I (=RESTORE INPUT of wel herstel input) en het toetsenbord wordt inputorgaan. Er is dus nu regel 2.1 bij het programma gevoegd. Er bestaat dus een mogelijkheid om een FOCAL programma te maken dat zelf weer een FOCAL programma schrijft. Of een FOCAL programma dat een regel BASIC in een string opslaat en die string vervolgens gaat vertalen naar FOCAL en daarna die string (wat nu geheel FOCAL is) wegschrijft in het geheugen. Een soort BASIC naar FOCAL vertaler, geschreven in FOCAL, krijgt men dan.

#### HET WERKEN MET INTERRUPTS:

Het is, in FOCAL, mogelijk om mbv. een interrupt een FOCAL programma te beïnvloeden. Dmv. het 'setten' van bitjes en een statement in FOCAL kan men dan nog enkele interrupts prioriteit geven. Er wordt dan door FOCAL zelf gekeken of er een interrupt plaats vindt en wat er daarna gedaan moet worden waarbij de interrupt met het hoogste prioriteit als eerste wordt uitgevoerd. (zelf heb ik hiermee nog niet geëxperimenteerd).

J Janssen

Gerardsweg 30

65 25 RT NIJMEGEN

tel.:080-562082

## C64-Centronics printer interface

### Source listing print routine

```

0010          .BA $CF00
0020 : C64-CENTRONICS PRINT ROUTINE
0030 :
0040 : COPYRIGHT (C) 1983
0050 :
0060 : ANTON MULLER
0070 : SINJELR SEMEYNSSTRAAT 78 1
0080 : 1061 GM AMSTERDAM
0090 :
0100 CIAIDRA .DE $DD00      : I/O REGISTER A
0110 CIAIDRB .DE $DD01      : I/O REGISTER B
0120 CIADRA  .DE $DD02      : DATA DIR REG A
0130 CIADRB  .DE $DD03      : DATA DIR REG B
0140 LF      .DE $0A        : LINEFEED
0150 CR      .DE $0D        : CARRIAGE RETURN
0160 :
CF00- A9 7F 0170 INIT     LDA #$7F      : INIT PORT B
CF02- 8D 03 DD 0180     STA CIADRB
CF05- AD 02 DD 0190     LDA CIADRA      : PA2 = OUTPUT (STROBE)
CF08- 09 04 0200     ORA #4
CF0A- 8D 02 DD 0210     STA CIADRA
CF0D- A9 18 0220     LDA #L.PRINT    : DUTCH VECTOR
CF0F- 8D 25 03 0230     STA $0325      : POINTS NOW
CF12- A9 CF 0240     LDA #H.PRINT    : TO THE CENTRONICS
CF14- 8D 27 03 0250     STA $0327      : PRINT ROUTINE
CF17- 60 0260     RTS          : RETURN TO CALLER
0270 :
CF18- 48 0280 PRINT     PHA          : SAVE CHAR TO BE PRINTED
CF19- 2C 01 DD 0290 PRINT010 BIT $DD01    : REPEAT READ BUSY LINE
CF1C- 36 FB 0300     BMI PRINT010    : UNTIL PRINTER NOT BUSY
CF1E- 58 0310     PLA
CF1F- 48 0320     PHA          : RELOAD AND STORE CHAR
CF20- C9 0D 0330     CMP #CR      : IF CARRIAGE RETURN THEN
CF22- D0 02 0340     BNE PRINT020
CF24- A9 0A 0350     LDA #LF      : CHANGE IT INTO A LINEFEED
CF26- 8D 01 DD 0360 PRINT020 STA CIAIDRB  : STORE CHAR INTO I/O REG B
CF29- AD 00 DD 0370     LDA CIAIDRA  : GIVE STROBE PULSE
CF2C- 29 FB 0380     AND #$FB     : LOW TO HIGH TRANSITION
CF2E- 8D 00 DD 0390     STA CIAIDRA
CF31- AD 00 DD 0400     LDA CIAIDRA
CF34- 09 04 0410     ORA #4
CF36- 8D 00 DD 0420     STA CIAIDRA
CF39- 68 0430     PLA          : RESTORE CHAR
CF3A- 4C CA F1 0440     JMP $F10A    : GOTO C64 OUTPUT ROUTINE
0450 :
CF3D- A9 CA 0450 CLOSE  LDA #L,$F10A  : RESTORE DUTCH VECTOR
CF3F- 8D 26 03 0470     STA $0325
CF42- A9 F1 0480     LDA #H,$F10A
CF44- 8D 27 03 0490     STA $0327
CF47- 60 0500     RTS          : RETURN TO CALLER

```

```

0510 ;
CF48- 20 00 CF 0520 MONINIT JSR INIT ; ENTRY POINT FOR SUPERMON-C64
CF4B- 00 0530 BRK ; RETURN TO SUPERMON-C64
0540 ;
CF4C- 20 3D CF 0550 MONCLOSE JSR CLOSE ; EXIT POINT FOR SUPERMON-C64
CF4F- 00 0560 BRK ; RETURN TO SUPERMON-C64
0570 ;
CF50- A9 1D 0580 NARROW LDA #1D ; PRINT 16.5 CHARACTERS PER INCH
CF52- 20 18 CF 0590 JSR PRINT ; ON THE OKI MICROLINE 80
CF55- 60 0600 RTS ; RETURN TO CALLER
0610 ;
CF56- A9 1E 0620 NORMAL LDA #1E ; PRINT 10 CHARACTERS PER INCH
CF58- 20 18 CF 0630 JSR PRINT ; ON THE OKI MICROLINE 80
CF5B- 60 0640 RTS ; RETURN TO CALLER
0650 ;
CF5C- A9 1F 0660 WIDE LDA #1F ; PRINT 5 CHARACTERS PER INCH
CF5E- 20 18 CF 0670 JSR PRINT ; ON THE OKI MICROLINE 80
CF61- 60 0680 RTS ; RETURN TO CALLER
0690 ;
CF62- 20 50 CF 0700 MONNARROW JSR NARROW ; SUPERMON-C64 NARROW PRINT INIT
CF65- 00 0710 BRK ; RETURN TO SUPERMON-C64
0720 ;
CF66- 20 56 CF 0730 MONNORMAL JSR NORMAL ; SUPERMON-C64 NORMAL PRINT INIT
CF69- 00 0740 BRK ; RETURN TO SUPERMON-C64
0750 ;
CF6A- 20 50 CF 0760 MONWIDE JSR WIDE ; SUPERMON-C64 WIDE PRINT INIT
CF6D- 00 0770 BRK ; RETURN TO SUPERMON-C64
0780 .EN

```

LABEL FILE: ( \* = EXTERNAL )

```

DD00 = *CIAIDRA      DD01 = *CIAIDRB      DD02 = *CIAIDRA
DD03 = *CIAIDRB      DD0A = *LF           DD0D = *CR
CF00 = INIT          CF18 = PRINT        CF19 = PRINT010
CF26 = PRINT020      CF3D = CLOSE         CF48 = MONINIT
CF4C = MONCLOSE      CF50 = NARROW        CF56 = NORMAL
CF5C = WIDE          CF62 = MONNARROW    CF66 = MONNORMAL
CF6A = MONWIDE

```

0000, CF6E, CF6E

Aan de user poort van de Commodore-64 wordt een 24 polige edge connector aangesloten. De pennen zijn genummerd van 1 t/m 12 aan de bovenkant en A t/m N aan de onderkant (de letters G en I worden niet gebruikt, om verwarring met de 6 en de 1 te voorkomen).

Aan de printer kant wordt een 36 polige centronics 'male' connector aangesloten. De pennen zijn genummerd van 1 t/m 36.

Commodore-64 user port	Centronics connector
+-----+	+-----+
! PEN A = GND !<----->!	PEN16 = GND !
+-----+	+-----+
! PEN C = PB0 !----->!	PEN 2 = DATA 1 !
+-----+	+-----+
! PEN D = PB1 !----->!	PEN 3 = DATA 2 !
+-----+	+-----+
! PEN E = PB2 !----->!	PEN 4 = DATA 3 !
+-----+	+-----+
! PEN F = PB3 !----->!	PEN 5 = DATA 4 !
+-----+	+-----+
! PEN H = PB4 !----->!	PEN 6 = DATA 5 !
+-----+	+-----+
! PEN J = PB5 !----->!	PEN 7 = DATA 6 !
+-----+	+-----+
! PEN K = PB6 !----->!	PEN 8 = DATA 7 !
+-----+	+-----+
! PEN L = PB7 !<-----!	PEN11 = BUSY !
+-----+	+-----+
! PEN M = PA2 !----->!	PEN 1 = STROBE !
+-----+	+-----+

Maak de kabel niet te lang, hooguit twee meter. Mocht de kabel toch langer moeten, gebruik dan twisted pairs.

CF00 Init subroutine

CF19 Print subroutine

CF3D Close subroutine

CF48 Init routine Supermon-C64

CF4C Close routine Supermon-C64

CF50 Subroutine set print mode narrow 16.5 characters per inch

CF56 Subroutine set print mode normal 10 characters per inch

CF5C Subroutine set print mode wide 5 characters per inch

CF62 Set print mode narrow 16.5 characters per inch for Supermon-C64

CF66 Set print mode normal 10 characters per inch for Supermon-C64

CF6A Set print mode wide 5 characters per inch for Supermon-C64

## NOGMAALS: DE BASIC-COMPRESSOR

Rond mijn programma BASIC COMPRESSOR uit 6502-kenner 25 zijn een paar vragen gerezen:

### 1. Waar dient het ding voor ?

Antwoord: Om fatsoenlijke BASIC programma's te kunnen schrijven die, behalve dat ze werken, ook nog door anderen te lezen en te begrijpen zijn. Dus inplaats van een spaghetti sliert als b.v.

```
100 FOR A=0 TO 100:PRINT A::PRINT A*A::PRINT SQR(A):NEXT
```

Schrijven we voortaan:

```
100 FOR A=0 TO 100           :REM VOOR DE GETALLEN VAN 0 TOT 100
110 : PRINT A               :REM PRINT GETAL
120 : PRINT A*A             :REM PRINT HET KWADRAAT
130 : PRINT SQR(A)         :REM PRINT DE WORTEL EN GA NAAR EEN NIEUWE REGEL
140 NEXT                   :REM VOLGEND GETAL
150 END                     :REM EINDE
```

De vele spaties en rem-statements zijn een MUST ! Alleen zo is een programma voor iedereen verstaanbaar. We praten dus helemaal niet meer over geheugenruimte en executietijd. Daar zorgt straks, na het afdrucken van de listing de BASIC COMPRESSOR wel voor. Overigens: Dok regels vol met cursor besturingskarakters zoals de meeste COMMODORE eigenaars zo heerlijk weten te produceren zijn uit den boze. Gebruik voor cursor right liever TAB. Cursor 5 stappen terug ? Gebruik dan niet een soort statement als:

```
250 PRINT "IIIII";
```

Maar gebruik:

```
250 LE%=CHR$(157)           :REM CURSOR NAAR LINKS
260 FOR LE=1 TO 5          :REM VOOR 5 STAPPEN TERUG
270 : PRINT LE%           :REM STUUR CURSOR NAAR LINKS
280 NEXT                   :REM VOLGENDE STAP
```

### 2. Het programma werkt niet op JUNIOR en KIM

Antwoord: Nee, maar daar doen we nu iets aan. Per abuis staat by de versie voor SYM-BASIC, aangepast voor de Junior, dat deze versie ook op KIM en JUNIOR zou werken. Pas de volgende adressen aan op onderstaande regelnummers:

REGNR/LABEL	KIM BASIC V1.0	JUNIOR KBS	VIC-20	COMMODORE 64
1220	STRT \$7B	\$7B	\$2B	\$2B
1230	ENDP \$7A	\$7D	\$2D	\$2D
1240	ENDP1 \$7A	\$7D	\$2D	\$2D
1250	BASIC \$2348	\$227B	\$C474	\$A474

Zoals de VIC en COMMODORE 64 bezitters zien, werkt het programma ook op hun systeem. Voor de COMMODORE is echter nog een aanpassing noodzakelijk: De fabrikant is er in geslaagd zeropage voor I/O te gebruiken, zodat alle zeropage adressen met 3 moeten worden verhoogd. De statement "token's" zijn voor beide laatste machines gelijk aan die voor PET en CBM.

Ruud Uphoff

## USR ROUTINES

In Microsoft BASIC is het mogelijk zelf nieuwe functies te schrijven in machinetaal en deze door het initialiseren van een vector, de USR-vector, aan de basicinterpreter te hangen. Vanaf dat moment is zo'n functie dan tot onze beschikking. In de loop van de tijd heb ik heel wat van deze routines geschreven die U mogelijk ook wel wilt gebruiken. Alle routines zijn geassembleerd op de APPLE maar de adressen voor een aantal bekende basicinterpreters kan ik U er steeds bij geven. Dat is dan mogelijk voor:

KIM BASIC V1.0 / JUNIOR KB9 / VIC-20 / COMMODORE-64 / PET / CBM-3032

Laat ik beginnen met te waarschuwen voor de KIM BASIC omdat die geen USR-vector bezit, maar waarbij het adres van de USR routine in de tabel van functie-adressen moet worden veranderd. Verder mag ik er vanuit gaan dat degene die zelf patches in zijn BASIC heeft aangebracht, ook wel in staat zal zijn daar bij het aanpassen van de routines rekening mee te houden. Deze keer wil ik beginnen met een routine die U in staat stelt op een INPUT-statement response te geven door het intypen van een expressie, iets wat normaal niet mogelijk is. Voor de KIM moeten de regels 0370 en 0380 gevuld worden met 5 NOP instructies!

### De syntax.

Na het schrijven van de routine, wordt de functie als volgt gebruikt:

```

100 LET INIT=0000           :REM HET DOOR U GEASSEMBLEERDE STARTADRES
110 INPUT "EXPRESSIE ";X$   :REM INPUT EXPRESSIE
120 IF LEN(X$)>79 THEN 110   :REM TEST OP MAX 79 KARAKTERS, NIET NOODIG VOOR APPLE
130 CALL INIT               :REM SYS INIT VOOR COMMODORE
140 X=USR(X$)               :REM EVALUEER EXPRESSIE VAN X$
150 PRINT X                 :REM PRINT RESULTAAT
160 END

```

Op sommige systemen zal de initialiserings routine moeten vervallen omdat het commando "CALL" of "SYS" niet bestaat. U kunt echter ook dit commando in uw BASIC versie aanbrensen, want het is onmisbaar. Zonder CALL moet U ook af zien van een selectie op al of niet reeds gecodeerd zijn van de string. Zet in dat geval vijf NOP's op de regels 540 en 550. De routine wordt dan wel traag, als de string meerdere malen moet worden gevalueerd!

Om CALL in de basicinterpreter te brengen kunt U het vrijwel nutteloze commando NULL verwijderen, tenzij U nog met een "teletype" werkt. Breng de navolgende wijzigingen aan: (opcode notatie)

KIM	JUNIOR	
===	=====	
\$20C7: 43 41	\$20C7: 43 41	Vervang NU door CA in de tabel
\$2728: EA	\$26EE: EA	Vervang NULL routine door CALL
\$272C: EA	\$26EF: EA	
\$272D: EA	\$2670: EA	
\$272E: EA	\$2671: EA	
\$272F: EA	\$2672: EA	
\$2730: 20 34 2C	\$2673: 20 71 2B	CALL: Haal adres in FLP-accu
\$2733: 20 EF 35	\$2676: 20 38 35	Converteer naar adres in vector
\$2736: 6C 19 00	\$2679: 6C 1C 00	Spring er naartoe

Van gebruik van "conditional assembling" wordt hier afgezien, omdat dat erg veel plaatsruimte in de

6502-KENNER zou innemen. Pas de adressen van de onderstaande labels, per systeem als volgt aan:

LABEL	PET	CBM	VIC	C-64	KIM	JUN
====	===	===	===	====	===	===
TXTP1	00C9	0077	007A	007A	00C7	00CA
TXTPH	00CA	0078	007B	007B	00C8	00CB
FLPAC	00B0	005E	0061	0061	00AE	00B1
TMPOL	0071	001F	0022	0022	006F	0072
USRVJ	0000	0000	0000	0310	---	000A
USRVL	0001	0001	0001	0311	2040	000B
USRVH	0002	0002	0002	0312	2041	000C
CHARGET	00C2	0070	0073	0073	00C0	00C3
DLOCSTR	D57B	D57D	D6A3	B6A3	349A	33E3
ENCODE	C48D	C495	C579	A579	2466	23C9
EVALEX	CCB8	CC9F	CD9E	AD9E	2C48	2B85
NONDIR	D28B	D280	D3A6	B3A6	31A8	30EA
BUFFER	000A	0200	0200	0200	001B	001E

Het voert hier te ver om te gaan vertellen wat elke routine nu precies doet, maar in de zeer nabije toekomst is er een lijst beschikbaar met bruikbare routines, wat ze doen en hoe ze moeten worden aangeroepen. Meer nieuws in een van de volgende nummers.

Tenslotte een praktische toepassing van de USR-routine. Stel dat we een BASIC programma willen schrijven dat van een ingevoerde reeks getallen een zekere functie  $y=f(x)$  afdruckt. Het invoeren van beginwaarde, eindwaarde en stap is geen probleem. Wel de functie zelf. We kunnen deze vastleggen in een DEF statement maar dan moeten we voor elke nieuwe functie dat DEF statement in het programma veranderen. Het bijgaande BASIC programma laat U nog iets zien: Gestructureerd programmeren is in basic niet onmogelijk. De afdruck werd gemaakt met een speciale "list" routine voor de APPLE, dat in een gestructureerd programma de structuren herkent. Helaas kan ik, voorlopig alleen APPLE bezitters een copie op disk aanbieden (dos 3.3 of 3.2) Stuur mij een blanco diskette met retour porto en heb een weekje of twee geduld.

Ruud Uphoff

### Uitnodiging KIM-Club Bijeenkomst.

Datum : zaterdag 17 september 1983  
 Lokatie : Chr. Atheneum "Adriaen Pauw"  
 Ir. Lelylaan 18, HEEMSTEDE. Tel.: 023 - 281702

#### Reisroute :

- per auto: snelweg Den Haag - A'dam of A'dam - Den Haag;  
 afslag HOOFDDORP.

Richting (Hoofddorp) HEEMSTEDE.

Na de brug over de ringvaart Haarlemmermeer  
 (stoplicht) rechtdoor.

