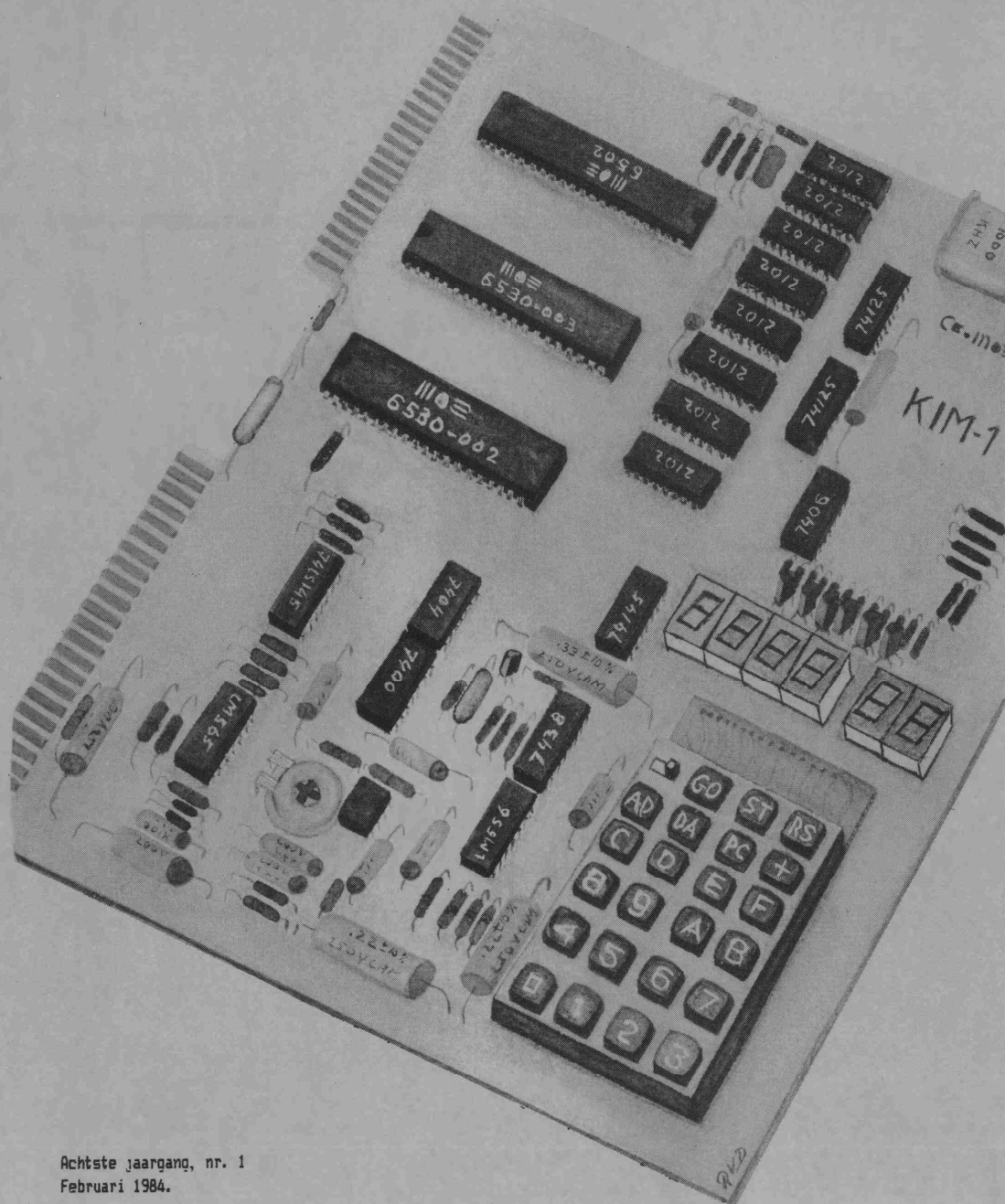




KIM GEBRUIKERS CLUB

NEDERLAND

een club van 6502 gebruikers

Het doel van de vereniging is: het bevorderen van de kennisuitwisseling tussen de gebruikers van 6502-computers, zoals KIM, SYM, JUNIOR, AIM-65, SYSTEM 65, ACORN, PET en CBM, APPLE, CHE-1, ATARI, de Proton-computers, ITT 2020, PEARCOM, PC-100, OSI, BBC, VIC-20 en COMMODORE 64, BASIS 108, PALLAS, SBC 65/68, FORMOSA, ORIC-1, ACE 1000, KEMPAC Systeem 4, NCS 6502, MINTA, enz. enz.

Deze kennisuitwisseling wordt o.a. gerealiseerd door zesmaal per jaar DE 6502 KENNER te publiceren, vijf maal per jaar clubbijeenkomsten te houden, een cassettebibliotheek in stand te houden en paperware-service te bieden. Daarnaast organiseren de leden zelf regionale bijeenkomsten.