Volgende stoplicht OMKEREN en terug OVER DE  
 PARALLELWEG

Aan het einde van deze weg:  
 CHR. ATHENEUM "ADRIAEN PAUW".

\*\*\* NEEM EEN CLUBGENDOT OF KENNIS MEE \*\*\*  
 \*\*\* BRENG OOK UW APPARATUUR MEE \*\*\*

Toesang : incl. koffie, lunch en frisdrankje : Fl. 25,==

### Programma KIM Club Bijeenkomst Heemstede

- 09.30 Onvanst met koffie
- 10.00 Openings door de voorzitter
- 10.15 Bob van de Oudewetering demonstreert  
supersnelle Floppy Disk Operating  
System
- 11.15 Koffiepauze
- 11.45 Voortzetting demo FDOS (mogelijk met  
koppeling aan APPLE)
- 12.15 Lunch
- 13.30 Consumentenvoorlichting over floppy's  
door Rinus Vleesch Dubois (voorzitter)
- 14.30 Markt
- 14.45 Theepauze
- 15.15. Informeel gedeelte
- 17.00 Sluiting

```

0010      .LS
0020      ;
0030      ;*****
0040      ;*
0050      ;* UPHOFF'S USR ROUTINE LIBRARY *
0060      ;*
0070      ;* NR.1 "INPUT AN EXPRESSION" *
0080      ;*
0090      ;*****
0100      ;
0110      ;=== ASSEMBLED FOR APPLE II ===
0120      ;
0130      ;MICROSOFT PAGE ZERO:
0140      ;
0150      .DE $00B8      ;LOW BYTE OF PTR INTO PROGRAM
0160      .DE $00B9      ;HIGH BYTE OF SAME
0170      .DE $009D      ;FLP-ACCU (EXPONENT)
0180      .DE $005E      ;TEMPORARY PTR (LOW)
0190      .DE $000A      ;JMP BEFORE USR VECTOR
0200      .DE $000B      ;LOW BYTE OF USR VECTOR
0210      .DE $000C      ;HIGH BYTE OF SAME
0220      ;
0230      ;MICROSOFT SUBROUTINES
0240      ;
0250      .DE $00B1      ;GET NEXT CHARACTER FROM STATEMENT
0260      .DE $E5FD      ;DEALLOCATE TEMPORARY STRING
0270      .DE $D559      ;ENCODE KEYWORDS IN STATEMENT
0280      .DE $DD7B      ;EVALUATE EXPRESSION
0290      .DE $E306      ;CHECK: NO DIRECT MODE
0300      .DE $0200      ;INPUT BUFFER
0310      ;
0320      .BA $0300
0330      .OS
0340      ;
0350      ;INITIALIZE BY CALL 768
0360      ;
0370      LDA #$4C      ;GET OPCODE 'JMP' (3 NOP'S FOR KIM)
0380      STA USRVJ      ;PUT BEFORE USR VECTOR (3 NOP'S FOR KIM)
0390      LDA #L,INPEXP ;GET LOW BYTE FROM START ADDRESS
0400      STA USRVL      ;PUT INTO LOW BYTE FROM USR VECTOR
0410      LDA #H,INPEXP ;GET HIGH BYTE FROM START ADDRESS
0420      STA USRVH      ;PUT INTO HIGH BYTE FROM USR VECTOR
0430      LDA #$80      ;GET ANY NEGATIVE VALUE
0440      STA FLAG      ;SET FLAG 'FIRST ENTRY'
0450      RTS          ;RETURN TO BASIC PROGRAM
0460      ;
0470      ;USR ROUTINE "INPUT AN EXPRESSION"
0480      ;
0490      JSR NONDIR    ;CHECK FOR NON DIRECT MODE
0500      LDA *TXTPL    ;SAVE TEXT POINTER
0510      PHA
0520      LDA *TXTPH    ;
0530      PHA
0540      BIT FLAG      ;CHECK TYPE OF ENTRY
0550      BPL ENCODED   ;IF FIRST CALL
0560      JSR ENCODER   ; ENCODE EXPRESSION IN BUFFER
0570      JSR GETPTR   ;INITIALIZE TEXT POINTER

```

0329- 20 B1 00	0580	JSR CHARGET	:GET FIRST CHARACTER
032C- 20 7B DD	0590	JSR EVALEX	:EVALUATE EXPRESSION
032F- 68	0600	PLA	:RESTORE ORIGINAL TEXT POINTER
0330- 85 B9	0610	STA *TXTPH	
0332- 68	0620	PLA	
0333- 85 B8	0630	STA *TXTPL	
0335- 68	0640	PLA	:REMOVE OWN RETURN ADDRESS
0336- 68	0650	PLA	
0337- 60	0660	RTS	:AND RETURN WITHOUT CHECK ON STRING FLAG
	0670	:	
	0680	:	:ENCODE EXPRESSION IN BUFFER
	0690	:	
0338- 20 FD E5	0700	JSR DLOCSTR	:GET PTR TO STRING, LENGTH IN A
033B- 85 9D	0710	STA *FLPAC	:STORE LENGTH
033D- A0 FF	0720	LDY #FF	:INITIALIZE INDEX
033F- C8	0730	INY	:WHILE MORE CHARACTERS
0340- C4 9D	0740	CPY *FLPAC	: DO:
0342- B0 08	0750	BCS ENDSTR	
0344- B1 5E	0760	LDA (TMPOL),Y	: GET CHARACTER
0346- 99 00 02	0770	STA BUFFER,Y	: COPY INTO BUFFER
0349- 4C 3F 03	0780	JMP COPY	:ENDWHILE
034C- A9 00	0790	LDA #00	:PUT 'END OF LINE'
034E- 99 00 02	0800	STA BUFFER,Y	:BEYOND STRING
0351- 20 5E 03	0810	JSR GETPTR	:GET POINTER TO BUFFER
0354- 20 B1 00	0820	JSR CHARGET	:READ FIRST CHARACTER
0357- 20 59 05	0830	JSR ENCODE	:ENCODE KEYWORDS IN STATEMENT
035A- 4E 67 03	0840	LSR FLAG	:SET 'FIRST CALL DONE'
035D- 60	0850	RTS	:RETURN
	0860	:	
	0870	:	:GET POINTER TO BUFFER
	0880	:	
035E- A9 FF	0890	LDA #L,BUFFER-1	:GET ADDRESS IN Y/A
0360- A0 01	0900	LDY #H,BUFFER-1	
0362- 85 B8	0910	STA *TXTPL	:STORE INTO TEXT POINTER
0364- 84 B9	0920	STY *TXTPH	
0366- 60	0930	RTS	:AND RETURN
	0940	:	
	0950	:	:INTERNAL VARIABLE
	0960	:	
0367-	0970	.DS 1	:1 BYTE FOR ENTRY FLAG
	0980	:	
//0000,0368,0368			



# DE 6502 KENNER

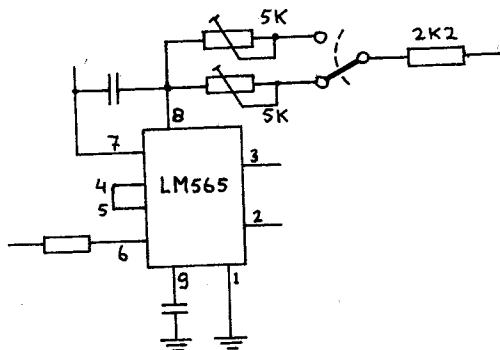
SUGGESTIES T.A.V. BASICODE-PUBLIKATIE IN EDITIE 23.

W. VAN GOOR, SOPHIASTRAAT 21, 2641 HL PIJNACKER.

DE DOORSPRONKELIJKE CASSETTE-HARDWARE VAN DE KIM KUN JE AANPASSEN AAN DE BASICODE-NORM VAN:

1200 Hz => 0  
2400 Hz => 1

ALS JE DE POTMETER VERDRAAIT, KUN JE HET SCHAKELPUNT VAN 3000 Hz IN DE KIM VERLEGGEN NAAR 1800 Hz. DE PLL VAN DE KIM LAAT NAMELIJK EEN BETERE INLEZING VAN BANDEN TOE DAN DE D-COMPARATOR VAN SC/MP (ervaringen van Exidy Sorcerer-mensen!). UITERAARD MOET HET PROGRAMMA DAN HERSCHREVEN WORDEN OP DE INPUT PIA VAN DE KIM CASSETTE-ROUTINE, DUS EEN NIEUW PROGRAMMA DAT GEBRUIK MAAKT VAN DE ORIGINELE HARDWARE. JE KUNT DE POTMETER OOK MET EEN SCHAKELAAR UITVOEREN, DAN HEB JE TWEE STANDEN: BASICODE/KIM.



JE KUNT DE POTMETER AFREGELLEN MET BEHULP VAN EEN TEST-BAND WAAROP 1900 Hz STAAT, DIT MOET DAN NET EEN "1" OPLEVEREN AAN DE PLL TESTUITGANG (X VAN EXPANSIE-CONNECTOR).

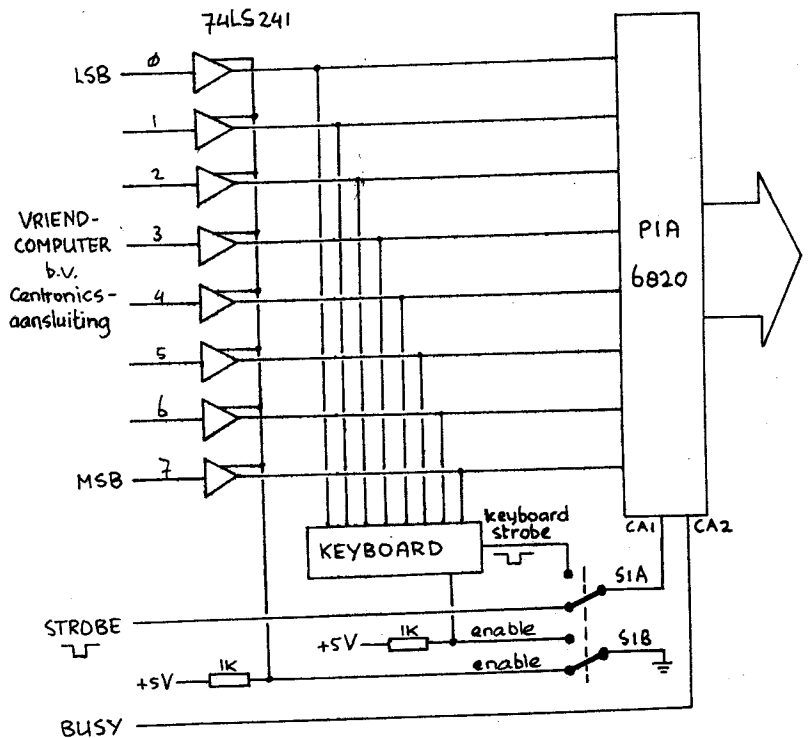
WIL JE GRAAG EEN BASICOPROGRAMMA HEBBEN VAN JE VRIEND, MAAR HEB JE GEEN BASICODE EN JE VRIEND EEN ANDER TYPE COMPUTER? GEEN MOED!

MAAK EEN VERBINDING VAN EEN OUTPUT-POORT VAN DE VRIEND-COMPUTER NAAR JE PIA (ZIE SCHEMA). JE VRIEND-COMPUTER DIENST ALS AUTOMATISCH KEYBOARD!!

WERKING: VRIEND-COMPUTER LAADT ZIJN BASICOPROGRAMMA EN 'LIST' HET UIT VIA ZIJN PRINTEROUTPUTROUTINE (BIJV. CENTRONIC). JIJ ONTVANGT HET OP JOUW KEYBOARD INPUT, WELKE WEL EEN PARALLEL-TYPE DIENST TE ZIJN.

## Basicode op band/papier

Het programma Basicode van Jaap van Toledo is gepubliceerd in editie 23. Het is in Micro-Ade formaat en in code beschikbaar op de Junior-cassette nr. 2 welke werd gepubliceerd in editie nr. 24. De listing van het programma is ook verkrijgbaar op papier. Junior-cassette nr.2 F.12,50 Listing Basicode : F. 5,00 Het betreffende bedrag overschrijven op postrekening nr 841433 t.n.v. W.I. van Peit te Krimpen a.d. IJssel, met vermelding van hetseem wordt verlanad.



;Van Nieuwenhove Koen

;\*\*\*\*\* KOPPELING VAN JUNIOR AAN 8" FLOPPY \*\*\*\*\*

-----

;Deze tekst beschrijft de koppeling van een 8 inch floppy disk drive aan de Junior. De beschrijving bestaat uit twee gedeeltes. Eerst is er de hardware. Deze bestaat uit een controller op basis van een FD1771. Deze controller werd gebouwd op eurokaart-formaat. Hij laat toe in 8" single density (of 5" double density) te werken. Tevens is in de mogelijkheid voorzien om met twee drives te werken. De controllerkaart kan op de busprint van de Junior gestoken worden. Ze bevindt zich op de geheugenplaatsen : \$1900-\$1904. Dit is mogelijk door enkele wijzigingen aan de interfacekaart aan te brengen.

;De software bestaat uit drie gedeeltes. Een eerste gedeelte verzorgt het datatransport tussen computer en drive op het laagste niveau: lezen en schrijven van sectoren en tracks, evenals het formatteren van de disks. De hierboven genoemde hard- en software is grotendeels gebaseerd op een artikel verschenen in BYTE van mei en juni 1981.

;Het tweede gedeelte van de software is een disk operating system gebaseerd op het boek 'A floppy disk operating system for the 8080' van Kenneth B. Welles. Dit boek wordt uitgegeven door Byte Books. Het hierin beschreven operating system heb ik vrij omgezet naar 6502. Het biedt zeer ruime mogelijkheden voor de gebruiker. Meer daarover verder.

;Het derde gedeelte is een programma met utilities ten behoeve van het operating system. De totale software neemt ongeveer 6K geheugen in beslag. Deze mag zich zowel in ram als in eprom bevinden. Tevens maakt deze software gebruik van de geheugenruimte van \$200 tot \$600. Dit alles biedt het voordeel dat bestaande soft als cassettebesturing en vooral assembler, basic en vroeger ontwikkelde programma's bruikbaar blijven. Bij mij draait de assembler van Moser al met het DOS. Ik ben bezig patches te maken voor de Basic. In het vooruitzicht is ook nog Forth en Pascal.

; HARDWARE.

-----

;De volledige hardware omvat een twintigtal TTL-ic, een FD1771 en een 6821. In bijlage vindt U een copy van een artikel uit byte dat een beschrijving geeft van het grootste gedeelte van de werking. In bijlage vindt U tevens een schema van de hardware zoals die bij mij werkt. Er zijn geen essentiële verschillen. Er zijn wel enkele dingen aan toe gevoegd. Dit zowel naar de computer toe, in de vorm van een volledige buffering van adres- en datalijnen. Ook alle controlelijnen zijn gebufferd. Verder is een adresdecodering aanwezig en enige poorten om de read/write lijn te sturen. Aan de kant van de floppy is een en ander toegevoegd om aan de specifieke eisen van deze floppy tegemoet te

;komen. Ook is een simpele voorziening gemaakt om een rustiger  
;gedrag te krijgen van de leeskop van de floppy. Ten laatste  
;vindt U nog wat extra electronica om met twee drives te kunnen  
;werken. Dit is een zeer summiere beschrijving. Het is dan  
;ook niet bedoeld voor nabouw.

;  
;SOFTWARE.

;  
;-----  
;Voor het eerste gedeelte van de software verwijs ik terug  
;naar een bijlage. De werking van de basis-software is uit-  
;voerig beschreven in dat artikel. Ik heb gebruik gemaakt  
;van de volgende configuratie van de floppy : 77 tracks, 26  
;sectoren per track en 128 bytes per sector. IBM-compatibel  
;dus.