Verschijningsdata DE 6502 KENNER =====

derde zaterdag
van de maanden:
februari
april, juni
augustus
oktober
december

Regio Den Haag/R'dam e.o. =====

Inl. Gerard van Roekel
v.d. Palmstraat 11-C
3135 LK Vlaardingen
Tel.: 010 - 351101

Bijeenkomsten van de club =====

derde zaterdag
van de maanden:
januari
maart
mei
september
november

Regio Arnhem e.o. =====

1e en 3e Ma-avond v.d. maand
in HET DORP
(zaal wordt aangegeven in de
hal van de personeelsflat)
Inl.: Henk Deutekom
Tel.: 08303 - 8987

De KIM GEBRUIKERS CLUB NEDERLAND is een volledig onafhankelijke vereniging met statuten en een bestuur. De club is ingeschreven bij de Kamer van Koophandel en Fabrieken voor Hollands Noorderkwartier te Alkmaar, onder nummer 634305.

Samenstelling van het bestuur is thans als volgt:

Voorzitter:

Dagelijks bestuur:
Rinus Vleesch-Dubois
Fl. Nightingalestraat 212
2037 NG Haarlem (023 - 330993)

Erevoorzitter: Sekretaris:

Siep de Vries
Ruud Uhoff
Voorburgpad 10
6843 EM Arnhem (085 - 816935)
(19.00 - 20.00)

Penningmeester:

Riet Uhoff - Burghoorn
voor adres: zie bij sekretaris

Ledenadministratie: Redaktie DE 6502 KENNER

Leden:
Ruud Uhoff (zie bij sekr.)
Willem L. van Pelt
Jacob Jordaensstraat 15
2923 CK Krimpen a.d. IJssel

Hard-/Software:

Adri Hankel (01807 - 19881)
John van Sprang (05490 - 51151)
Erwin Visschedijk (01807 - 20589)
Jan Tompot (05490 - 71416)
Jan Tompot (070 - 931417)

Promotie:

Ereleden:

Mw. H. de Vries - Van der Winden
Anton Mueller

Adres van de vereniging: Lidmaatschappen:

zie adres sekretaris
Fl.45. = per kalenderjaar, postrekening 3757649 t.n.v. KIM Gebruikers Club te Arnhem

Advertentietarieven:

Op aanvraag bij de redaktie

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 IJSSEL.
Tel.: 01807 - 19881

Redactie DE 6502 KENNER:
Vaste medewerkers:

Willem van Pelt
Gerard van Roekel
Frans Smeehuijzen
Jaap van Toledo

Freelance medewerkers:

Adri Hankel
Renee de Hoo
Fridus Jonkman
Herman Kuyvenhoven
Roger Langeveld
Anton Mueller
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 6 x per jaar en heeft een oplage van 500 exemplaren.

Copyright (C) 1984 KIM Gebruikers Club Nederland.

De voorpagina is een aquarel van een KIM, geschilderd door:
Rinus Vleesch Dubois

INHOUDSOPGAVE DE 6502 KENNER NR. 30 FEBRUARI 1984

1. Van het bestuur	2.
2. Agenda	3.
3. UITNODIGING Ledenvergadering/Clubbijeenkomst GELDROP	4.
4. Van de redactie	3.
5. COMAL-nieuws	2.
6. Commodore 64 Recorder/Telex aan Commodore 64 ... Fer Weber	5.
7. BASIC Bugs in Superstartrek ... Rob. Banen Wiskunde programma: determinanten berekenen, kwadratische vergelijkingen oplossen, N boven K uitrekenen, binominale kansen berekenen, cumulatieve binominale kansen berekenen. ... Jaron Borensztajn Uitbreidingen op SYM-1 Basic voor KIM-1 en JUNIOR: Automatische regelnummering. ... Frans Smeehuijzen	16. 40. 37.
8. Micro-ADE ... System description	22.
9. FORTH Using Forth with the 6502 ... Fridus Jonkman	48.
10. Usurpator II, Schaakprogramma voor JUNIOR ... Dirk Verzijden	8.
11. PRCHA, patch on PM-monitor ... Fernando Lopez, Portugal	9.
12. Programmeren van 2764 ... Roger Langeveld	10.
13. Bouw eens een MILJONOR: 2 miljoen bytes voor Uw JUNIOR ... Jan H. Vernimmen	11.
14. Program Cassette 1: Junior/apple formaat cassetteroutines. (deel 2 in volgende editie) ... Ad Brouwer	17.
15. Schuiven of Springen: vervolg van editie 29 ... Roland de Pretere, België	23.
16. Led's op JUNIOR ... Hans Buurman	34.
17. Printroutine voor STAR DP 510 ... Sebo T. Woldrinch	35.
18. Lichtkrant Beeldscherm (geprint met Tandy-plotter) ... Ivo van Rijssel	42.
19. TUTOR, morse-trainingsprogramma ... Theo G. Driever	45.
20. Load your JUNIOR Sym-Basic Tapes into KB-9: zie ook editie 29 Dubbeladressering VIA 6522 ... Hans Buurman	47. 47.
21. Missing Data in Junior-book 4	39.
22. Informatiepakket DOS VNK	41.
23. Teleac-cursussen	16.47.
24. News from 6502 front	16.
25. Vraag en aanbod	2.16.41.

VAN HET BESTUUR

Hobby en de economische recessie

"Wat krijgen we nu?", zult U zich afvragen, wanneer U bovenstaande aankondiging leest. Zijn de dag- en weekbladen nu al niet genoeg gevuld met doorgaans negatieve berichten? Moet mijn hobby er nu ook al aan geloven? Welnee! Natuurlijk, een hobby als micro's is niet bepaald goedkoop, al zijn de laatste jaren de prijzen van hard- en software fors gedaald. Een aspect van onze hobby is het feit, dat je de "horizon nooit ziet". In den beginne was er de single-board computer met hex-toetsenbordje en een zestal disolay's. Via uitbreidings- en geheugenkaarten, groot toetsenbord, VDU, enz., denk je: "Nu is het al een aardig systeem". Maar ja, een printer is wel erg handig, en floppy's zijn toch veel sneller dan je cassette recorder... Waar is het einde? Maar goed, er moet toch wat te wensen overblijven, nietwaar?

Waar slaat die titel dan wel op? Welnu, op de algemene ledenvergadering van 19 november 1983 werd het bestuur met een voor haar geheel nieuw probleem geconfronteerd. Een van de daar aanwezige leden stelde, dat de tot op heden gehanteerde toegangsprijs van Fl. 25,- voor veel leden een te hoge drempel is, een en ander in samenhang met de laatste jaren sterk gestegen vervoerskosten. Het bestuur zegde de leden toe zich over dit probleem te zullen buigen. Uitgangspunt is altijd geweest, dat bijeenkomsten financieel gezien uitte moesten draaien, en dus geen beslag mogen leggen op de overige middelen. Het bestuur is van oordeel dat dit zo moet blijven. De grootste onkostenfactor bij bijeenkomsten was de te betalen zaalhuur en lunch. Tot op heden worden een aantal, min of meer vaste lokaties gebruikt waarvoor een wisselende huurprijs geldt.

Om nu de entreeprijs te drukken, heeft het bestuur de volgende besluiten genomen:

- de bijeenkomsten worden zoveel mogelijk gehouden in accommodaties die gratis, dan wel voor een luttel bedrag te huur zijn (bijvoorbeeld: bedrijfskantines, aula's van scholen, etc.);
- de lunch dient voortaan door de bezoeker zelf verzorgd te worden;
- voor de jaarvergadering (de belangrijkste bijeenkomst) wordt geen entree gevraagd;
- voor alle overigen bijeenkomsten wordt de entreeprijs vastgesteld op Fl. 10,==.

Het bestuur hoopt met deze besluiten te bereiken dat nog meer leden dan voorheen in staat zijn de bijeenkomsten te bezoeken, en op die manier een hoger rendement uit hun lidmaatschap halen.

Deze nieuwe regeling treedt in werking op de bijeenkomst te Geldrop op 17 maart 1984.

Namens het bestuur,
Adri Hankel.

LAATSTE NIEUWS

Op de altijd weer bijzonder gezellige bijeenkomst te Krommenie heeft onze sekretaris Ruud Ushoff zijn uit een Basic interpreter geschreven COMAL*.KGN geïntroduceerd. COMAL is een afkorting die staat voor COMMON ALgorithmic Language. De aanduidingen erachter hebben te maken met een clubcode. KGN betekent Kim Gebruikersgroep Nederland, de gehele aanduiding komt vanwege onze cassettebibliotheek. De aanduiding als door ons gebruikt brengt tevens tot uitdrukking dat er onderscheid moet worden gemaakt met COMAL-80. Er is echter vrijwel geen verschil. Er is sprake van gestructureerd programmeren. Borg Christensen, leraar aan het Seminarium voor Leraren in Tindor (Denemarken) ontwierp COMAL op basis van de structuren van PASCAL en zette deze structuren om in de optekenstijl van BASIC. COMAL*.KGN kent een aantal ovaallende commando's en control-structuren, waaronder Delete, Rename, Label, Goto, Goto, Resume, If-Then-Else-Endif, While-Do-Endwhile, Repeat-Until, For-To-Step-Endfor, Case-When-Otherwise-Endcase, Proc-Endproc, Exec!.

Voor JUNIOR is COMAL*.KGN verkrijgbaar op band samen met een bekende handleiding voor Fl. 22,50 bij de redactie. Betaling op giro 841433 t.n.v. W.L. van Pelt te Krimpen a.d. IJssel. Eurocheque mag ook. Of storting op giro 3050. AMRO-bank te Krimpen a.d. IJssel ten gunste van rekening 44.11.06.471 op naam van W.L. van Pelt.

CURSUS Comal-80 voor beginners.