;  
;Het tweede gedeelte van de software omvat het DOS zelf.  
;Het is ongeveer 4K lang. De werking van het DOS is gebaseerd  
;op het gebruik van een bitmap. Dat wil zeggen dat men het  
;ruimte op de schijf gaat voorstellen door een verzameling  
;van bits. Elke bit komt overeen met een sector. Dit laat toe  
;op eenvoudige wijze orde te houden in de ruimte die beschik-  
;baar is. Deze bitmap is hier ongeveer 250 bytes lang. Het  
;tweede element in de organisatie van het DOS is de directory.  
;Bitmap en directory nemen track 0 van de schijf in beslag.  
;De directory biedt ruimte aan 194 entries . Elke entry bestaat  
;uit 16 bytes. 6 bytes voor de naam van de file, 2 bytes die  
;de eerste track en sector aangeven, 2 bytes die het aantal  
;sectoren aangeeft waaruit de file bestaat, 2 bytes die de  
;datum aangeven waarop de file ingevoerd is en enkele vrije  
;bytes voor verdere uitbreidingen.  
;Een ander element waarmee voortdurend in het DOS gewerkt  
;wordt is het blok. Een blok komt overeen met een sector  
;op de schijf. Een blok is dus 128 bytes lang. Elk blok  
;bestaat uit de volgende elementen :

;  
; byte 0 - 5 : een filenaam van 6 tekens  
;  
; byte 6 .. 7 : de reverse link, dit is een pointer be-  
;staande uit een track- en een sector-  
;nummer. Deze pointer wijst naar de voor-  
;afgaande blok. Voor goed begrip: in de  
;directory vindt men track en sector van  
;de eerste blok van een file. In die eerste  
;blok vindt men dan een pointer naar de  
;vorige blok (die is er in dit geval niet  
;en wordt dan arbitrair gelijk aan nul ge-  
;steld) en een pointer naar de volgende  
;blok. In de laatste blok is de forward  
;pointer gelijk aan nul. Dit biedt de moge-  
;lijkheid zowel vooruit als achteruit door  
;een file te stappen.  
;  
; byte 8 .. 9 : forward link  
;  
; byte 10 : ongebruikt  
;  
; byte 11 : teller die het aantal nuttige bytes aan-  
;geeft in een blok dat slechts gedeeltelijk  
;met data gevuld is.  
;

```
; byte 12 .. 13 : blokteller, geeft aan het hoeveelste
;                 blok in een file dit is.
; byte 14 .. 127 : data
;Hiermee zijn de voornaamste elementen aangestipt.
;Wat nu de bevelenset betreft van dit basisdos, die is erg
;beperkt. Dat is met opzet zo gedaan. Alle gangbare bevelen
;zijn samengevat onder meer handelbare vorm in de utilities.
;Die had ik deel drie genoemd van de software.
;bevelenset:
;-----
;1) save :save zet een file onder een filenaam op schijf.
;        Er zijn twee mogelijkheden in dit bevel. Men
;        kan een file saveen met de bedoeling dat men ze
;        zal laden en direct runnen (een voorbeeld hier:
;        dat kan BASIC zijn). In dat geval laad men die
;        file en wordt ze na laden direct gestart op een
;        vooraf opgegeven adres.
;        vb. save basic:0
;           *limits 2000,4aff (de computer prompts met *)
;           *go 4065
;        De file is nu van $2000 tot $4aff geladen op de disk
;        in drive 0 onder de naam basic. Indien de file nu
;        geladen wordt door 'run basic:0' dan zal na laden
;        basic starten op $4065
;        Een tweede mogelijkheid is een file gewoon bewaren.
;        Bijvoorbeeld een datafile.
;        vb. save basic:0
;           *limits 2000,4aff
;           *quit
;        Dos herkent slechts de eerste letter van een bevel.
;2)run : laad een file en start eventueel (zie boven).
;        vb. run basic of
;           run basic:0 (default drive nummer is 0)
;3)limits : geeft de grenzen aan van de te saveen file.
;        vb. limits 1000,2000
;4)go : geeft het startadres aan voor runnen na laden.
;        vb. go 2000
;5)quit : geeft aan dat na laden van een file naar het dos
;        moet teruggekeerd worden.
;6)delete : vaast een file uit de directory.
;        vb. delete basic
;7)rename : geeft een nieuwe naam aan een file.
;        vb. rename bas basic
;        Een eigenaardigheid is dat men eerst de nieuwe naam
;        moet opgeven en dan de oude. Ook mag men niet renamen
;        van een schijf naar een andere.
;8)jump :voert een sprong uit naar een hexadecimaal opge-
;        geven adres.
;        vb. jump 2000
```

;)format : dit bevel formatteert een schijf in drive 0. Het  
; moet met de nodige voorzichtigheid gehanteerd worden.  
; Formatteren houdt in dat alles van de schijf overschreven  
; wordt. Er is als optie voorzien dat kan nagegaan worden  
; of de formattering goed verlopen is.  
; vb. format  
; Ziezo dat waren de enkele bevelen van het basisdos.  
; Het heeft echter nog veel meer in petto. De structuur van  
; het programma laat o.a. toe dynamisch geheugen toe te  
; kennen aan een file. Ik zal hierna nog enkele van de sub-  
; routines die aanwezig zijn en enkele van hun mogelijkheden  
; opsommen.  
; Daar ik niet direct weet waar te beginnen zal ik een aantal  
; dingen die ik bij het doorbladeren van de documentatie tegen  
; kom na elkaar opsommen en wat uitlezen.  
; De mogelijke te gebruiken filenamen moeten bestaan uit 1 tot  
; 6 ascii tekens met een waarde groter dan \$20. Het mogen dus  
; ook allen cijfers zijn of iets dergelijks. De filenamen kunnen  
; uitgebreid worden met een extensie die aanduidt op welke drive  
; de file staat of moet komen. vb. assemb:0  
; De commandostrings die men opgeeft gaan eerst een buffer in  
; waarin ze nog enige editing kunnen ondergaan alvorens uitge-  
; voerd te worden. Een manier van werken ongeveer analoos aan  
; wat we bij basic of de assembler kennen.  
; De commandos worden door een routine TEXTIN in een buffer  
; geladen en men kan dan met behulp van een andere routine  
; er telkens de volgende letter uithalen.  
; Interessante routines zijn FDREAD en FDWRITE die toelaten  
; van respectievelijk een karakter te schrijven naar of te  
; lezen van disk. Men kan dus zonder vooraf iets te weten over  
; de lengte van de file steeds een byte meer wegschrijven.  
; Bijvoorbeeld wegschrijven van een file gaat als volgt:  
; men opent de file voor output. Dat wil zeggen dat men eerst  
; controleert of de naam die men er wil aan geven nog niet  
; bestaat en dan een buffer met die file associeert. Er gebeurt  
; ook nog wat huishoudelijke administratie. Van geen belang  
; nu. Eens de file geopend kan men naar hartelust bytes weg-  
; schrijven d.m.v. de routine BYTOT. Men moet zich van wat er  
; intern gebeurt niets aantrekken. Oordeelt men dat alles er  
; in staat dan sluit men de file d.m.v. CLSOT en klaar is kees.  
; Op een analoge eenvoudige manier kan men files lezen.  
; Indien men bijvoorbeeld voor een two-pass assembler een file  
; twee keer moet lezen opent men de file voor input en leest.  
; De tweede keer lezen eist alleen het opnieuw openen voor input  
; en men kan van vooraf aan beginnen.  
; Er bestaat ook de mogelijkheid van random inlezen en uitlezen  
; van sectors via BLKIN en BLKOT. Bijvoorbeeld het lezen van  
; een willekeurige sector vereist het juist zetten van de para-  
; meters voor die sector, dan BLKIN aan te roepen. Dan kan men  
; opnieuw met BYTIN steeds de volgende byte uit dat blok halen.

;Er zijn nog meer door de gebruiker aanroepbare routines, maar  
;ik ga het hier bij houden. Een laatste mogelijkheid van dit  
;programma is toch nog het vermelden waard: program chaining.  
;Daaronder wordt verstaan dat het mogelijk is verschillende  
;programmas aan mekaar te rijgen zo dat bij het beëindigen  
;van een programma automatisch een volgende geladen wordt  
;en uitgevoerd.

;Het derde deel van de software is een programma met utilities.  
;het is ongeveer 2K groot. Het mag zich zowel in ram als in  
;eprom bevinden. De commandinterpreter herkent de bevelen aan  
;de eerste drie karakters. Dit programma biedt het voordeel  
;dat men meer duidelijke foutenbevelen krijgt dan bij het dos  
;zelf. Verder, en niet het minst belangrijke, zijn natuurlijk  
;de nieuwe bevelen die men er in aantreft. Ik zal ze system-  
;atisch behandelen en bij elk een of meer voorbeelden geven  
;die een en ander moeten verduidelijken.

;1) copy : copy laat toe een duplicaat te maken van een file  
;of een of meerdere files samen te voegen tot een. Men kan  
;copieren tussen verschillende drives. Geldige bevelen:

```
; copy Jan:1 from Jan:0  
; copy Jan from piet  
; copy tuur:1 from piet:0,klaas ,Jef:1  
; copy tuur:0 from tuur:1
```

;2) delete : delete vaast een file uit. Geldige bevelen:

```
; delete Jan  
; del Jan:0  
; delniksnill Jan  
; delete Jan ,piet:0,tuur:1,peer ,klaas
```

;3) directory : drukt de directory af van de aangeduide  
;schijf. Er is ook de mogelijkheid die directory op papier  
;af te drukken. Een voorbeeld verduidelijkt dit wat. Zie ook  
;in bijlage.

```
; directory          (drukt direct. van drive 0 af)  
; dir:0  
; dir:1:1 of dir:1:1 (drukt de direct. van drive 1 af op scherm  
;                      en printer)
```

;4) exit : indien geen parameter wordt opgegeven springt exit  
;naar het dos terug. Anders gaat het naar het opgegeven adres.  
; exit

```
; exit 2003 (naar warme start assembler)bv.)
```

;5) free : drukt het aantal vrije bloks af. Een lege schijf  
;heeft 1976 bloks beschikbaar.

```
; free  
; free:1  
; free:1:1  
; free:1
```

```
;  
;6) list : is een bevel dat een file kan listen onder drie  
;verschillende opties. Default optie is F. Dit wil zeggen  
;dat de file zal gelist worden met alle tabs geëxpandeerd en  
;alle carriage returns en line feeds uitgevoerd, vb. voor teksten.  
;Een tweede mogelijkheid is een asciidump te maken van de  
;file: A. De derde mogelijkheid is een hexdump:H. Beide laatste  
;mogelijkheden drukken de file af in blocks van 128 bytes.  
;Zie ook in de bijlagen. De printer optie is ook van toe-  
;passing.  
; list file:0  
; list:l:a file:0  
; list:f file:0  
; list:h basic  
;  
;7) read : leest een programma van cassette. Idem als Junior.  
; read 12  
;  
;8) save : schrijft een programma naar cassette. Idem als  
;Junior.  
; save 23,2000,2345  
;  
;9) rename : geeft een nieuwe naam aan een file. Men mag allen  
;renamen op dezelfde schijf.  
; rename Jan to piet  
; rename Jan:l to pol:l  
; rename Jan to pol:l is echter niet geldig  
;  
;10) run : run werkt op dezelfde manier als in het dos.  
;  
;11) trace : trace is een zeer nuttig bevel als er iets  
;missgelopen is met een file op de schijf. het drukt van  
;de file die getraced wordt voor elke blok de naam af, de  
;reverse link, de huidige sector en track en de forward  
;link. Dus zolang als een file kan getraced worden kan ze  
;ook gelezen worden. In geval de trace ergens midden in  
;een file misloopt, kunnen in ieder geval alle bloks tot  
;waar het misliep gerecupereerd worden door de links in de  
;laatst bekende blok aan te passen via het random-bevel.  
; trace:l basic:l  
; trace demo  
; tra demo:0  
;  
;12) store : werkt op dezelfde wijze als het save-bevel uit  
;het dos.  
;  
;13) format : werkt op dezelfde wijze als het formatbevel uit  
;het dos.  
;  
;14) random : laat toe een willekeurige sector of track te  
;lezen en te schrijven.  
; random in a000 from 29 dit bevel leest track 29 vanaf a000  
; in het geheugen  
; random in a000 from 29,2 dit bevel leest track 29, sector 2  
; vanaf a000 in het geheugen.  
;
```

```
; random out a000 to 29   dit bevel schrijft 26 sectoren vanaf
;                          a000 (dus 26x128 opeenvolgende bytes)
;                          in track 29
; random out a000 to 29,2 dit bevel schrijft 128 bytes vanaf
;                          a000 naar track 29, sector 2
;
;Dat was het dan. Alle reacties graag aan mijn adres of aan
;de redactie
; Van Nieuwenhove Koen
; Consciencestraat 50 bus 4
; b-3000 Leuven
; België
```

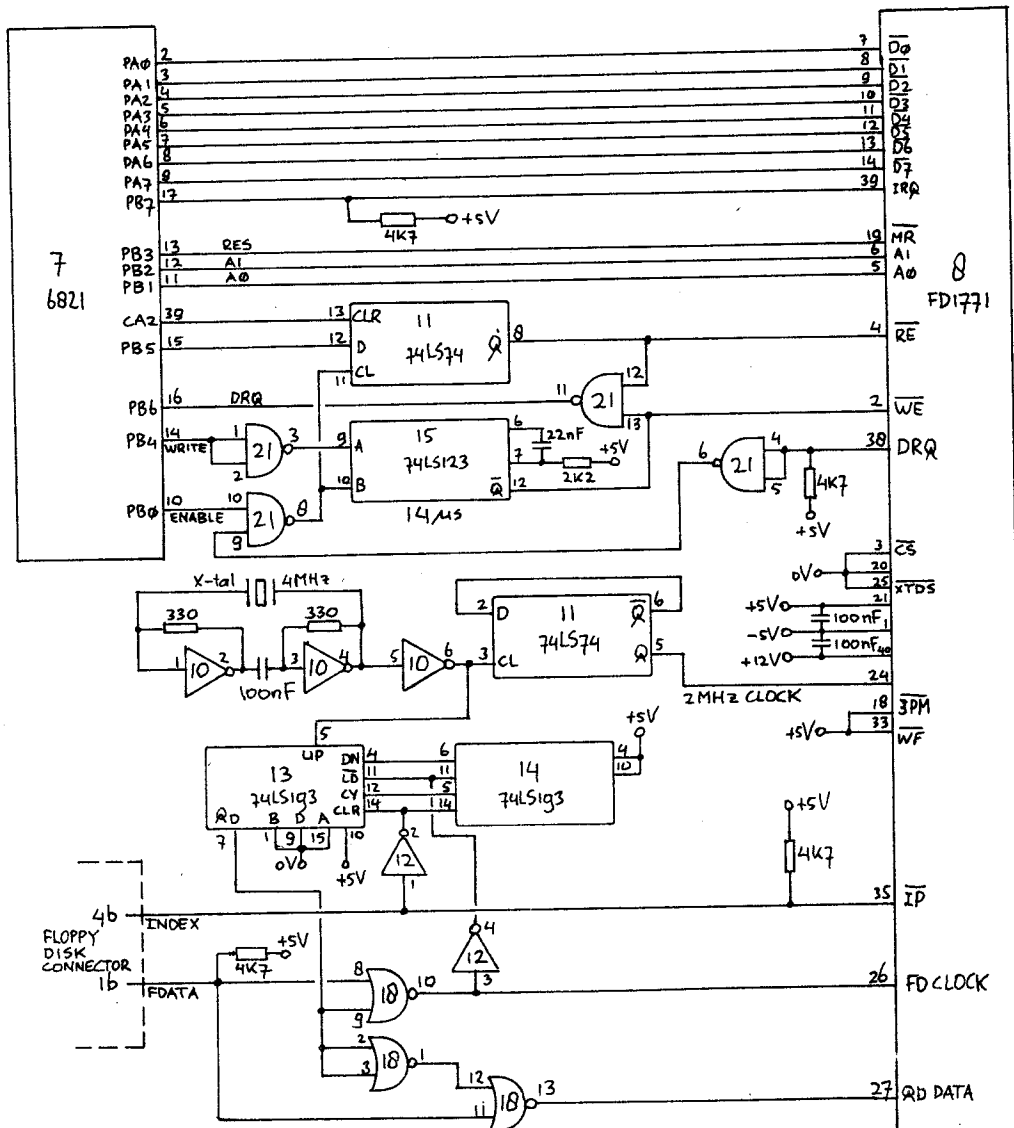
### ;Aanvullins

```
;1) return : return is een nieuw bevel toegevoegd aan de uti-
;ilities. De bedoeling hiervan zal duidelijk worden in de be-
;schrijving van de aanpassing van de assembler van C.Moser aan
;het dos.
;2)Aanpassing van de assembler van C.Moser aan het dos.
;In de assembler zijn drie bevelen aanwezig die voorzien zijn
;voor het werken met een diskdrive: ENTER, LOOKUP en DC.
;ENTER laat toe een file in de directory in te brengen. Ik
;zal een voorbeeld geven van het gebruik:
; ENTER demo          men opent de file demo voor output
; PU F25 2000 2100   dit schrijft in file demo alles wat in de
;                   textfile zit tussen regel 2000 en 2100.
;                   Men geeft een nummer aan dit blok omdat
;                   het mogelijk is meerdere bloks in een file
;                   te hebben.
; PU F34 2000 2400   Dit schrijft bv. een tweede blok in de file
;                   demo.
; ENTER              Dit sluit de file voor output.
;LOOKUP laat toe een file te lezen van de disk. Weer een voor-
;beeld:
; LOOKUP demo        Dit opent de file demo voor input.
; GET F34            Dit leest blok 34 uit demo.
; LOOKUP             Dit sluit de file demo voor input.
;Een laatste commando in de assembler is DC (disk command).
;DC laat toe alle commandos van de utilities van in de assembler
;aan te spreken. Een voorbeeld:
; DC                Dit bevel doet ons overgaan naar de utilities
; ++++++ 6502 PIP VERSION 1.1 ++++++
; +                Dit is de reactie van de computer. We kunnen
;                   nu alle utilities gebruiken.
; RETURN            Zo keren we terug naar de assembler.
;
//
>
```

```

;Aanpassing van Basic V1.1 voor Junior aan Fdos 1.1
;Van Nieuwenhove Koen 23-1-1983
;
;Om de Basic samen met het dos te kunnen gebruiken heb ik er
;een aantal functies aan toegevoegd. Dit had tot gevolg dat
;de Basic nu de geheugenruimte van $2000 tot $4f00 in beslag
;neemt, evenals de zero-page. Het geheugen van $200 tot $7ff
;wordt nu gebruikt door het dos.
;Om al de hierna volgende functies te kunnen toevoegen en het
;probleem toch niet al te complex te laten worden heb ik be-
;sloten het LET-bevel uit basic te elimineren. Dit bevel heeft
;toch geen onvervangbare plaats in de Basic. De routines die
;LET uitvoeren blijven natuurlijk bestaan maar de naam verdwijnt
;uit de bevelenset.
;Alle bevelen behalve DOS APP, DOS LPR en DOS SPR zijn zowel
;vanuit direct als vanuit program mode uitvoerbaar.
;De hierna volgende bevelen werden toegevoegd:
;
;1)DOS APP (filenaam) : dit bevel voert het reeds bekende APPEND
;uit. Het voert tevens een automatische hernummering uit van
;het totale programma.
; vb. DOS APP test:1
;2)DOS LPR (filenaam) : dit bevel laadt het programma met naam
; (filenaam) van disk in het geheugen.
; vb. DOS LPR demo:0
;3)DOS SPR (filenaam) : dit bevel bewaart het aanwezige pro-
;gramma onder de naam (filenaam) op disk.
; vb. DOS SPR test:0
;4)DOS LDA (var),... : dit bevel laadt variabelen van disk in
;het geheugen.
; vb. DOS LDA A$,B,c,c$(I),e,a(I,J)
;5)DOS SDA (var),... : dit bevel bewaart de waarden van de
;gendeemde variabelen op disk.
; vb. DOS SDA a,b,c,s(23,48),a$,c$(8)
;6)DOS OOO (filenaam) : dit bevel opent de file (filenaam)
;voor input.
; vb. DOS OOO TEST9:1
;7)DOS OOT (filenaam) : dit bevel opent de file (filenaam)
;voor output.
; vb. DOS OOT TEST8:1
;8)DOS CLOSE : sluit de op dat moment geopende file voor output.
; vb. : DOS CLOSE
;9)DOS PIP : laat toe van al de utilities van PIP gebruik te
;maken vanuit BASIC. Men keert terug naar BASIC door RETURN.
; vb. DOS PIP
;
; ...
;     RET
;     ) ....      terug in Basic
;
//
>
```