Voor beginners met enige kennis van Basic of andere computertaal wordt in de Scholengemeenschap "De Krimpenwaard" te Krimpen a.d. IJssel een cursus COMAL-80 gegeven op 15, 22, en 29 maart en op 5 april 1984 des avonds. De kosten bedragen Fl. 150,== per persoon. Inlichtingen: Hans Moot, o/a Stichting Pascal, Postbus 93525, 2509 AN Den Haag, Tel: 070 - 470261.

COMAL-80 DAG.

Op zaterdag 7 april 1984 organiseert de COMAL-80 Gebruikersgroep de eerste landelijke COMAL-80 dag, van 10.00 tot 14.00 uur in bovengenoemde Scholengemeenschap. De geestelijke vader van de programmeertaal COMAL-80 en andere voorafstaande COMAL-kenners zijn uitgenodigd. Inlichtingen over deze dag, waaraan geen kosten zijn verbonden, zijn te verkrijgen bij eerdergenoemde Hans Moot.

VRAAG en AANBOD.

=====

Te koop aangeboden:
Keyboard + Elektterminal video + RS232 in/out.
Met de nodige manuals, alles in een kast.
Prijs: Fl. 150,==.
Will Cuijpers, Telefoon: 01660 - 3566.

OPROEP.

Een programma ontwikkeld? Hardware ontwikkeld?
Stuur op naar de redactie, o/a Willem L. van Pelt
Jacob Jordaensstraat 15, Krimpen a.d. IJssel.
Tel.: 01807 - 19881.

A G E N D A

- ZA 17-03-84 CLUBBIJEENKOMST TE GELDROP
Gemeenschapshuis "De Zes Gehuchten"
Papenvoort 10, Geldrop (bij Eindhoven)
Toegangspreis verlaagd van Fl.25,- naar
Fl. 10,- !! Lunchpakket zelf medenemen.
- VR 04-04-84 CeBIT - Hannover Fair
DO 10-04-84 Info: Deutsche Messe und Ausstell. AG,
Messegelede, 3000 Hannover 82,
Germany
- ZA 07-04-84 REGIONALE 65XX BIJEENKOMST
CLUBHUIS DE BOSKANT, UILEBOMEN, DEN HAAG
Informatie: G. van Roekel, Tel. 010-351101
- DI 08-05-84 COMPEC EUROPE 84
WO 09-05-84 Brussel, België.
- DI 22-05-84 MICRO EXPO
VR 25-05-84 Paris, France
Info: Sybex, 4 Place Felix Eboue,
75583 Paris Cedex 12, France.
- ZA 09-06-84 REGIONALE 65XX BIJEENKOMST
CLUBHUIS DE BOSKANT, UILEBOMEN, DEN HAAG
Informatie: G. van Roekel, Tel. 010-351101
- Iedere 1e en 3e Ma-avond v.d. maand in HET DORP:
REGIONALE 65XX BIJEENKOMST
zaal wordt aangegeven in hal personeels-
flat
Informatie: Henk Deutekom, 08303-8987

's Zondags vanaf 19.10 uur via Hilversum 2
Hobbyscoop, vaak met computerprogramma's, die
m.b.v. een cassette recorder zijn op te nemen.
Door het gebruik van 'Basicode' is elk uitge-
zonden programma met Uw computer in te lezen.
De Basicode voor de OSI en voor de APPLE II
zijn gepubliceerd in de HCC-Nieuwsbrief nr.34
van nov/dec 1981.
De Basicode voor de JUNIOR werd gepubliceerd in de
Elektuur van oktober 1983 en heeft betrekking op
de Microsoft Basic KB-9.

O P R O E P

De redactie verzoekt eenieder
die op zijn systeem een Disk
Operating System heeft draai-
en daarvan de redactie te in-
former. Wilt U daarbij ook
mededelen welk systeem er is
toegespast, welke mogelijkhe-
den het biedt, of welke te-
kortkomingen er zijn?
Heeft U nog geen DOS, maar is
dat wel een wens, geef het op.

Een zesde editie is voor de redactie een welkome uitbrei-
ding van haar mogelijkheden om de doorstroming van aanoe-
boden copy te vergroten. Waar het gaat om de inzending
van artikelen door de leden, nu zelfs uit Portugal, kun-
nen we nu in enige mate tegemoet komen aan de wens snel-
ler kennis uit te wisselen. Die wens bestaat echter bij
de gratie van het enthousiasme waarmee men inzendt. Aan
het adres van de nieuwe leden onder ons moet dit wel een
teken zijn hoe belangrijk het is niet passief te consume-
ren wat er zoal in DE 6502 KENNER verschijnt. Wij zijn
immers een club waar leden met elkaar hun kennis uitwis-
selen, zoveel mogelijk via DE 6502 KENNER, en niet enkel
omwille van het feit dat dat documentatie oplevert, maar
ook omdat U er mensen mee bereikt die niet in Uw direk-
te omgeving wonen. De documentatieve waarde wordt nogal
eens onderschat. Wat denkt U van de mogelijkheden die de
redactie kan bieden aan mensen die pas een computer aan-
geschaf hebben, maar die over nog onvoldoende informa-
tie beschikken. Menig telefonisch verzoek kon worden in-
gewilligd, dank zij Uw bijdragen aan DE 6502 KENNER!

Enige tijd geleden heb ik, enigszins in mijn clubbeur aan-
getast, de redactie van Personal Computer Magazine ooge-
beld. Alvorens tot de eerste uitgave van dit blad over
te gaan, brachten zij een goed gedocumenteerde Koooids
op de markt, waarin zij melding maakten van het feit dat
zij aandacht zouden schenken aan de computerclubs in ons
land. Waar ik ook zocht, wij waren niet te vinden. Toen
ik de redactie in haar oor fluisterde dat je dan toch op
z'n minst aandacht moet schenken aan de oudste in Neder-
land bestaande computerclub, de KIM Gebruikers Club, heb-
ben zij onmiddellijk beloofd die fout te herstellen. Dus
in het januari-nummer van Personal Computer Magazine een
lang verhaal over onze club. We zijn er verguld mee.

Over publicaties, speciaal patches op bepaalde talen, zo-
als Basic, Forth, Focal, Micro-Ade, de assembler van C.
Moser (RAE), Pascal, de club-COMAL, etc. wil ik een oo-
merking maken. In meerdere gevallen leeft men in de ver-
onderstelling dat gepubliceerde patches per definitie de
originele versie wijzigen. Dat is niet ons beleid. De pu-
blikaties geven slechts aan op welke wijze er eventueel
veranderd kan worden. Indien voorgaande publicaties in-
vloed hebben op te publiceren patches, dan proberen we
daarvan in elk geval melding te maken. Het kan dus voor-
komen dat een patch op adressen staat welke al eerder ge-
bruikt zijn. Het hangt van Uzelf af of U de patch wilt
implementeren. Met Micro-Ade kunt U gemakkelijk wijzi-
gingen aanbrengen of de patch verplaatsen naar andere de-
len van het geheugen.

De samenstelling van het bestuur is gewijzigd. Adri Han-
kel, John van Sprang, Jan Tompot en Erwin Visschedijk ko-
men de geleerden versterken. Vanuit de redactie wensen
wij hen allen veel wijsheid en daadkracht toe. Moge hier-
mede de club een nog aantrekkelijker tehuis voor compu-
terende 65XX gebruikers worden. Wellicht kunnen we hen
in hun taak bijstaan door met z'n allen te trachten onze
club groter te maken.

De redactie mag dan in een feeststemming zijn vanwege de
in 1984 uit te brengen zes edities, het feit dat we het
dertigste nummer uitbrengen, omhullen we met het gegeven
dat haar enthousiasme zich ontploft in een nieuw een nummer
dat dank zij Uw medewerking geslaagd kan worden genoemd.

W.L. van Pelt.

UITNODIGING BIJeenkomst

Datum : zaterdag 17 maart 1984
Lokatie : Gemeenschapshuis "De Ies Gehuchten",
Papenvoort 10 te Geldrop (bij Eindhoven)

Reisroute :

- per auto - vanaf snelweg E3 Belgie - Venlo
Afslag Geldrop, onder E3 door bij Sheraton hotel.
Eerste stoplicht linksaf. Na 800 meter bij kruising linksaf. Die weg 1 Km volgen tot iets voorbij de kerk. Hiernaast is het.
- vanuit Eindhoven
Rondweg, afslag Geldrop bij DAF. Bij stoplicht voor spoorwegviadukt in Geldrop rechtsaf. De weg 1 Km volgen tot iets voorbij de kerk. Hiernaast is het.
- vanuit Nuenen - Helmond
Richting Eindhoven door Geldrop volgen. Bij eerste stoplicht na spoorwegviadukt linksaf. Verder als bij bovenstaande route.
- per trein - Stoptrein Eindhoven - Weert (ieder half en heel uur vanuit Eindhoven). In Geldrop uitstappen (!) Voetgangerstunnel naar station door. Deze weg (Toernooiveld) oversteken en rechtsaf slaan. Linksaf (Hertogenlaan). Rechtsaf (Gildestraat). Linksaf (Diepenvaart). Aan het einde bij de kruising is het (rechts).