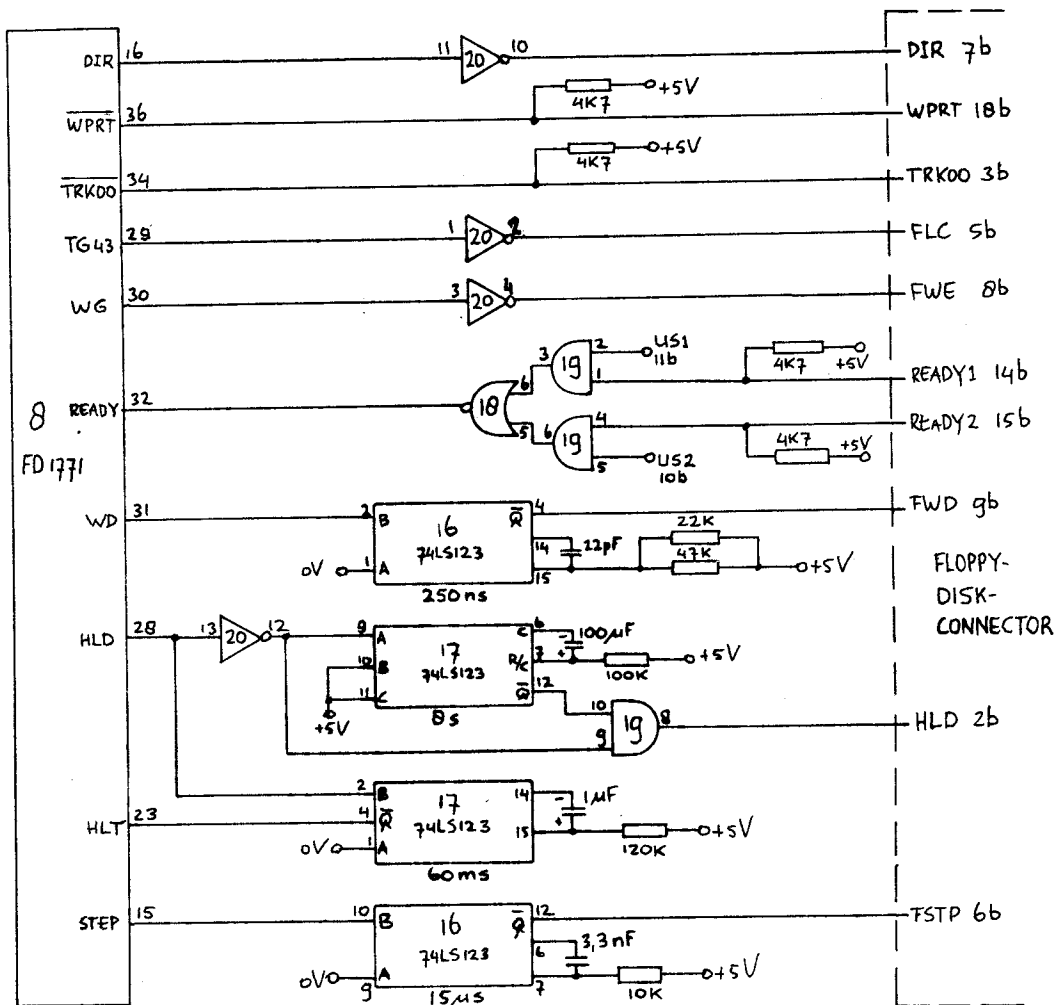
Dit artikel probeert U een indruk te geven van de mogelijkheden welke het D.O.S. te bieden heeft zoals het door onze vriend Koen van Nieuwenhove werd ontwikkeld op basis van de in het artikel genoemde literatuur.

Samen met John van Sprang ben ik enige tijd geleden naar de werkende D.O.S. van Koen gaan kijken in Leuven (Belgie). We zijn van dit uitstapje teruggekeerd met de zekerheid dat

er sprake is van een erg interessant operating system waaraan een groot voordeel is verbonden: het gaat vergezeld van volledige documentatie, waarin o.a. een listing met commentaar. Het is geschreven in het Engels.

De documentatieset zal net als de controllerkaart en de coding in Eprom ter beschikking komen. U kunt bij de redactie inlichtingen inwinnen. (W.L. van Pelt)

(Uitwerking van de schema's: Fridus Jonkman)



## BOEKENLIJST VOOR 6502 GEBRUIKERS.

In de boekenlijst voor 6502 gebruikers wordt informatie verstrekt over in de handel verkrijgbare boeken en tijdschriften.

### CBM 8050 DOS-listing

Betriebsysteem in Detail  
 Dr. Ruprecht, ca. 168 pagina's PB, 10sbl. in sept. kl.  
 ISBN 3-88322-015-9 Prijs: Fl. 125,- 3e druk

Dit naslaawerkje bevat het volledig van commentaar voorzien besturingssysteem van de Commodore 8050 schijfheugeneenheid. Samen met de inleiding van de auteur in de principiële opbouw van het DOS en de manier, waarop de beide processoren gebruik maken van het geheugen bent u in staat elke functie van het DOS te begrijpen, voorzover u de 6502 machinetaal beheerst.

Bij nauwkeurige bestudering vindt u bijvoorbeeld een beschermingsvlag tegen ongewenst wissen in de directory; een mogelijkheid USA-files als machinetaalprogramma's via de schijfheugeneenheid te verwerken; een auto-start optie; of geheugenplaatsen, die aanloop- en uitlooptijd van de diskette bepalen. Wanneer u effectief gebruik maakt van de geboden informatie, kunt u copier- en duplicerbeveiligingen uitwerken en eigen diskette-formaten schrijven of nieuwe data-structuren ontwikkelen.

Naast deze vrij specialistische toepassingen wordt ook de nieuwsgierigheid van programmeurs gewekt: u krijgt hiermee inzicht over de samenwerking tussen twee 6502-processoren. Hierbij komt nog het dataverkeer via PIA's met de centrale 6502-processor van de CBM. Het werken met drie processoren is op zich al een studie waard.

K2FDOS : A Floppy Disk Operating System for the 8080

Kenneth B. Welles \$ 20.00

BYTE BOOKS

ISBN 0-07-069206-8

(Aan dit boek ontleende Koen van Nieuwenhove het D.O.S. uit bovenstaand artikel.)

### PET INTERFACING

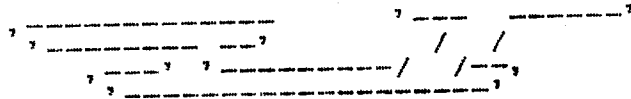
J.M. Downey en S.M. Rogers  
 Aanwijzingen voor bouw en gebruik van een drietal typen gespecialiseerde interfaces.  
 262 p.; Prijs: Fl. 70,10

### APPLE PASCAL GAMES

D. Hersert en J.F. Kalash  
 Computerspelletjes in alle vormen en variëteiten geprogrammeerd in een gemakkelijk leesbare programmeertaal.  
 371 p.; Prijs: Fl. 65,70

\*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*  
 \*\*\*\*\*

STAR - TREK



MOEILIJKS GRAAD 1, 2 OF 3 :  
 (1=NORMAAL, 2=MOEILIJK, 3=GEVAARLIJK.) : ?  
 MOMENTJE A.U.B.

DIT ZIJN JE ORDERS:  
 Vernietig de 21 Klingonen en  
 3 Klingonbasischepen.  
 Je hebt daarvoor de tijd tot sterdatum: 3531  
 er zijn 2 basissterren om de Enterprise  
 opnieuw te bevoorraden. Pas op voor aanvallen op  
 een basisster. Meer dan 3 Klingonschepen kunnen  
 hem vernietigen.

JE OPDRACHT BEGINT IN HET KWADRANT: DENEB IV  
 -----  
 \* \* \*  
 \* \*  
 (\*)  
 \* \*  
 -----  
 STERREDAGEN  
 STERREDATUM  
 TOESTAND  
 KWADRANT  
 SECTOR  
 PHOTON TORPEDO'S  
 TOTAAL ENERGIE  
 SCHILD  
 KLINGONEN  
 KLINGONBASIS

31  
 3500  
 GROEN  
 2, 8  
 6, 7  
 10  
 3000  
 0  
 21  
 3

COMMANDO:  
 COMPUTER WACHT OP COMMANDO:

COMPUTER OVERZICHT VAN HEELAL VOOR KWADRANT: 2, 8

	1	2	3	4	5	6	7	8
1	*****	*****	*****	*****	*****	*****	*****	*****
2	*****	*****	*****	*****	*****	*****	*****	0006
3	*****	*****	*****	*****	*****	*****	*****	*****
4	*****	*****	*****	*****	*****	*****	*****	*****
5	*****	*****	*****	*****	*****	*****	*****	*****
6	*****	*****	*****	*****	*****	*****	*****	*****
7	*****	*****	*****	*****	*****	*****	*****	*****
8	*****	*****	*****	*****	*****	*****	*****	*****

COMMANDO:  
 KOERS (0-9)?  
 SPRONGFACTOR (0-8)?  
 SCHADE CONTROLE MELDT: PHOTON LANCEERBUIZEN REPERATIE GEVORDERD.

LT. UHURA MELDT EEN BOODSCHAP VAN DE VLOOTCOMMANDANT:  
 GEEN TOESTEMMING OM RUIMTEGRENZEN TE Overschrijden.  
 SPRONGMOTOREN GESTOPT IN SECTOR: 6, 8  
 VAN KWADRANT: 2, 8  
 AT SECTOR 6, 8 OF QUADRANT 2, 8

-----  
 \* \* \*  
 \* \*  
 (\*)  
 \* \*  
 -----  
 STERREDAGEN  
 STERREDATUM  
 TOESTAND  
 KWADRANT  
 SECTOR  
 PHOTON TORPEDO'S  
 TOTAAL ENERGIE  
 SCHILD  
 KLINGONEN  
 KLINGONBASIS

30  
 3501  
 GROEN  
 2, 8  
 6, 8  
 10  
 2982  
 0  
 21  
 3

**SUPER STAR TREK**

Nederlandse gebruiksaanwijzing  
Door: A.S.Hankel,  
W. Kloosstraat 32,  
7606 BB Almelo.

1. Inleiding:

Super-star-trek is een Basic-programma, gebaseerd op de televisie-serie **STARTREK**. Er bestaan diverse versies, geschreven voor diverse micro-computers, meestal voornamelijk afhankelijk van het gebruikte type terminal. Deze versie gaat uit van van een VDU van 16 regels, elk met 64 kolommen, en is dus geschikt voor gebruik met de door de meeste Junior-eigenaars gebruikte Elekterminal.

De eerste versie van dit programma is al in 1972 geschreven door ene Mike Mayfield, voor een HP2000C. Het werd een standaard-programma in februari 1973, toen het werd opgenomen in de software-bibliotheek van HP. (Hewlett Packard)

Later werd het programma aangepast voor BASIC-PLUS, de basic die gebruikt wordt door DEC-computers. Weer later werd het aangepast voor Microsoft-Basic, dus meerdere statements op één regel. De laatste wijziging bestaat uit het vertalen van het programma in het Nederlands.

2. De opzet:

Met Super-star-trek bevind je je in het heelal, dat voor deze gelegenheid is verdeeld in 2 kolommen, elk van 8 regels. Elke (kolom, regel) is weer verdeeld in 4 kolommen, elk met zijn eigen -astrologische- naam. Dit resulteert uiteindelijk in een 8 x 8 matrix. Elke matrix-eenheid is weer opverdeeld in 64 sekties, ook weer 8 x 8.

3. Het doel:

Jij, de man/vrouw achter het toetsenbord, bent de gezagvoerder van het ruimteslagschip "Enterprise". Je taak is, het heelal, en dus ook de aarde, ruimtelijk bolwerk van cultuur, techniek en filosofie, te bevrijden van de aanvallen van de alom gehate 'Klingon', een wreed en brutaal soort wezens. Hulpmiddelen daarbij zijn een ultra-modern slagschip, gebouwd volgens de meest moderne technologie, een bekwame staf van medewerkers, een aantal krachtige wapens, en een zeer effectief afweersysteem. Helaas..... zelfs de meest moderne techniek faalt wel eens, zoals je zal blijken.

4. Vriend & vijand:

Goed, je hebt een, laat we zeggen, perfect slagschip. Helaas heeft elk slagschip, hoe perfect het ook mag zijn, slechts een beperkte hoeveelheid munitie aan boord. Gelukkig maar, dat daarin is voorzien. Ter ondersteuning van jouw missie hebben de aardse autoriteiten gezorgd voor een aantal zogenaamde **basissterren**. Hopelijk kom je ze op tijd tegen, anders zou het niet onmogelijk zijn dat de Klingons je te grazen nemen doordat je munitie en/of energie verbruikt is. Ook Klingon's hebben hun ondersteuning, in de vorm van, van een moeilijk doordringbaar schild voorzien, **Klingonbasis**. Klingonbases hebben, vanwege hun relatief gebrekkige technologie, alleen een achtergrondfunctie. Maar: mocht je een Klingonbasis tegenkomen, vernietig hem dan wel, zodat bevoorrading van Klingon's niet meer mogelijk is. Helaas gaat de vernietiging van een Klingonbasis niet onopgemerkt voorbij aan eventueel andere, in het heelal aanwezige Klingonbases. Hun reactie zal zijn, dat er, ten eerste, een extra aantal Klingon's in de strijd komen. Ten tweede, zal de Kling-

onbasis Klingon's afsturen op jouw basisster met het doel hem te vernietigen.

5. Moeilijkheidsgraad: 1-Beginner 2-Gevorderde 3-Waaghals:

Als je het programma hebt gestart, wordt op een gegeven moment gevraagd, volgens welke moeilijkheidsgraad je wilt functioneren. Je kunt kiezen tussen 1, 2 of 3. Als je dit voor het eerst leest, neem dan 1. Denk je een professional te zijn, neem dan 2. Mocht je daarop ook uitgekeken zijn probeer dan 3 maar. Je komt er dan wel achter, dat het verschil tussen 2 en 3 veel groter is dan tussen 1 en 2. Hoe? .... Dat merk je vanzelf!

6. Spelregels en opmerkingen:

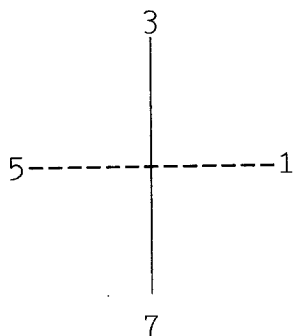
Binnen Super-star-trek worden de volgende symbolen gebruikt:

- <\*> Enterprise
- >!< Basisster
- +K+ Klingon
- \* Ster
- +M+ Klingonbasis
- 000 Beschermerschild om Klingonbasis

Er zijn 9 kommando's, elk van 3 letters. Het is voldoende om de eerste letter ervan in te toetsen, gevolgd door een CR. De kommando's zijn:

- NAV Navigatie; koers en afstand.
- KAS Korte afstand scan; overzicht huidige sekte.
- LAS Lange afstand scan; overzicht omliggende kwadranten.
- PHA Phaser controle.
- TOR Torpedo controle.
- SCH Schild controle; bescherming tegen vijandelijke wapens.
- DAM Overzicht schadetoestand.
- COM Hulpcomputer wordt ingeschakeld.
- XXX Om ontslag in te dienen.

Toelichting op een aantal kommando's:



NAV vraagt om Koers (0-9) en Sprongfactor (0-8). Een koersgetal tussen 0 en 10 komt overeen met de richting, gerelateert aan een kompas, en wel als volgt: 1=oost 2=noordoost 3=noord 4=noordwest 5=west 6=zuidwest 7=zuid 8=zuidoost 9=oost, en is dus gelijk aan 1. Een sprongfactor mag variëren tussen 0 en 8. Faktor 1 komt overeen met een sprong van precies 1 kwadrant. Let wel op dat je binnen de grenzen van het heelal blijft. Zowel koers als sprongfactor hoeven niet persé integer te zijn.

LAS geeft een overzicht van de je omringende kwadranten. Jijzelf staat daarbij in het midden. Info over de omringende kwadranten haal je uit het rijtje getallen in elk kwadrant. De betekenis ervan is, van links naar rechts: 1-Aantal Klingonbases 2-Aantal Klingon's 3-Aantal basissterren 4-Aantal sterren.

PHA vraagt om de hoeveelheid phaser-energie die je op een Klingon, of op meerdere Klingon's wilt afvuren. Deze hoeveelheid gaat van de totale energievoorraad af. Te weinig phaser-energie zal weinig of geen effect geven.

TOR vraagt om de richting van de af te vuren torpedo. Voor de koers geldt hetzelfde richtings-effect als voor NAV.

SCH controleert de hoeveelheid energie die je gebruikt voor het beschermerschild om de Enterprise. Te weinig schild kan tot vernietiging leiden, terwijl een te groot schild teveel energie aan de

totaal-hoeveelheid, waardoor je niet meer zou kunnen navigeren. DAM geeft een overzicht van de status van essentiële systemen van de Enterprise. Een status kan variëren tussen 1 en +1. Waarde 0 is neutraal. Een negatieve waarde betekent dat dat systeem tijdelijk- uitgeschakeld is.

COM schakelt de hulpcomputer in. Dit kommando is weer onderverdeeld in een 6-tal sub-kommando's, namelijk :

- 0 Overzicht totale ruimte.
- 1 Rapport huidige toestand.
- 2 Photon(en) torpedobaan gegevens.
- 3 Basisster navigatiegegevens.
- 4 Koers/afstand berekening.
- 5 Ruimte regio overzicht.

**Vernietigen van een Klingonbasis.**

Het vernietigen van een Klingonbasis kan alleen m.b.v. torpedo's. Eén is er nodig voor het beschermchild, één voor de Klingonbasis zelf. Het is niet mogelijk om torpedobaan naar de Klingonbasis door COM te laten berekenen. Wanneer de torpedo de Klingonbasis mist, zal het resultaat zijn, dat jij a.h.w. door het heelal geslingerd wordt. Je komt dan in een niet te voorspellen kwadrant terecht.

In het oorspronkelijke programma was het mogelijk dat er een foutmelding optrad, wanneer je je verplaatste binnen het kwadrant, en er 1 of meerdere Klingon's aanwezig waren. Dit is, althans gedeeltelijk, ondervangen. In deze versie bestaat er een melding van UFO's, die de zaak bedreigen. Gevolg is echter wel, dat het door KAS weergegeven beeld onsamenhangend overkomt, althans op de VDU. Bij twijfel is het raadzaam COM te raadplegen omtrent de juiste positie van Klingon('s).

Het opnieuw bevoorraden van de Enterprise geschiedt door tegen een basisster te manoeuvreren, met behulp van NAV. Je krijgt dan nieuwe torpedo's en energie.