GEWIJZIGDE

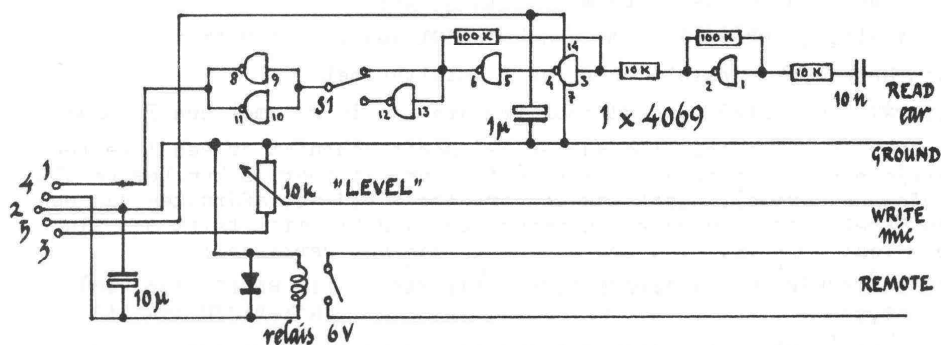
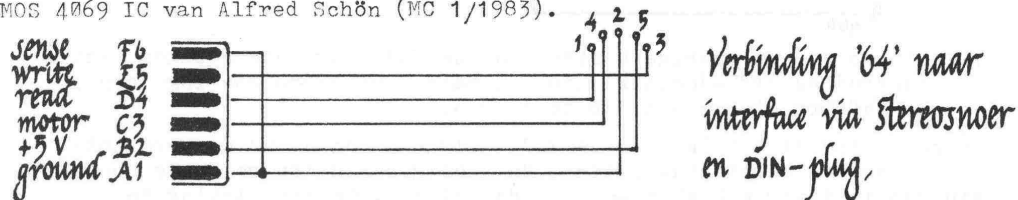
TOEGANGSPRIJS: Fl. 10,=. Lunchoakket zelf meenemen.
Consumpties tegen betaling.
Deze regeling is een experiment op verzoek van de ledenvergadering. (Toegangsprijs v/h Fl.25,=)

Programma:

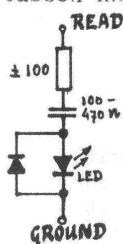
- 9.30 Zaal open
- 10.15 Opening
- 10.30 De P.A.L. (Programmable Array Logic), een lezing door Erwin Visschedijk uit Almelo (Wat de Eeprom is voor software, is de P.A.L. voor hardware)
- 11.45 Koffiepauze
- 12.00 Forum
- Aansluitend Lunchpauze.
- 13.30 **INFORMEEL GEDEELTE**
Het middagedeelte wordt geheel gevormd door het gebruikelijke informele gedeelte. Het bestuur wil bevorderen dat men elkaar zoveel mogelijk ontmoet, kennis kan nemen van elkaars systemen en kennis.
BRENG DAAROM OOK UW SYSTEEM MEE ALS DAT ENIGSZINS LUKT !!
MARKT. Op eigen tafel(s) te regelen.
- 17.00 Sluiting.

RECORDER/TELEX AAN COMMODORE 64

Ook ik kreeg bij de aanschaf van de Commodore 64 te maken met het feit dat ik bij mijn KIM-1 al twee cassetterecorders had staan. Uiteraard voelde ik er weinig voor nog eens een derde te moeten aanschaffen en dus heb ik eens in mijn papierwinkel rondgesnuffeld en vond een ruim aantal artikelen over een recorderinterface voor VIC en 64. Daar mijn rommelbak meestal vrij vol zit, had ik weinig moeite een apart kastje te bouwen waarin een interface en een eigen voedinkje onderdak vonden. Aan de buitenkant bevinden zich weliswaar nogal veel verbindingskabels (afhankelijk van type recorder), schakelaars, chassisdelen en LEDjes, maar het geheel werkt alweer een half jaar naar tevredenheid. Kern van het geheel is een schakelingetje rond een CMOS 4069 IC van Alfred Schön (MC 1/1983).



In de getekende stand van S1 werkt mijn recorder, maar andere recorders kunnen de geïnverteerde stand nodig hebben. Ook handig is het om de data binnen te zien komen, dus zet als volgt een LEDje tussen READ (EAR-uitgang van recorder) en GROUND.



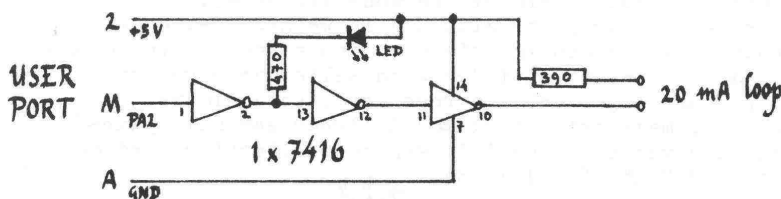
De gebruikte diodes zijn bv 1N4148.

Aangezien de 'sense'-leiding direct aan massa ligt, neemt de 64 altijd aan dat een toets op de recorder is ingedrukt. Zelf dus denken aan het indrukken van de juiste toetsen.

Voordeel van een eigen recorder is dat het niveau instelbaar is waardoor nog wel eens iets is in te lezen dat slecht op de band staat. Overigens leert de ervaring dat het lezen kritischer is dan met de Commodore recorder. Moeilijkheden zijn praktisch altijd terug te voeren op afwijkende kopstanden.

Wie verder wil experimenteren vindt in Microcomputing July 1983 nog iets aardigs evenals bv in oudere nummers van MICRO.

Na het aansluiten van mijn eigen cassetterecorder aan de Commodore 64 vond ik het ook gepast om mijn telex niet alleen door mijn KIM-1 te laten afbeulen. Zonder de mogelijkheid tot het afdrukken van allerlei "gekke" tekens is het weliswaar af en toe wat behelpen (tot ik ook dat met software heb opgelost), maar bij ontstentenis van een Commodore printer liggen er voldoende nuttige taken voor een telex. De hardware voor een 20mA loop bleek uiterst simpel te zijn:



Ikzelf heb aan de overige buffers van de 7416 nog andere zaken (zoals een eenvoudige RTTY-decoder) hangen, maar niet gebruikte ingangen van de buffers dienen aan GND te liggen.

Daarmee hing mijn telex aan de RS232 uitgang van de 64 en werd het tijd voor een eerste 64-listing. Na lezing van de betreffende hoofdstukken in diverse boeken was ook dat simpel. Om een listing te krijgen van een BASIC-programma tik ik nu in:

```
OPEN128,2,3,CHR$(131):CMD128,"naam":LIST gevolgd door 'return'
en het hele zaakje ratelt erop los. Afsluiten met
```

```
PRINT#128:CLOSE128 gevolgd door 'return' en ik kan met een listing
op papier aan de slag. File nummer 128 wordt gebruikt om een line feed
te krijgen na een carriage return (zie voor dat soort zaken boeken als
de Reference Guide). Bleef nog een probleem: bij het afdrukken van een
volle regel duurde de carriage return te lang bij mijn telex met als
gevolg fout printen. Niet erg elegant, maar het werkt wel:
```

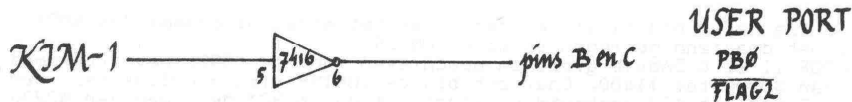
```
FOR I = 40960 TO 49151:POKE I,PEEK(I):NEXT (de BASIC naar RAM)
POKE 1,54 (RAM-BASIC draait)
```

Nu kan naar hartelust, edoch met overleg, geknutseld worden aan de BASIC. Ik geef het hier in hex:

```
AAD7 20 00 C8 EA EA (even naar een stukje vrije RAM)
C800 A9 0D (geef alsnog 'CR' uit)
C802 20 47 AB
C805 A9 00 (stuur nog een extra 'null')
C807 20 47 AB
C80A 60 (en weer terug naar BASIC)
```

Welnu, ik dus blij en telex blij. Maar ja, stilzitten is ook zo wat, dus bedacht ik dat een aantal van mijn KIM-BASICprogramma's best ook wel op de 64 tot hun recht zouden komen. Opnieuw intikken? Moet dat nou! Ik heb op de KIM-1 toch ook 20mA loops, dus laat die 64 maar netjes een ASCII-file binnenhalen. Tenslotte beschikt de 64 over een prachtige editor waarmee een BASIC-regel die op het scherm staat ook als BASIC-regel in het geheugen kan worden gezet... Cursor erop en op 'return' tikken. De BASICs zelf komen aardig overeen, alleen moet de

tekst aangepast worden aan het 40-karakter scherm van de 64. Goed, op mijn KIM-1 had ik al drie gelijktijdige outputs (terminal en telex printen tegelijkertijd) in de vorm van drie 20mA loops. Een daarvan naar de 64:



Nu moet de KIM via LIST telkens een aantal regels sturen (het scherm van de 64 moet niet overlopen). Voor het gemak laat ik de KIM met 110 baud zenden. Nu moet de 64 nog ontvangen:

```
60000 OPEN 1,2,3,CHR$(131)
60010 GET#1,A$:IF A$=""THEN 60010
60020 PRINT A$;
60030 GOTO 60010
```

Nu RUN 60000 en de 64 wacht op input van de KIM. Heeft die een aantal regels verzonden via LIST XXX-YYY dan kan de 64 worden gestopt met de STOPtoets.

Nu staan een aantal regels van het KIM-programma op het 64-scherm als tekst. Regels controleren op afwijkingen zoals te lange regels, CLEAR veranderen in CLR en tekst op 40-koloms formaat. Cursor op de regel zetten en op 'return' drukken. Op deze manier hebben we ook niets te maken met 'tokens'. Het geeft wat toetsarbeid maar ook hier geldt dat het werkt...

En toen gebeurde er iets waar ik niet op verdacht was omdat ik het nergens tegengekomen was. Bij het RUNnen van zo'n binnengehaald programma verscheen bij een bepaalde INPUT-regel telkens een foutmelding ondanks dat alles in orde leek te zijn.