Wanneer jouw basisster aangevallen wordt door Klingon's -en dat zal het geval zijn als jij een Klingonbasis vernietigd hebt- zal de aanwezigheid van 3 of meer Klingon's in het zelfde kwadrant als waar jouw basisster zich bevindt, ertoe leiden dat die basisster vernietigd wordt. Afhankelijk van het aantal basissterren kan het noodzakelijk zijn, dat met alle mogelijke middelen te voorkomen. Immers, zonder basisster ben je helemaal op jezelf aangewezen.

Opmerkingen over PHA (phaser).

A: Het effect van PHA neemt af naarmate de afstand tussen jou en de Klingon('s) groter is.

B: Wanneer je hard genoeg geraakt wordt door een Klingon, zal hij schade veroorzaken, zodat er automatisch een DAMreport komt.

C: Als jij een Klingon niet hard genoeg raakt, zal hij geen schade hebben. Je sensors zullen je daar info over geven.

D: DAM laat je automatisch weten wanneer één of meerdere systemen gerepareerd zijn.

**VEEL SUCCES !**

Het programma SUPERSTARTREK is op een aparte band te verkrijgen tezamen met de handleiding door Fl. 17,50 over te schrijven op postrekening 841433 t.n.v. W.L. van Pelt te Krimpen a.d. IJssel.

De listing van het Basic programma is tevens op papier te verkrijgen voor Fl. 7,50.

TEKST 6502 ASSEMBLER V3.1 PAGE 01

0010: 1780  
 0020:  
 0030:  
 0040:  
 0050:  
 0060:  
 0070:  
 0080:  
 0090:  
 0100:  
 0110:  
 0120:  
 0130:  
 0140:  
 0150:  
 0160:  
 0170:  
 0180:  
 0190:  
 0200:  
 0210:  
 0220:  
 0230:  
 0240:  
 0250:  
 0260:  
 0270:  
 0280:  
 0290:  
 0300:  
 0310:  
 0320:  
 0330:  
 0340:  
 0350:  
 0360:  
 0370:  
 0380:  
 0390:  
 0400:  
 0410:  
 0420:  
 0430:  
 0440:  
 0450:  
 0460:  
 0470:  
 0480:  
 0490:  
 0500:  
 0510:  
 0520:  
 0530:  
 0540:  
 0550:  
 0560: 1780  
 0570: 1780  
 0580: 1780  
 0590:  
 0600: 1780 D8  
 0610: 1781 A2 00  
 0620: 1783 18  
 0630: 1784 A0 20  
 0640: 1786 8C 95 17

TEKST ORG \$1780

\*\*\* TEKSTVERWERKING MET TELETYPE KSR33 OF ASR \*\*\*

DOOR: J. C. J. BEIJER  
 BASTINGLAAN 7  
 2614 GP DELFT  
 TEL. 015-134269

**KIM-1**

DIT PROGRAMMA IS GEBASEERD OP EEN PROGRAMMA TEKSTVERWERKING VAN DRS. A. R. VAN DER BURG, DIE ZO VRIENDELIJK WAS ZIJN ORIGINELE VERSIE TER BESCHIKKING TE STELLEN. IN AFWIJKING VAN HET ORIGINELE PROGRAMMA ZIJN DE "WRITE" EN "REWRITE" PROGRAMMA'S SAMENGEPERST IN HET GEHEUGENDEELTE VAN ADRES \$1780 T/M \$17E5 MET ALS KOPIEENTELLER ZERO-PAGE ADRES \$00EE, TERWIJL TEVENS FLOWCHARTS EN AANGEPASTE GEBRUIKSAANWIJZING ZIJN TOEGEVOEGD. DOOR TOEPASSING VAN DATAOPSLAG VANAF ADRES \$2000 IS HET MOGELIJK PER CA. 4K BYTES EEN DICHTBEDRUKTE "A4" PAGINA TE PRODUCEREN. DE MOGELIJKHEDEN VAN HET PROGRAMMA ZIJN:

1. WRITE EEN VIA HET TOETSENBORD VAN DE TELETYPE INGETIKTE TEKST OP PAPIER AFDrukKEN VIA DE KIM-1 (DUS NIET RECHTSTREEKS OF LOCAL)
2. DAARNA 1 A 100 KOPIEEN LATEN AFDrukKEN DOOR DE TELETYPE M.B.V. DE BIJ "1" IN HET GEHEUGEN VAN DE KIM-1 OPGESLAGEN DATA.
3. TIJDENS HET INTIKKEN (WRITE) KORREKTIE VAN ZOJUIST GEMAAKTE FOUTEN (HET GEHEUGEN "VERGEET" DE N KARAKTERS ALS U N MAAL RD (RUB-OUT) INDRUKT.

OPMERKINGEN:

1. ALLEEN VOOR VERANDERAARS !

VOOR HET OPNEMEN VAN DIT (NIET RELOCATABLE) PROGRAMMA IN EEN EPROM MOETEN DE VARIABELEN BIJEN HET PROGRAMMA WORDEN GEBRACHT (BIJV. NAAR "ZERO PAGE"). HIERTOE ZIJN WIJZIGINGEN NOODZAKELIJK VAN DE INHOUD VAN DE VOLGENDE ADRESSEN:  
 \$1787-\$1788-\$179F-\$17A0-\$17B6-\$17B7-\$17BD-\$17BE-\$17C0  
 \$17C1-\$17CA-\$17CB-\$17CD-\$17CE  
 EN/OF ALS U EEN ANDER "EERSTE DATA-ADRES" KIEST DAN OOK:  
 \$1782-\$1785-\$1794-\$1795-\$17BB-\$17C3-\$17D0-\$17D1-\$17D6-\$17D7

2. VOOR HET AANSLUITEN VAN DE TELETYPE VIA 4 DRAADJES AAN DE APPLICATION-CONNECTOR WORDT VERWEZEN NAAR DE KIM-1 USER MANUEl (BLZ 17-18-19-20-50-51-52 EN 53) U VINDT HIER TEVENS GEGEVENS OVER EEN SCHAKELAAR DIE NOG NODIG IS VOOR DE KEUZE: KIM-1 TOETSENBORD OF TELETYPE-TOETSENBORD (KIM-1 LOCAL). NB. EEN PRETTIGE WERKWIJZE IS DE SCHAKELAAR OP DE KAST VAN DE TELETYPE EN VIA EEN KABEL MET EEN 6(OFF MEER) POLIGE KONNEKTOR OP DE KIM-1 AANSLUITEN. BIJ EEN "VERHUIZING" HOEFT U DAN NIET ALLES TEGELIJK TE VERSHOVENEN.

GETCH \* \$1ESA  
 PRTBYT \* \$1E3B  
 OUTCH \* \$1EA0

START1 CLD : INIT-ROUTINE  
 LDxIM \$00  
 CLC  
 LDyIM \$20  
 STY \$1795

```

0650: 1789 90 01          BCC HAALCH ; OMDAT $00F1 IS GELADEN MET $00 IS CARRY ALTIJD CLEAR
0660: 178B CA          TERUG DEX ; VERMINDER X MET 1 ZODAT VORIG TEKEN KAN WORDEN OVERSCHREVEN
0670: 178C 20 5A 1E     HAALCH JSR GETCH ; MONITOR SUBROUTINE OM TEKEN VAN TELETYP OVER TE NEMEN
0680: 178F C9 7F          CMPIM $7F ; KONTROLEER OF RUB-OUT IS INGEDRUKT
0690: 1791 F0 F8          BEQ TERUG ; ZO JA, DAN NAAR "TERUG"
0700: 1793 9D 00 20     STAAX $2000 ; ZO NEE, BERG HET TEKEN DAN OP OP ADRES $2000+X+DELTA
0710: 1796 E8          INX ; VERHOOG X, ZODAT HET VOLGENDE ADRES WORDT AANGEWEEZEN
0720: 1797 C9 04          CMPIM $04 ; KONTROLEER OF HET EINDE VAN DE TEKST (EOT) BEREIKT IS
0730: 1799 D0 10          BNE VERDER ; INDIEN NIET, NAAR "VERDER"
0740: 179B 20 B1 17     JSR SUBR ; INDIEN WEL, DAN NAAR "SUBR"
0750: 179E AD 95 17     LDA $1795 ; BRENG HO-DEEL LAATSTE DATA-ADRES NAAR ACCU
0760: 17A1 20 3B 1E     JSR PRTBYT ; EN PRINT BYTE
0770: 17A4 8A          TXA ; BRENG LO-DEEL LAATSTE DATA-ADRES NAAR ACCU
0780: 17A5 20 3B 1E     JSR PRTBYT ; EN PRINT BYTE
0790: 17A8 4C 2D 19     JSR $192D ; TERUG NAAR DE MONITOR
0800: 17AB 20 B1 17     VERDER JSR SUBR
0810: 17AE 18          CLC
0820: 17AF 90 DB          BCC HAALCH
0830: 17B1 E0 00          SUBR CMPIM $00
0840: 17B3 D0 03          BNE RTS
0850: 17B5 EE 95 17     INC $1795 ; VERHOOG HO-DEEL VAN DATA-ADRES MET 1
0860: 17B8 60          RTS
0870: 17B9 D8          START2 CLD ; REWRITE
0880: 17BA A0 20          LDYIM $20
0890: 17BC 8C D1 17     STY $17D1
0900: 17BF 8C D7 17     STY $17D7
0910: 17C2 A2 00          LDXIM $00
0920: 17C4 E8          NEXT INX
0930: 17C5 E0 00          CMPIM $00
0940: 17C7 D0 06          BNE PAGTEL
0950: 17C9 EE D1 17     INC $17D1
0960: 17CC EE D7 17     INC $17D7
0970: 17CF BD 00 20     PAGTEL LDAAX $2000
0980: 17D2 20 A0 1E     JSR DUTCH ; PRINT CHARACTER
0990: 17D5 BD 00 20     LDAAX $2000
1000: 17D8 C9 04          CMPIM $04 ; END OF TRANSMISSION ?
1010: 17DA D0 E8          BNE NEXT
1020: 17DC F8          SED
1030: 17DD A5 EE          LDAZ $EE ; BRENG INHOUD VAN AANTAL "KOPIEEN"-TELLER NAAR ACCU
1040: 17DF 38          SEC
1050: 17E0 E9 01          SBCIM $01 ; (A) - 1 NAAR A
1060: 17E2 85 EE          STAZ $EE ; (A) NAAR "AANTAL KOPIEEN"-TELLER
1070: 17E4 D0 D3          BNE START2
1080: 17E6 00          BRK ; LAAT TELETYP PROMPTING MESSAGE AFDRIJVEN
1090:
1100:
1110:
1120:
1130: NETSPANNING AAN
1140: RS (RESET)
1150: $17F2 - 93
1160: $17F3 - 02
1170: $17F9 - 00
1180: $17FB - 1C
1190: $17FC - 00
1200: $17FD - 1C
1210: $17FE - 00
1220: $17FF - 1C
1230: $00F1 - 00
1240:
1250: $1873
1260: GO
1270: START CASSETTERECORDER MET PROGRAMMA "TEKSTVERWERKING"
1280: NADAT PROGRAMMA IN GEHEUGEN STAAT DE CASSETTE TERUGSPOELN.

```

TEKST

6502 ASSEMBLER V3.1

PAGE 03

```

1290:
1300:          OPNEMEN VAN TEKST IN GEHEUGEN.
1310:
1320:          $1780
1330:          TELETYPE VAN LOCAL OP KIM-1
1340:          G
1350:          RETURN
1360:          RETURN
1370:          LINE FEED
1380:          LINE FEED
1390:          GEMENSTE TEKST INTIKKEN, EEN PRAKTISCH BEDIN IS BIJVOORBEELD
1400:          EEN KNIPLIJD:-----
1410:          GEVOLGD DOOR
1420:          RETURN
1430:          LINE FEED
1440:          LINE FEED
1450:          OM DAARNA MET DE EIGENLIJKE TEKST TE BEGINNEN EN VERVOLGENS
1460:          LINE FEED
1470:          LINE FEED
1480:          LINE FEED
1490:          LINE FEED
1500:          INDIEN GEMENST EEN KNIPLIJD EN
1510:          AFSLUITEN MET DE TWEE TOETSEN CTRL EN EDT/D (TEGELIJKERTIJD).
1520:
1530:          REWRITE.
1540:
1550:          ZIE HIERVOOR TOT EN MET HET TERUGSPOELEN VAN DE CASSETTE.
1560:          $ADRES MOET HET GEMENSTE AANTAL KOPIEEN BEVATTEN, 01 A 99 (VOOR
1570:          100 KOPIEEN 00 INTIKKEN).
1580:          START OP ADRES $1789
1590:          NA SPATIE MET G STARTEN.
1600:          HET INGETOETSTE AANTAL KOPIEEN WORDT HIERNA AFGEDRUKT.

```

## MAAK VAN JUNIOR EEN SENIOR

In DE 6502 KENNER nr. 26 publiceerde de heer P. Fransen het artikel met bovengenoemde titel, handelend over de wijze waarop de 5 1/4 inch floppy drives aan de JUNIOR gekoppeld kunnen worden en het PROTON disk operating system dat hiermee in verband staat. Zoals bekend worden aan leden van de KIM Gebruikers-Club Nederland kortingen gegeven op disk-drive en controllerkaart, waarbij men bovendien een aanzienlijk pakket software ontvangt voor rond Fl. 1085,=.

Inlichtingen hierover zijn te verkrijgen bij de redactie.

Het artikel is op papier beschikbaar door Fl. 10,= te gireren op postrekening nr. 841433 t.n.v. W.L. van Pelt te Krimpen a.d. IJssel.

## FOCAL VOOR JUNIOR

Deze Basic-achtige interpreter met veel wiskundige functies zoals:  $\sin(x)$ ,  $\cos(x)$ ,  $\tan(x)$ ,  $\arctan(x)$ , worteltrekken,  $\ln(x)$ ,  $\log(x)$ , en z'n krachtige strings is in Nederland!

Van de Firma The 6502 Program Exchange, 2920 West Moana, Reno, NV 89509, America, heeft dhr. J. Jansen toestemming om namens deze firma de JUNIOR FAST FOCAL hier in Nederland en omgeving te verkopen.

In America kost de cassette met manual \$40,= plus 15% verzendkosten. Focal zal de leden worden aangeboden voor vrijwel dezelfde prijs.

Informaties:  
Dhr. J. Jansen,  
Gerardsweg 30,  
6525 RT Nijmegen.  
Tel.: 080 - 562082.

## BASIC COMPRESSOR FOR THE PET

Gebruikers van een PET kunnen bovengenoemd programma uit editie 25 op band bestellen door contact op te nemen met Ruud Uphoff, sekretaris van onze club.

## BASIC PROGR. SERVICE RAPPORT

In DE 6502 KENNER nr. 25 publiceerde Maarten van Hintum dit omvangrijke programma dat in de praktijk wordt gebruikt. Het is een educatief stuk programmeertechniek dat voor de liefhebbers apart op band verkrijgbaar is. Bestellen door Fl. 12,50 over te schrijven op postrekening nr. 841433 t.n.v. W.L. van Pelt te Krimpen a.d. IJssel, met vermelding: Cassette Service Rapporten.



0010:  
0020:  
0030:  
0040:  
0050:  
0060:  
0070:  
0080:  
0090:  
0100:  
0110:  
0120:  
0130:  
0140:  
0150:  
0160:  
0170:  
0180:  
0190:  
0200:  
0210:  
0220:  
0230:  
0240:  
0250:  
0260:  
0270:  
0280:  
0290:  
0300:  
0310:  
0320:  
0330:  
0340: 0200  
0350:  
0360:  
0370:  
0380:  
0390:  
0400:  
0410:  
0420:  
0430:  
0440:  
0450:  
0460:  
0470:  
0480:  
0490:  
0500:  
0510: 0200 A9 10  
0520: 0202 85 10  
0530: 0204 A9 50  
0540: 0206 85 11  
0550: 0208 A9 60  
0560: 020A 85 12  
0570: 020C A9 FF  
0580: 020E 8D 83 1A

**\* TELEFOONBEANTWOORDER \***

DOOR : VANDENDRIESSCHE HARRY  
ELISABETHLAAN 26  
3070 KORTENBERG  
BELGIE.  
TEL.: 02 / 7599777

DE MEESTE HOME COMPUTERS ZIJN UITGESCHAKELD WANNEER MEN AFWEZIG IS , OM NU DEZE " SPARE TIME " TE BENUTTEN MET HIERNAVOLGEND SIMPEL PROGRAMMA WAARDIJ MEN DE COMPUTER GEBRUIKT ALS TELEFOONBEANTWOORDER.

ALS HARDWARE VOLSTAAT HET DE TELEFOONLIJN VAN EEN BELSTROOMDETEKTIESTROOMMETER EN EEN IMPEDANTIETRANSFO (400 OHM ) TE VOORZIEN ZOALS HET SCHEMA AANGEEFT , MEN GEBRUIKT PB7 ALS SENSE POINT VOOR DE BELSTROOM EN PB0,PB1 EN PB2 ALS OUTPUT  
DE JUNIOR IS VOORZIEN VAN 2 CASSETTERECORDERS EN DEZE GEBRUIKT MEN RESP. VOOR WEERGAVE EN OPNAME.  
DE AUDIO VERBINDING VAN BEIDE WORDT AAN DE MINI-INTERFACE GESCHAKELD TERWIJL IN DE OPNAME RECORDER EEN LEGE CASSETTE WORDT GEZET EEN CASSETTE MET MEERDERE MALEN EEN OP VOORHAND OPGENOMEN BERICHT WORDT IN DE WEERGAVEREORDER GEPLAATST EN START HET PROGRAMMA OP 0200 BELKADANS ,WEERGAVE EN OPNAMETIJD KUNNEN GEWIJZIGD WORDEN DOOR IN PAG. 0 ANDERE WAARDEN TE PLAATSEN 004 IS ONG. 1 SEC.

TELEF ORG 0200

**PAGINA 0 ADRESSEN**

10 00	RINGK *	0010	TIJD TUSSEN 2 BELSTROMEN
11 00	RESPT *	0011	ANTWOORDTIJD
12 00	RECOT *	0012	OPNAMETIJD
13 00	COUNT *	0013	TIJDELIJKE COUNTER

**POORT ADRESSEN**

00 18	PBD *	1800	DATA REG. VIA 6522
02 18	PBDD *	1802	DATA DIRECTION REG. VIA 6522
82 1A	RBD *	1A82	DATA REG. RIOT 6532
83 1A	RBDD *	1A83	DATA DIRECTION REG. RIOT 6532

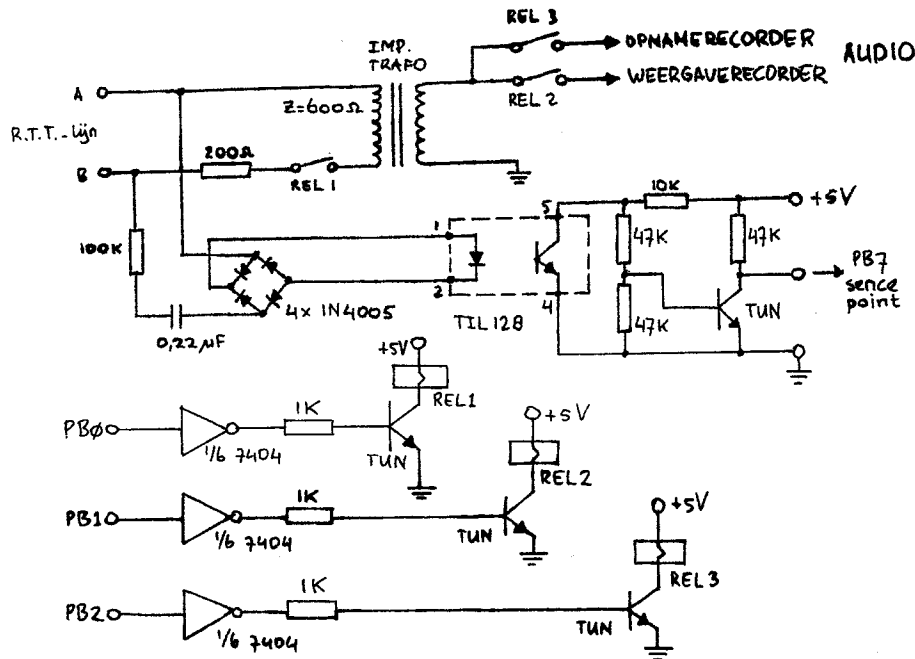
INIT	LDAIN \$10	ZET TIJDEN
	STAZ RINGK	RINGKADANS = 4 SEC
	LBAIN \$50	
	STAZ RESPT	ANTWOORDTIJD = 20 SEC
	LBAIN \$60	
	STAZ RECOT	OPNAMETIJD = 30 SEC
	LDAIN \$FF	
	STA RBDD	ZET ALLE POORTEN OP

0590:	0211 8D 82 1A	STA	RBD	UITGANG IN RIOT
0600:	0214 A9 7F	LDAIN	\$7F	ZET PBO TOT PB6 ALS UITGANG
0610:	0216 8D 02 18	STA	PBDD	EN PB7 ALS INGANG IN VIA
0620:				
0630:				
0640:				
0650:				
0660:	0219 A9 00	MAIN	LDAIN \$00	CLEAR COUNTER
0670:	021B 85 13		STAZ COUNT	
0680:	021D AD 00 18		LBA PBD	LEES INGANG
0690:	0220 29 80		ANDIN \$80	ENKEL BIT 7
0700:	0222 C9 80		CHPIN \$80	IS DEZE EEN
0710:	0224 F0 29		BEG RITWO	GA NAAR TWEDE BELSTROOM
0720:	0226 A9 FF	CLEAR	LDAIN \$FF	ZONIET RESET POORTEN
0730:	0228 8D 00 18		STA PBD	
0740:	022B 8D 82 1A		STA RBD	
0750:	022E A9 01		LDAIN \$01	
0760:	0230 85 13		STAZ COUNT	WACHT ONG. 0,25 SEC
0770:	0232 20 38 02		JSR DELAY	
0780:	0235 4C 19 02		JMP MAIN	EN GA TERUG NAAR BEGIN
0790:				
0800:				
0810:				
0820:				
0830:	0238 A2 C5	DELAY	LDXIN \$C5	
0840:	023A A0 D2	LOOPA	LDYIN \$D2	
0850:	023C 88	LOOPB	DEY	
0860:	023D C0 00		CPYIN \$00	
0870:	023F D0 FB		BNE LOOPB	
0880:	0241 CA		DEX	
0890:	0242 E0 00		CPXIN \$00	
0900:	0244 B0 F6		BNE LOOPB	
0910:	0246 C6 13		DECZ COUNT	
0920:	0248 A5 13		LBAZ COUNT	
0930:	024A C9 00		CHPIN \$00	
0940:	024C B0 EA		BNE DELAY	
0950:	024E 60		RTS	
0960:				
0970:				
0980:				
0990:				
1000:				
1010:	024F A5 10	RITWO	LDAZ RINGK	
1020:	0251 85 13		STAZ COUNT	WACHT TOT
1030:	0253 20 38 02		JSR DELAY	VOLGENDE BELSTROOM
1040:	0256 AD 00 18		LBA PBD	IS PB7 OPNIEU HOOG ?
1050:	0259 29 80		ANDIN \$80	
1060:	025B C9 80		CHPIN \$80	
1070:	025D D0 BA		BNE MAIN	ZONIET NAAR BEGIN
1080:	025F A9 FC		LDAIN \$FC	ANDERS SLUIT LUS
1090:	0261 8D 00 18		STA PBD	
1100:	0264 A9 BF		LDAIN \$BF	EN ZET ANTWOORDRECORDER
1110:	0266 8D 82 1A		STA RBD	
1120:	0269 A5 11		LDAZ RESPT	WACHT ANTWOORDTIJD
1130:	026B 85 13		STAZ COUNT	
1140:	026D 20 38 02		JSR DELAY	

\* DELAY ROUTINE \*

\* TWEDE BELSTROOMCHECK \*

1150: 0270 A9 FA	LBAIN 9FA	SWITCH OVER NAAR
1160: 0272 8D 00 18	STA PBD	
1170: 0275 A9 BF	LBAIN 8DF	OPNAHERECORDER
1180: 0277 8D 82 1A	STA RBD	
1190: 027A A5 12	LBAZ RECOT	
1200: 027C 85 13	STAZ COUNT	WACHT OPNAMETIJD
1210: 027E 20 38 02	JSR DELAY	
1220: 0281 4C 26 02	JMP CLEAR	



## MICRO - ADE ASSEMBLER, DIS-ASSEMBLER EN TEXT-EDITOR.

De Micro-Ade, waarvan sinds enige tijd onze club de copyrights heeft verworven, bestaat uit drie hoofdprogramma's en een aantal utility-programma's. De hoofdprogramma's zijn een assembler, een disassembler, en een tekst-editor. De minimum systeemconfiguratie bestaat uit een 6502 computer met minimaal 16K RAM, 1 of 2 cassetterecorders met of zonder start/stop-control en een ASCII input/output device.

Voor gebruikers van andere computers dan KIM of JUNIOR is de code niet op band te verkrijgen; zij kunnen het beste het manual en source van MICRO-ADE bestellen en het geheel aan hun computer aanpassen.

NIEUW:  
8K JUNIOR SOURCE LISTING  
FL. 65,00

Cassette 4K KIM (hypertape)	12,50
Cassette 8K KIM (hypertape)	12,50
Cassette 8K JUNIOR (hypertape)	12,50
Manual	30,00
Source 4K KIM	30,00
Patches S.Woldringh (4K KIM wordt 8K KIM)	75,00
Patches a.Hanke! (alleen bij cassette voor JUNIOR + Manual)	gratis

Aanschaf van source en patches is niet perse direct noodzakelijk; men kan volstaan met cassette + manual + patches S.Woldringh dl.10 (zie De 6502 Kenner nr. 17)

Patches Dl. 10 apart 5,00

Overschrijving op postrekening 841433 t.n.v. W.L. van Peit te Krimpen a.d. IJssel, of d.m.v. blauwe girobetaalkaart.  
Geen andere wijze van bestellen mogelijk!

0510: B.I.J. DE JUNIOR, VOORZIEN VAN DE UITBREIDINGS KAART, IS  
 0520: DE 6522 PIA BEREIKBAAR OP PAGINA \$18 EN \$19.  
 0530: DEZE DUBBEL ADDRESSERING WORDT VOORKOMEN DOOR DE HARDWARE-  
 0540: WIJZIGINGEN VAN FIGUUR 2.  
 0550: DE PIA IS DAN ALLEEN TE BEREIKEN IN PAGINA \$18.  
 0560: PAGINA \$19 IS NU BESCHIKBAAR VOOR DE I/O.

0570: HARDWARE WIJZIGINGEN I/O KAART.  
 \*\*\*\*\*  
 0630: OP DE JUNIOR I/O KAART WORDT PEN 3 VAN IC15 (N40)  
 DOORGESNEDEN TEN OPZICHTE VAN DE +5 VOLT.  
 0650: OP PEN 3 WORDT NU HET CS SIGNAAL VAN DE I/O  
 AANGESLOTEN. DE DUBBELADDRESSERING VAN DE VIA WORDT  
 0670: UITGESCHAKELD DOOR PEN 13 VAN IC13 (N35) LOS TE SNIJDEN  
 0680: VAN MASSA EN DOOR TE VERBINDEN MET A88.  
 0690: DOOR DEZE INGREEP IS DE PIA ALLEEN NOG WERKZAAM IN  
 0710: PAGINA \$18.  
 0720: VERDER WORDEN OP DE I/O KAART DE VERBINDINGEN VAN DE  
 0730: SERIELE INTERFACE NAAR P80 EN P87 LOSGEKOPPELD VAN  
 0740: DE 5332 VIA OP DE BASISKAART.  
 0750: DE SERIELE INTERFACE KAN NU WORDEN GEBRUIKT VOOR DE  
 0760: I/O SCHAKELING. VERDER IS IN FIGUUR 2 AANGEGEVEN  
 0770: HOE DE AANSLUITINGEN P EN Q OP DE I/O KAART GEBRUIKT  
 0780: KUNNEN WORDEN VOOR EEN FULL/HALF DUPLEX MOGELIJKHEID.

0790: DE I/O SCHAKELING VAN FIGUUR 1 IS OPGEBOUWD ROND DE  
 0810: AY-3-1015. DIT IC IS NAGENOEG IDENTIEK AAN DE IN DE  
 0820: ELEKTRONICA TOEGEPASTE AY-3-1013 OF MM 5303.  
 0830: HET ENIGE VERSCHIL IS DAT DE AY-3-1015 ALLEEN EEN  
 0840: VOEDINGSSPANNING VAN +5 VOLT NODIG HEEFT.  
 0850: IN FIGUUR 3 ZIJN DE AANSLUITINGEN VAN DE  
 0860: AY-3-1015 AANGEGEVEN. DE UITGANGEN VAN HET IC ZIJN  
 0870: TTL VERENIGBAAR, HETGEEN WIL ZEGGEN DAT PER UITGANG  
 0880: 1 TTL INGANG GESTUURD MAG WORDEN.

0890: DE PENNEN 5-12, DE ONTVANGEN DATA, EN DE PENNEN  
 0900: 13, 14, 15, 19 EN 22 VAN HET STATUSWOORD,  
 0910: BEZITTEN TRI-STATE UITGANGEN, HIERDOOR IS HET MOGELIJK  
 0920: DEZE UITGANGEN DIRECT OP DE DATABUS VAN EEN COMPUTER-  
 0930: SYSTEEM AAN TE SLUITEN. DE BESTURING VAN DE UITGANGEN  
 0940: VINDT DAN PLAATS MET DE PENNEN 16 (STATUSWOORD ENABLE)  
 0950: EN 4 (DATA ENABLE). DE TE VERZENDEN DATA OP DE PENNEN  
 0960: 26-33, WORDT INGEKLOKT EN VERZONDEN MET BEHULP VAN  
 0970: EEN DATA STROBEPULS OP PEN 23. INDIEN ER GEEN DATA  
 0980: STROBEPULS WORDT GEGEVEN DAN MAG DE DATA OP DE  
 0990: INGANGEN 26-33 WILLEKEURIG ZIJN, SERIEEL GEBEURT

1000: 1000: INGANGEN 26-33 WILLEKEURIG ZIJN, SERIEEL GEBEURT

I/O SCHAKELING VOOR JUNIOR  
 \*\*\*\*\*  
 A. A. ZWART, GOOR

DE MEESTE SINGLE BOARD- EN PERSONAL COMPUTERS ZIJN  
 VOORZIEN VAN EEN SERIELE TRANSMISSIE AANSLUITING.  
 DEZE AANSLUITING WERKT MEESTAL MET BEHULP VAN  
 SOFTWARE DRIVERS WELKE ONDER ANDERE BEPALEN:  
 - SNELHEID VAN DE TRANSMISSIE  
 - AANTAL BITS  
 - BIJVOORBEELD DATA- START- EN STOPBITS  
 - VOORBEELDEN VAN SERIELE TRANSMISSIE AANSLUITINGEN  
 - ZIJN RS232 EN RS232-C-20 MA CURRENTLOOP.

HET NADEEL VAN SOFTWARE INTERFACES, ZOALS BIJVOORBEELD  
 TOEGEPAST IN HET JUNIOR SYSTEEM, IS DAT DE SNELHEID VAN  
 DE TRANSMISSIE NIET EENVOUDIG VERANDERD KAN WORDEN  
 0190: EN DAT DE SNELHEID MEESTAL BEPERKT IS.  
 0200: TEVENS WORDT DOOR DE SOFTWARE BEPAALD HOEVEEL DATA-BITS  
 , STOPBITS, PARITEITSBITS, CODESYSTEEM EN DERGELIJKE  
 0220: WORDT OF IS INGESTELD.

DE MEESTE BEELDSCHERM EENMEDIEN ZIJN VOORZIEN VAN EEN  
 I/O SCHAKELING WELKE DE SERIELE TRANSMISSIE REBELT.  
 HET LIGT DAN VOOR DE HAND OM DE COMPUTER EVENEENS TE  
 VOORZIEN VAN EEN I/O INTERFACE SCHAKELING.

DE HIER BESCHREVEN SCHAKELING HEEFT DE VOLGENDE  
 EIGENSCHAPPEN:  
 - MEMORY MAPPED  
 - BAUDRATE 75 - 9600 BAUD, EVENTUEEL VANAF  
 50 BAUD TEN BEHOEVE VAN EEN TELEX  
 - AANTAL DATABITS INSTELBAAR  
 - PARITEITS KEUZE INSTELBAAR  
 - STATUSWOORD VOOR CONTROLE VAN DE INTERFACE  
 EN DE TRANSMISSIE  
 - EENVOUDIGE SOFTWARE DRIVERS VOOR  
 ONTVANGEN EN ZENDEN  
 - RELATIEF GOEDKOOPELIJK, DAAR GEEN  
 DUUR ONDERDELEN ZIJN TOEGEPAST.  
 DE PRIJS VAN HET I/O IC BEDRAAGT ONGEVEER  
 F15 = TOT F25, =

DE SCHAKELING IS MEMORY MAPPED, DIT BETEKENT DAT HET  
 STATUSWOORD, DE TE ONTVANGEN DATA EN TE VERZENDEN DATA  
 OP DRIE GEHEUGENPLAATSEN IN HET COMPUTER GEHEUGEN TE  
 LEZEN OF SCHRIFVEN IS.  
 DIT VINDT DAN PLAATS OP EENZELFDE MANIER ALS DE  
 ADDRESSERING VAN EEN VIA OF PIA.

0010:  
 0020:  
 0030:  
 0040:  
 0050:  
 0060:  
 0070:  
 0080:  
 0090:  
 0100:  
 0110:  
 0120:  
 0130:  
 0140:  
 0150:  
 0160:  
 0170:  
 0180:  
 0190:  
 0200:  
 0210:  
 0220:  
 0230:  
 0240:  
 0250:  
 0260:  
 0270:  
 0280:  
 0290:  
 0300:  
 0310:  
 0320:  
 0330:  
 0340:  
 0350:  
 0360:  
 0370:  
 0380:  
 0390:  
 0400:  
 0410:  
 0420:  
 0430:  
 0440:  
 0450:  
 0460:  
 0470:  
 0480:  
 0490:  
 0500:

1010: ER NIETS, DE ENIGE PEN DIE GEEN TAX-STATE UITGANG  
 1020: HEEFT, IS PEN 24 (EINDE KARAKTER).  
 1030:  
 1040: PEN 21, RESET, KAN VIA EEN INVERTER WORDEN AAN-  
 1050: GESLOTEN OP HET RESETSIGNAL VAN DE JUNIOR.  
 1060: NOODZAKELIJK IS DIT NIET, INDIEN MEN HET RESET-  
 1070: SIGNAL NIET TOEPAST, DAN DIENT PEN 21 AAN DE  
 1080: MASSA TE WORDEN GELEED.  
 1090:

1100: UART INSTELLINGEN.  
 1110: \*\*\*\*\*

1120:  
 1130: INSTELLING VAN DE PARITEITS KEUZE KAN MET DE DIP-  
 1140: SCHAKELAARS OP DE PENNEN 35 EN 39 WORDEN INGESTELD.

PEN 35	PEN 39	PARITEIT
0	0	ONEVEN
0	1	EVEN
1	X	EEN

X = MAAKT NIET UIT.

1210:  
 1220:  
 1230: HET AANTAL DATABITS PER KARAKTER WORDT INGESTELD OP DE  
 1240: PENNEN 37 EN 38.

NB1 PEN 38	NB2 PEN 37	AANTAL DATABITS
0	0	5
1	0	6
0	1	7
1	1	8

1290:  
 1300:  
 1310:  
 1320:  
 1330:  
 1340: HET AANTAL STOPBITS WORDT INGESTELD OP PEN 36.

PEN 36	AANTAL STOPBITS
0	2
1	1

1350:  
 1360:  
 1370:  
 1380:  
 1390:  
 1400:  
 1410: DE BESCHREVEN PARAMETERS WORDEN IN EEN INTERN  
 1420: PARAMETERREEDSTER OPGESLAGEN ALS DE CONTROLSTROBE,  
 1430: PEN 34, HOOG IS. DE CONTROLSTROBE IS HOOG ZOLANG DE  
 1440: UART SCHAKELING GEBELEED IS. DIT WORDT GEBEELD  
 1450: DOOR HET ADRESDECODER DEEL GEVORMD DOOR N1, N3-N5.  
 1460: DE ADRESDECODERING VINDT PLAATS MET IC2 EN N1-N5.  
 1470: K6, AFGKOMSTIG VAN DE DECODER OP DE JUNIOR BASISKAART,  
 1480: BEPAALT DAT HET ADRESGEBIED ZICH BEVINDT IN HET  
 1490: BEREIK \$1800 - \$18FF. A0, A1, A2, A8 EN A9 BEKEREN  
 1500: HET ADRESGEBIED TOT DE GEDURENPLAATSEN \$1900, \$1901 EN

1510: \$1902 MET:  
 1520:  
 1530: \$1900 = ZENDBUFFER, TE VERZENDEN DATA  
 1540: \$1901 = ONTVANGEN DATA  
 1550: \$1902 = STATUSWOORD  
 1560:  
 1570: HET RAM R/W SIGNAL VAN DE JUNIOR WORDT GEBRUIKT VOOR  
 1580: DE NOODZAKELIJKE TIMING VAN DE UART SIGNALEN OP DE  
 1590: DATABUS.

1600:  
 1610: ZENDEN VAN KARAKTERS.  
 1620: \*\*\*\*\*

1630:  
 1640: HET DATASTROBE SIGNAL WORDT GEVORMD DOOR RAM R/W,  
 1650: EINDE KARAKTER EN UITGANG 0 VAN IC2. ER IS EEN SIMPEL  
 1660: SET-RESET FLIPELOP TOEGEPAST (N14/N15) OM HET DATA  
 1670: STROBESIGNAL TE VOORMEN. INDIEN DATA WORDT BESCHREVEN  
 1680: NAAR ADRES \$1900 DAN WORDT, ALS RAM R/W LAAG IS, EEN  
 1690: NEERGAANDE FLANK OP PEN 23 GEGEVEN. RANGEZIEN DE DATABUS  
 1700: DAN GESTABILISEERD IS, WORDT DE TE VERZENDEN DATA IN  
 1710: HET ZENDBUFFER INGEKLOKT. OP HET MOMENT DAT DE GEDUREN-  
 1720: SELECTIE VAN ADRES \$1900 MEGVALT, DAN WORDT OP PEN 23  
 1730: EEN OPGANDE FLANK GEGEVEN, OP DIT MOMENT START DE UART  
 1740: DE SERIELE TRANSMISSIE VAN HET EERDE INGEKLOKTE  
 1750: KARAKTER. HET EINDE KARAKTER SIGNAL OP PEN 24 IS DAN 0  
 1760: GEGEVEN TEN TEKEN DAT DE TRANSMISSIE GAANDE IS.  
 1770: HET EINDE KARAKTERSIGNAL ZORGT ER SAMEN MET N13 VOOR  
 1780: DAT GEEN OPGANDE DATASTROBE SIGNAL OP PEN 23  
 1790: GEGEVEN VAN WORDEN ZOLANG HET KARAKTER NOG NIET GEHEEL  
 1800: VERZONDEN IS EN DE TRANSMISSIE NOG NIET BEEINDIGD IS.  
 1810: DE UITGANG VAN N13 IS NAAR HET STATUSWOORD GEVOERD OM  
 1820: TE KUNNEN ONDERZOEKEN OF DE SERIELE TRANSMISSIE GEHEEL  
 1830: GEREED IS. OP HET MOMENT DAT ZENDBUFFER LEEG VAN HET  
 1840: STATUSWOORD ANGEDEFT DAT ER EEN NIEUW KARAKTER IN HET  
 1850: ZENDBUFFER KAN WORDEN GELADEN, IS HET ZENDEN NOG NIET  
 1860: VOLLEDIG GEREED, ER DIENEN DAN NOG STOPBITS EN EVENTUEEL  
 1870: PARITEITSBITS DOOR DE UART TE WORDEN VERZONDEN.  
 1880: DEZE TJD KAN DAN WORDEN GEBRUIKT OM ALVAST EEN NIEUW  
 1890: KARAKTER IN HET ZENDBUFFER TE LADEN MIDDELS EEN  
 1900: NEERGAANDE FLANK VAN HET DATA STROBESIGNAL. OP HET  
 1910: MOMENT DAT EINDE KARAKTER ANGEDEFT DAT HET VORIGE  
 1920: KARAKTER VOLLEDIG IS VERZONDEN, INCLUSIEF STOP- EN  
 1930: EVENTUELE PARITEITSBITS, DAN WORDT EEN OPGANDE FLANK  
 1940: GEGENEREERD, DOOR DE SCHRIFTOPERATIE NAAR \$1900 WAS  
 1950: AL EEN NEERGAANDE FLANK GEGENEREERD EN HET ZENDBUFFER  
 1960: BELADEN, MET DE OPGANDE FLANK START DAN DE VERZENDING  
 1970: VAN HET NIEUWE KARAKTER.  
 1980:  
 1990:  
 1990:

2000: ONTVANGEN VAN DATA.  
 2010: \*\*\*\*\*  
 2020: OP HET MOMENT DAT ADRES \$1501 WORDT GELEZEN, WORDT ER  
 2030: DOOR RAM R/W, N6 EN N8 VOOR GEZORGD DAT, INDIEN  
 2040: RAM R/W HOOG IS, PEN 4 (READ DATA ENABLE) LAAG WORDT.  
 2050: NIJ WORDT DE ONTVANGEN DATA OP DE DATABUS GEZET.  
 2060: DIT HEEFT ALLEEN ZIJN ALS EEN KARAKTER ONTVANGEN IS.  
 2070: MET BIT 7 (DATA BESCHIKBAAR) VAN HET STATUSWOORD  
 2080: KAN WORDEN GEDONTROLEERD OF DAT HET GEVAL IS.  
 2090: HET WORDEN SIGNAAL, PEN 18 EN HIER VERBODEN MET DE  
 2100: SERIE INGAANG, RESET HET HET DATA BESCHIKBAAR SIGNAAL  
 2110: OP PEN 19, IN EEN ANDERE TOEPASSING DAN HIER  
 2120: BESCHREVEN, KAN PEN 18 WORDEN BESTUURD DOOR  
 2130: BIJVOORBEELD EEN ANTWOORDSIGNAAL VAN HET RAND-  
 2140: APPARAAT. DEZE KAN DAN OP PEN 18 AANGEVEN DAT HET  
 2150: ONTVANGEN KARAKTER IS VERMERKT.  
 2160: ALS RDW, PEN 18, PERMANENT LAAG IS, ZAL DATA  
 2170: BESCHIKBAAR, PEN 19, GEDURENDE EEN KLOKPERIODE  
 2180: VAN DE ONTVANGERKLOK HOOG ZIJN NA DE ONTVANGST VAN  
 2190: EEN KARAKTER.  
 2200: SCHRIJVEN VAN DATA NAAR ADRES \$1501 HEEFT GEEN ZIJN,  
 2210: DAAR HET OM EEN UITGANGSREGISTER GAAT. MOGELIJKE  
 2220: "SCHADE" WORDT DAN VOORKOMEN DOOR RAM R/W EN N8.  
 2230: PEN 4 WORDT DAN NIET LAAG ZOLANG RAM R/W LAAG  
 2240: (SCHRIJFOPERATIE) IS. DE DATAUITGANGEN VAN DE UART  
 2250: BLIJVEN DAN IN TRI-STATE.  
 2260: STATUSWOORD.  
 2270: \*\*\*\*\*  
 2280: OP DEZELFDE WIJZE ALS BIJ HET ONTVANGEN VAN DATA  
 2290: BESCHREVEN, ZORGEN RAM R/W, N7 EN N9 ERVOOR DAT  
 2300: HET STATUSREGISTER ALLEEN KAN WORDEN GELEZEN.  
 2310: OP ADRES \$1502.  
 2320: \*\*\*\*\*  
 2330: OM EINDE KARAKTER IN HET STATUSWOORD TE KRIJGEN, IS BUFFER  
 2340: N17 OPGENOMEN, INDIEN VAN DE TOEGEPASTE BEELDSCHEM-  
 2350: EENHEID EEN TERMINAL READY SIGNAAL BESCHIKBAAR IS,  
 2360: DAN KAN DIT WORDEN AANGESLOTEN OP N18, INDIEN DIT  
 2370: SIGNAAL NIET BESCHIKBAAR IS, DAN KAN DE INGAANG VAN  
 2380: N18 NAAR BELIEVEN AAN DE +5 VOLT DE DE MASSA WORDEN  
 2390: GELEID, DE AANSLUITINGEN VOOR HET STATUSWOORD OP  
 2400: DE DATABUS KUNNEN DAN ZIJN:  
 2410: \*\*\*\*\*  
 2420: \*\*\*\*\*  
 2430: \*\*\*\*\*  
 2440: \*\*\*\*\*  
 2450: \*\*\*\*\*  
 2460: \*\*\*\*\*  
 2470: \*\*\*\*\*  
 2480: \*\*\*\*\*  
 2490: \*\*\*\*\*  
 2500: \*\*\*\*\*

2510: FOUT  
 2520: OVERFLOW  
 2530: TERMINAL READY  
 2540: EINDE KARAKTER  
 2550: VIA N13 EN N17  
 2560: NIET GEDRUUKT  
 2570: ZENDBUFFER LEEG  
 2580: DATA BESCHIKBAAR  
 2590: \*\*\*\*\*  
 2600: \*\*\*\*\*  
 2610: \*\*\*\*\*  
 2620: \*\*\*\*\*  
 2630: \*\*\*\*\*  
 2640: \*\*\*\*\*  
 2650: \*\*\*\*\*  
 2660: \*\*\*\*\*  
 2670: \*\*\*\*\*  
 2680: \*\*\*\*\*  
 2690: \*\*\*\*\*  
 2700: \*\*\*\*\*  
 2710: \*\*\*\*\*  
 2720: \*\*\*\*\*  
 2730: \*\*\*\*\*  
 2740: \*\*\*\*\*  
 2750: \*\*\*\*\*  
 2760: \*\*\*\*\*  
 2770: \*\*\*\*\*  
 2780: \*\*\*\*\*  
 2790: \*\*\*\*\*  
 2800: \*\*\*\*\*  
 2810: \*\*\*\*\*  
 2820: \*\*\*\*\*  
 2830: \*\*\*\*\*  
 2840: \*\*\*\*\*  
 2850: \*\*\*\*\*  
 2860: \*\*\*\*\*  
 2870: \*\*\*\*\*  
 2880: \*\*\*\*\*  
 2890: \*\*\*\*\*  
 2900: \*\*\*\*\*  
 2910: \*\*\*\*\*  
 2920: \*\*\*\*\*  
 2930: \*\*\*\*\*  
 2940: \*\*\*\*\*  
 2950: \*\*\*\*\*  
 2960: \*\*\*\*\*  
 2970: \*\*\*\*\*  
 2980: \*\*\*\*\*  
 2990: \*\*\*\*\*  
 3000: \*\*\*\*\*

0B1  
 0B2  
 0B3  
 0B4  
 0B5  
 0B6  
 0B7

- PEN 15, OVERFLOW  
 - PEN 24, EINDE KARAKTER  
 - PEN 22, ZENDBUFFER LEEG  
 - PEN 19, DATA BESCHIKBAAR

PARITEITSFOUT : 1 = FOUT  
 SYNCHRONISATIE FOUT : 1 = FOUT (AANTAL STOPBITS KLOPT NIET)

OVERFLOW : 0 = GOED  
 1 = OVERFLOW  
 0 = GOED  
 1 = GOED

HET OVERFLOWBIT IS HIER NIET ACTIEF TOEGEPAST. PEN 18 IS DOORVERBODEN MET SERIE IN, OP PEN 18 KAN EEN SIGNAAL VAN HET RANDAPPARAAT AANGESLOTEN WORDEN WELKE AANGEeft DAT HET DOOR HET RANDAPPARAAT ONTVANGEN KARAKTER VERMERKT IS.

TERMINAL READY : 1 = NOG NIET GEREED  
 EINDE KARAKTER : 0 = GEREED EN ALAAR OM TE ZENDEN. LEEG, DE DATASTROBE MAG EEN NEGATIEF GAANDE FLANK GEVEN WAARDOOR DE TE VERZENDEN DATA WORDT INGEKLOKT IN HET ZENDBUFFER.  
 DATA BESCHIKBAAR : 1 = GEREED INDIEN ER EEN VOLLEDIG KARAKTER IS ONTVANGEN. GERESET DOOR HET STARTBIT VAN EEN SERIEEL ONTVANGEN KARAKTER.

ONTVANGER- EN ZENDERKLOK.  
 \*\*\*\*\*

DE KLOK IS DE GRONDMAKER VOOR DE UART. HET ALONGSIGNAAL DIEN'T TE WORDEN AANGESLOTEN OP DE PENNEN 40 EN 17 VAN DE UART. DE UART KAN EVENTUEEL MET EEN ANDERE Snelheid KARAKTERS VERZENDEN DAN ONTVANGEN. PEN 17 IS DE AANSluitING VOOR DE ZENDERKLOK. IN DE HIER BESCHREVEN TOEPASSING WORDT EVEN Snel VERZONDEN ALS ONTVANGEN, VANDAAR DAT ZENDER EN ONTVANGER HETZELFDE ALONGSIGNAAL KUNNEN AANGESLOTEN. DE KLOKPERIODE VOOR DE UART MOET 16 MAAL DE GEMENSTE BALDRATE BEDRAGEN. EEN EITTIJD DUURT NAmELLIJK 16 PERIODEN VAN DE KLOK.

UART  
 DATABUS  
 - PEN 13, PARITEITSFOUT  
 - PEN 14, SYNCHRONISATIE

3010: DE 16 PERIODEN ZIJN MOOZAKELIJK VOOR DE UART OM DE  
 3020: SERIELE TRANSMISSIE TE KUNNEN CONTROLEREN EN OM  
 3030: DE DATA IN TE KLOKKEN IN HET ZENDBUFFER VOORZIEN VAN  
 3040: START-, STOP- EN PARITEITSBITS.  
 3050: VOOR DE ONTVANGER GELDT, BIJ AANVANG VAN ONTVANGST,  
 3060: INDIEN NA 8 KLOKPULSEN DE INANG NOG STEEDS LANG IS,  
 3070: ER EEN GELDIG STARTBIT IS ONTVANGEN, 16 PULSEN DAARNA  
 3080: WORDT HET EERSTE DATABIT BINNEN GEHAALD, DIT GAAT ZO  
 3090: DOOR TOT HET COMPLETE BITPATTERN IS BINNENGEHAALD  
 3100: INCLUSIEF EEN CONTROLE OP PARITEITS- EN STOPBITS.

DIT BETEKENT :

BAUDRATE (BAUD)	UART-KLOK (KHZ)
75	1,2
110	1,75
150	2,4
300	4,8
600	9,6
1200	19,2
2400	38,4
4800	76,8
9600	153,6

3270: INDIEN HET KLOKSIGNAAL NIET BESCHIKBAAR IS VAN HET  
 3280: RANDAPPARAAT, DAN KAN HET EVENTUEEL WORDEN BEVORMD  
 3290: DOOR EEN KLOKGENERATOR ZOALS AANGEGEVEN IN FIGUUR 1.  
 3300: MET BEHULP VAN P1 KAN DE FREQUENTIE WORDEN INGESTELD.  
 3310: IN DE HANDEL ZIJN COMPLETE ENKEL CHIP'S BAUDRATE-  
 3320: GENERATOREN VERKRIJGBAAR, DIE IC OF KRISTAL GESTUURD  
 3330: WORDEN, EEN WADEFEL VAN DEZE CHIP'S IS DE RELATIEF  
 3340: HOGE PRIJS. EEN BAUDRATEGENERATOR KAN OOK WORDEN  
 3350: OPGEBUILD MET BEHULP VAN STANDAARD TTL IC'S, WAARBIJ  
 3360: HET SYSTEEMKLOKSIGNAAL &2 VAN DE JUNIOR WORDT GEBRUIKT.  
 3370: HIER ZAL NIET VERDER OP WORDEN INGEAAN.  
 3380: WEL DIENT ER OP TE WORDEN GELET DAT DE FREQUENTIE  
 3390: REDELIJK NAUWKEURIG INSTELBAAR IS, OM DE GEDEFINEERDE  
 3400: BAUDRATE'S REDELIJK STABIEL TE VERMAGLIGEN.  
 3410: HET IS MOGELIJK OM DE UART MET 50 BAUD TE LATEN  
 3420: WERKEN EN ZO BIJVOORBEELD EEN TELEX ALS PRINTER AAN  
 3430: TE STUREN. IN DE SOFTWARE DRIVERS DIENEN DAN WEL  
 3440: DE BAUDTAKTARAKTERS TE WORDEN BEPAALD, HET AANTAL  
 3450: DATABITS BEDRAAGT DAN 5 IN PLAATS VAN 6 OF 7 ZOALS  
 3460: BIJ DE ASCII CODE.  
 3470: INDIEN HET KLOKSIGNAAL VAN DE UART, VAN EEN UART IN HET  
 3480: RANDAPPARAAT KAN WORDEN BETROKKEN, DAN KAN BIJVOORBEELD  
 3490: TIJDENS EEN LISTING, DE SNELHEID MET EEN BAUDRATE-

3510: SCHAKELBAAR NAAR BELIEVEN WORDEN VERANDERD OMDAT DAN DE  
 3520: BEIDE UARTS MET GELIJKE SNELHEID WERKEN, INDIEN HET KLOK-  
 3530: SIGNAAL WORDT STOPGEZET, DAN STOPT OOK DE TRANSMISSIE  
 3540: NAAR HET RANDAPPARAAT.  
 3550: OP DEZE WIJZE KAN DAN HET BEELD VAN EEN  
 3560: BEELDSCHERM EENHEID WORDEN BEVOREN, ZODAT EEN BEPAALD  
 3570: STUK TEKST OF OUTPUT RUSTIG BEKEKEN KAN WORDEN.  
 3580: DE IN DIT ARTIKEL BESCHREVEN SOFTWAREDRIVERS ZORGEN ER  
 3590: DAN VOOR DAT DE TRANSMISSIE VERDER GAAT ZODRAN  
 3600: ER WEER KLOKSIGNAAL WORDT TOEGEVOERD.  
 3610: DE BESCHREVEN SOFTWARE VOOR DE LEES- EN SCHRIJF ROUTINES  
 3620: KUNNEN WORDEN GEBRUIKT IN PLAATS VAN DE JUNIOR ROUTINES  
 3630: RECCHA EN PRCHA.  
 3640: BIJ DE JUNIOR KUNNEN DE DIVERSE BENODIGDE SIGNALLEN  
 3650: ZOALS K6, CS, KLOK, SERIE IN EN SERIE UIT, WORDEN  
 3660: AANGESLOTEN OP DE NIET GEBRUIKTE AANSLUITINGEN VAN DE  
 3670: 64 POLIGE CONNECTORS EN ZO DOOR HET SYSTEEM WORDEN GELEID.

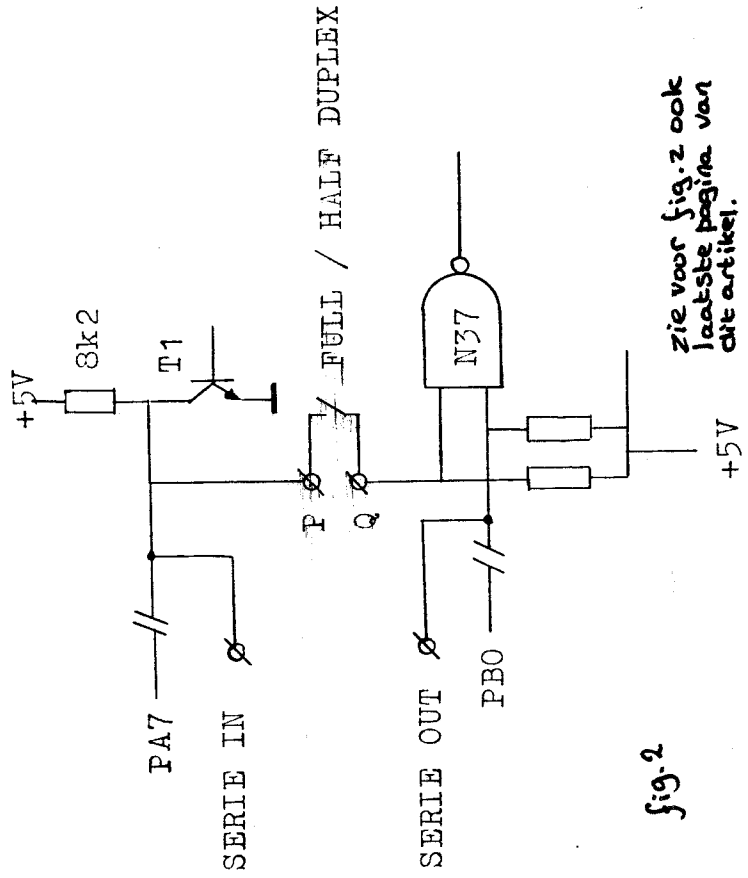


fig-2 zie voor fig.2 ook laatste pagina van dit artikel.

0510: 9033 88 DEY INDEX=INDEX - \$01  
 0520: 9034 00 E7 MELD  
 0530: 9035 5C 7A 1A BRKVEC SPRING NAAR BREAKADRES

0010: 9000 9000 ORG \$9000  
 0020: 9001 20 39 UART PHA  
 0030: 9002 00 22 UARTA JSR  
 0040: 9003 2C 02 19 BIT BCS  
 0050: 9004 50 F6 8VC JSR  
 0060: 9005 20 20 8VC BCS  
 0070: 9006 00 18 8VC LDRAM \$10  
 0080: 9007 20 02 19 AND STATUS  
 0090: 9008 10 F4 BNE ENDRCH  
 0100: 9009 00 0C JSR BREAK  
 0110: 9010 00 08 BCS ENDA  
 0120: 9011 20 02 19 AND STATUS  
 0130: 9012 00 F4 BNE START  
 0140: 9013 68 PLA  
 0150: 9014 80 00 STA  
 0160: 9015 60 FF RTS  
 0170: 9016 00 9A LDXIM \$FF  
 0180: 9017 00 09 TXS  
 0190: 9018 00 09 LDYIM \$09  
 0200: 9019 41 90 LDARY TABEL  
 0210: 9020 20 00 JSR UART

\* SUBROUTINE BREAK \*  
 ONDERZOEK OF BREAKKEY  
 IS INGEDRUKT.

BREAK CLC  
 BIT STATUS DATA BESCHIKBAAR ?  
 BNE ENDB  
 SEC BREAKKEY INGEDRUKT  
 RTS ENDB

TABEL = \$0D  
 = \$0A  
 = -  
 = 'B  
 = 'R  
 = 'K  
 = -  
 = \$0D  
 = \$0A

\*\* SUBROUTINE OM EEN ASCII \*\*  
 \*\* KARAKTER NAAR DE UART TE \*\*  
 \*\* STUREN.

DEZE SUBROUTINE KAN WORDEN  
 GEBRUIKT IN PLAATS VAN DE  
 JUNIOR SUBROUTINE PACHA.  
 IN DE SUBROUTINE WORDT  
 EEN SOFTWARE BREAKTEST  
 UITGEVOERD. DE BREAKVECTOR DIENT  
 VOORAF TE ZIJN GEFINEEERD.

INPUT : ASCII KARAKTER IN ACCU  
 Y REGISTER GAAT VERLOREN.  
 X REGISTER GAAT EVENTUEEL  
 VERLOREN.

0270: 9000 48 UART PHA  
 0280: 9001 20 39 UARTA JSR  
 0290: 9002 00 22 UARTA BCS  
 0300: 9003 2C 02 19 BIT BCS  
 0310: 9004 50 F6 8VC JSR  
 0320: 9005 20 20 8VC BCS  
 0330: 9006 00 18 8VC LDRAM \$10  
 0340: 9007 20 02 19 AND STATUS  
 0350: 9008 10 F4 BNE ENDRCH  
 0360: 9009 00 0C JSR BREAK  
 0370: 9010 00 08 BCS ENDA  
 0380: 9011 20 02 19 AND STATUS  
 0390: 9012 00 F4 BNE START  
 0400: 9013 68 PLA  
 0410: 9014 80 00 STA  
 0420: 9015 60 FF RTS  
 0430: 9016 00 9A LDXIM \$FF  
 0440: 9017 00 09 TXS  
 0450: 9018 00 09 LDYIM \$09  
 0460: 9019 41 90 LDARY TABEL  
 0470: 9020 20 00 JSR UART

\* SUBROUTINE OM EEN ASCII \*\*  
 \*\* KARAKTER VAN DE UART TE \*\*  
 \*\* HALLEN.

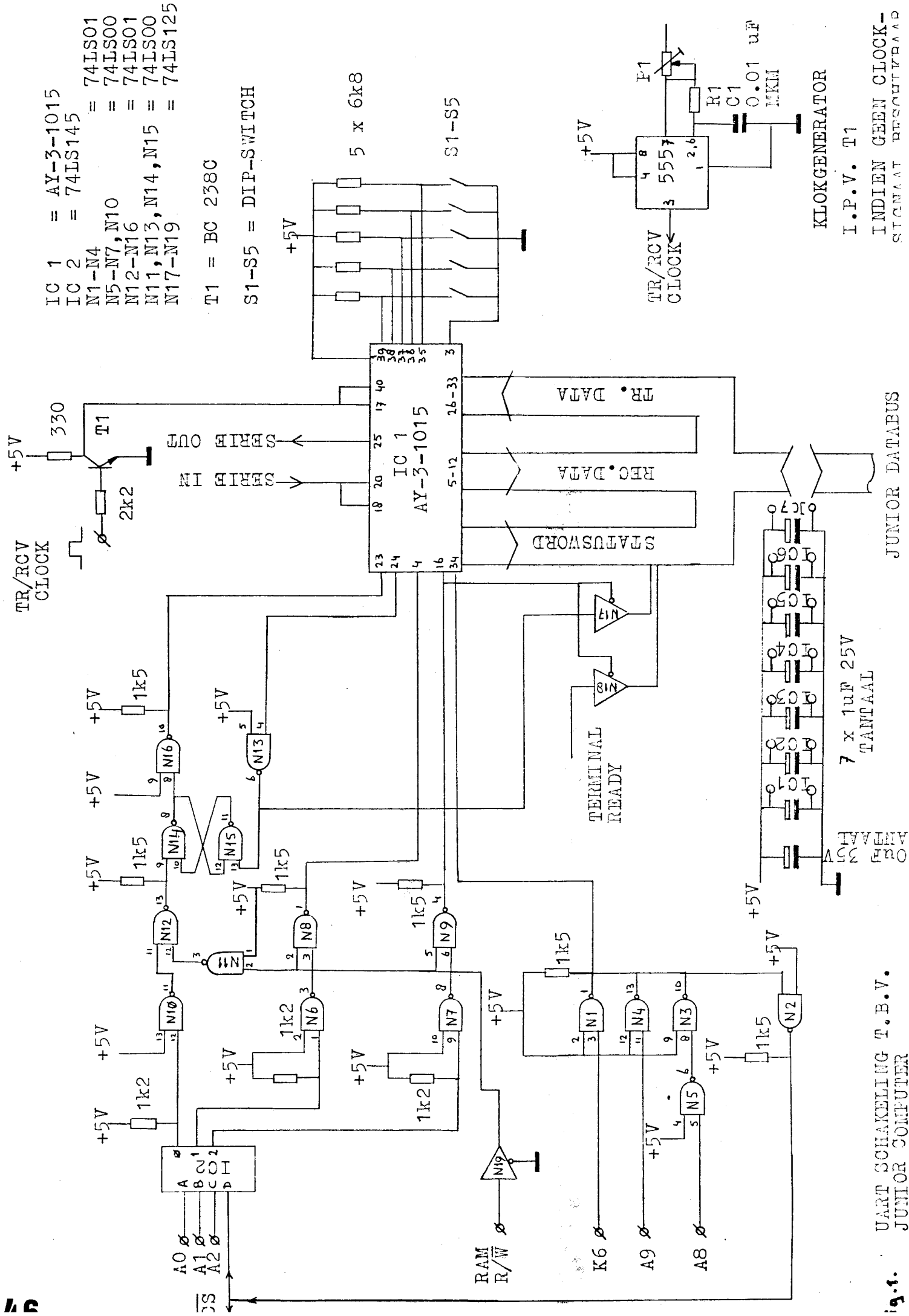
DEZE SUBROUTINE KAN WORDEN  
 GEBRUIKT IN PLAATS VAN DE  
 JUNIOR SUBROUTINE RECCHA.  
 RETURN MET ASCII KARAKTER  
 IN ACCU

STATUS DATA BESCHIKBAAR GERESET  
 DOOR STARTBIT ?  
 UARTRC NEE WACHT OP STARTBIT  
 STATUS JA WACHT TOT DATA  
 BESCHIKBAAR IS  
 UARC DATA NOG NIET BESCHIKBAAR  
 LDA RECELV BAAL KARAKTER UIT  
 DATA BUFFER  
 ANDIN \$7F STRIP TOT 7 BITS  
 RTS

ZET ACCU TIJDELIJK OP STACK  
 BREAK ?  
 STOP PRINTEN  
 ZENDBUFFER LEEG ?  
 BREAK ?

EINDE (VORIGE) KARAKTER ?  
 BREAK ?

HAAL ASCII VAN STACK  
 LAAD ZENDBUFFER  
 RESET STACKPOINTER  
 INDIEN NOODDZAKELIJK ANDERS  
 WEDLATEN  
 INDEX VOOR BREAKVELDING  
 PRINT ORK



- IC 1 = AY-3-1015
- IC 2 = 74LS145
- N1-N4 = 74LS01
- N5-N7, N10 = 74LS00
- N12-N16 = 74LS01
- N11, N13, N14, N15 = 74LS00
- N17-N19 = 74LS125

T1 = BC 238C

S1-S5 = DIP-SWITCH

KLOKGENERATOR

I.P.V. T1

INDIEN GEEN CLOCK-SIGNAAL BESCHIKBAAR

JUNIOR DATABASE

7 x 1uF 25V TANPAAL

UART SCHAKELING T.B.V. JUNIOR COMPUTER

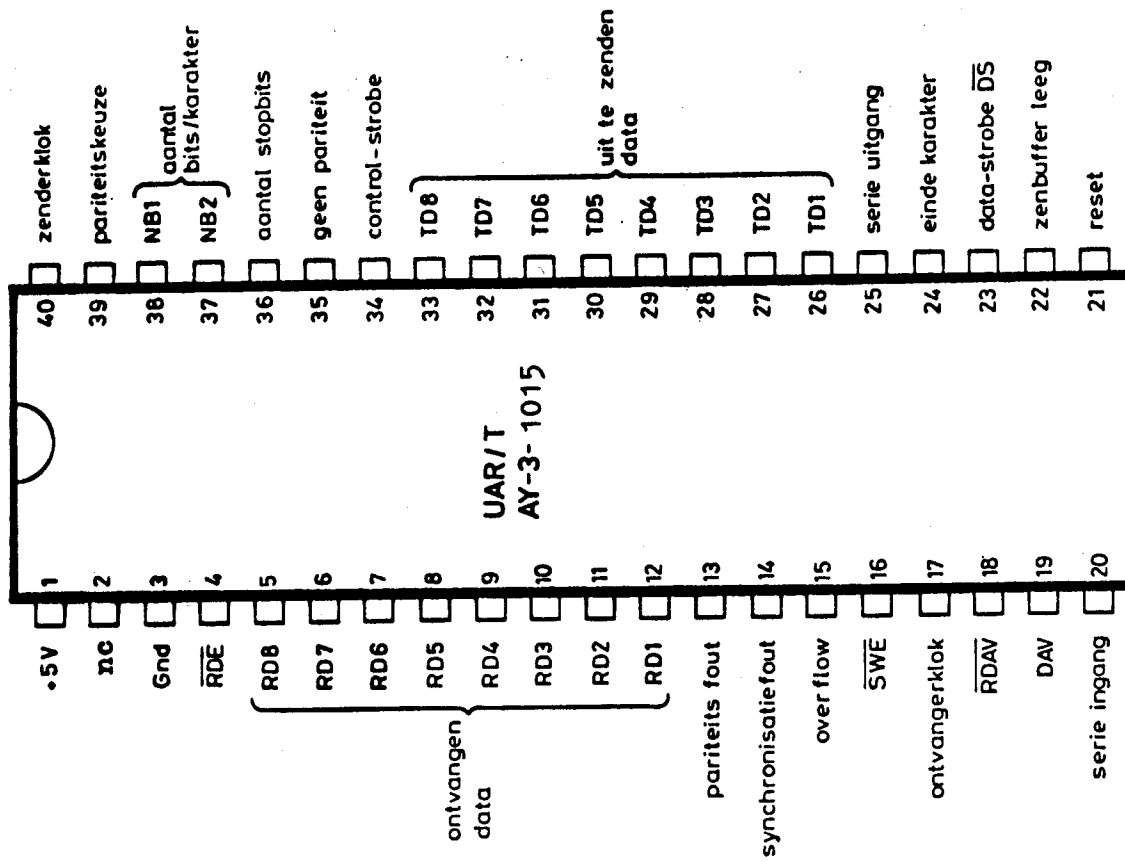


Fig. 3.

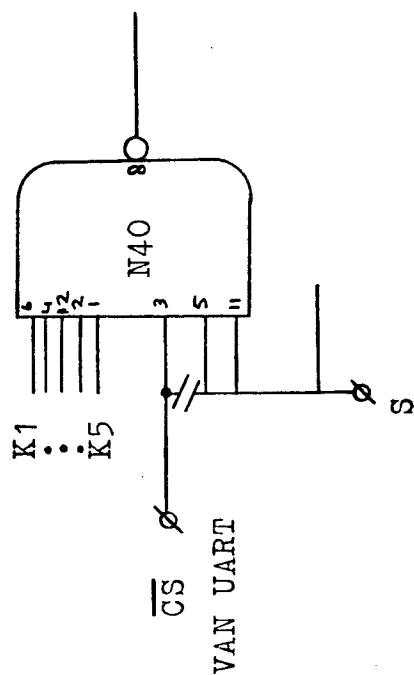
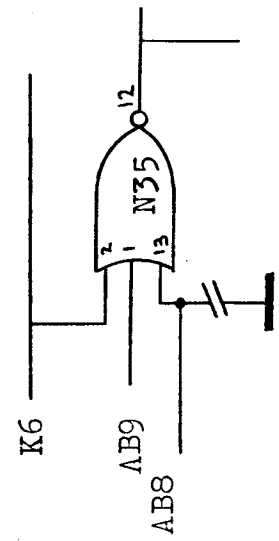


Fig. 2.



HARDWARE WIJZIGINGEN JUNIOR I/O PRINT  
 T.B.V. UART SCHAKELING.

AUTOMATISCHE REGISTER UITLEZING II

Paul van Niekerk  
Kievitslaan 8  
3233 BE OOSTVOORNE

Waarom nu al wèèr een register-uitleesprogramma?  
Was die uit 6502-Kenner nr. 16 soms niet goed?

Welnu, het programma uit 6502-Kenner nr. 16 werkt uitstekend, maar ... als je toevallig de cassette-interface van Elektuur bezit, dan kun je dat programma NIET op de band zetten. Dit komt, omdat de Elektuur-software gebruik maakt van de geheugenplaatsen 1A69 t/m 1A79. Hier mag dus geen programma staan! Het onderstaande programma voldoet ruim aan deze eis, en kan dus wel op de band gezet worden.

1A00	85	FB		STAZ	POINTH
1A02	86	FA		STXZ	POINTL
1A04	84	F9		STYZ	INH
1A06	A9	04		LDAIM\$04	
1A08	48			PHA	
1A09	28			PLP	
1A0A	20	8E	1D	JSR	SCANDS
1A0D	D0	FB		BNE	TERUG
1A0F	20	8E	1D	JSR	SCANDS
1A12	D0	F6		BNE	TERUG
1A14	A4	F9		LDYZ	INH
1A16	A6	FA		LDXZ	POINTL
1A18	A5	FB		LDAZ	POINTH
1A1A	4C	00	1C	JMP	SAVE

De registers A, X en Y moeten gedisplayed worden en dus wordt hun inhoud getransporteerd naar de displaybuffers. Vervolgens wordt het P-register goed gezet. Het origineel staat immers op de stack, evenals het terugkeeradres. Nu wordt er gedisplayed zolang er een toets is ingedrukt. Dit is natuurlijk de STEP-toets (ofwel de GO-toets), die men nog steeds vast heeft. Dan worden de registers weer geladen en springt het programma naar de SAVE-ingang van de monitor. In dit gedeelte van de monitor worden POINTL en POINTH van de stack gehaald. Nu staan weer alle monitorfuncties ter beschikking en kan men bijv. weer op GO drukken.

Misschien ten overvloede:

- Het NMI-adres moet gericht zijn op 1A00.
  - De STEP-schakelaar moet op 'on' staan.
  - Pagina 1A mag niet stap voor stap doorlopen kunnen worden.
- Er is nog ruimte genoeg over om bijv. ook de plaatsen PAD, PADD, PBD, PBDD en BYTES (00F6) tijdelijk op te bergen indien men deze plaatsen door het te testen programma laat veranderen.

Het bovenstaande werkt al geruime tijd uitstekend.