WAARSCHUWING: het blijkt dat bij een INPUT statement de tekst tussen aanhalingstekens slechts maximaal 38 karakters mag bevatten! Ikzelf was ervan uitgegaan dat daar de normale grenzen voor golden zoals twee schermregels, maximale invoerbuffer en zo. Dat is dus niet zo.

```
INPUT "meer dan 38 karakters";A      levert op: redo from start
INPUT "meer dan 38 karakters";A$    in de A$ wordt eerst de uitgegeven
                                     tekst opgenomen en dan nog eens
                                     de ingevoerde karakters!
```

OK, tot nu is dit makkelijk te onderkennen als ik dit eenmaal weet, maar het kan in een programma nog in een andere gedaante voorkomen:

```
PRINT "meer dan 38 karakters";
:
:
INPUT A
of INPUT A$
```

Ook dit gaat in de fout en daar de twee statements ver uiteen kunnen liggen, kan het gepuzzel besparen als je dit aantekent in bv de Reference Guide.

USURPATOR II

Schaakprogramma voor JUNIOR

Dirk Verzijden
Van Almondeweg 26
3231 CS Brielle

Geachte Redactie,

Sinds enige weken ben ik in het bezit van het schaakprogramma "USURPATOR" en heb het draaiend gekregen op mijn JUNIOR.
USURPATOR II is 4.5Kbyte groot en geschreven voor de 6502. Het programma loopt van \$0200 tot \$1400. Daar dit bij de JUNIOR niet mogelijk is, heb ik de HiBytes met \$20 verhoogd en staat het nu in RAM-geheugen van \$2200 tot \$3400. Uiteraard moesten nog een aantal monitorroutines worden omgezet, evenals een 9-tal indirecte jumps.
Het programma kent diverse niveau's c.o. bedenktijden, mogelijkheden zowel voor spelen met wit als voor zwart, probleemstellingen kunnen worden ingevoerd en tijdens het "denken" van de computer kan status-informatie worden geprint.
Voor diegenen die niet opzien tegen enige uren type- en zoekwerk is dit een leuk en leerzaam programma. Een volgende keer kom ik terug met een patch die het mogelijk maakt het speelbord op de monitor te presenteren.

USURPATOR II is een uitgave van uitg. WOLFKAMP
I.S.B.N.: 90-70556-06-5. Prijs: ca. Fl. 19,50.

Patches USURPATOR II

Verhoog alle HiBytes met \$20, zoals ik boven reeds schreef. Wijzig geen HiBytes met de waarde \$00 en \$01, zoals bij de regels 0940, 0988, 0992, 1046, enz.

Vervolgens moeten nog enige monitor-routines, VIA-aanroep en Zero-Page-aanroepen gewijzigd worden:

0207	222A	AE 04 18	0361	232F	20 E8 11	0611	24CD	A2 27
0286	22A0	A2 27	0378	234D	4C 5F 10	2090	2EB2	A9 2E
0288	22A4	EA EA EA	0390	235F	20 AE 12	2218	2F7B	20 34 13
0289	22A7	EA EA	0397	236D	20 AE 12	2456	30F3	20 E8 11
0291	22AB	EA EA EA	0424	2399	20 34 13	2458	30F8	20 8F 12
0292	22AE	20 E8 11	0432	23A5	4C 34 13	2459	30FB	20 F3 11
0296	22B9	EA EA EA	0443	23B0	A2 2F	2462	3102	20 8F 12
0316	22E5	20 E8 11	0460	23D2	A2 27	2463	3105	20 F3 11
0319	22EB	20 8F 12	0490	23FA	A9 25	2466	310C	20 8F 12
0320	22EE	20 F3 11	0503	2416	A9 27	2467	310F	20 F3 11
0322	22F4	20 F3 11	0542	2458	A2 2F	2469	3114	20 8F 12
0330	22FA	20 AE 12	0554	2471	A2 27	2470	3117	20 F3 11
0345	230D	A9 2E	0604	24BC	A2 25	2472	311D	20 8F 12

Start het programma op \$2200 en geef na (R) de eerste zet, bijvoorbeeld E2E4. Hierna een return en de computer begint met het zoeken naar de tegenzet met de hoogste waardering. Tijdens dit zoeken komt op de monitor een hoeveelheid status-informatie, met aan het eind de tegenzet, bijvoorbeeld E7E5. Vanaf dat moment kan de volgende zet worden ingetoetst.

Het niveau kan worden gewijzigd door op adres \$2254 een andere waarde in te voeren.

De subroutine van \$30F3 t/m \$3120 geeft mogelijkheden tot een andere presentatie. Zo kan men ook het bord laten zien door - gedecodeerd - de zero-page van \$0000 t/m \$0040 te laten printen.

PRCHA

PAGE 01

```

0010: ***** PRCHA *****
0020:
0030: This is a patch on the PM-monitor routine which I
0040: needed.
0050:
0060: Author : Fernando Lopez
0070:         Colegio dos Orfaos do Porto
0080:         Largo Padre Baltasar Guedes
0090:         4300 Porto, Portugal.
0100:
0110: )Transmit an ASCII-character stored in accumulator to Elekterminal.
0120: )Adapted from the original and modified to 8 bits
0130: (also modified is the bit format of the UART).
0140: )A jump is made to the printer routine ($0C6E) if the printflag
0150: is null (00).
0160: )Save index X: index Y not used
0170:
0180: 9F 00 PRFLAG * $009F ***** AT RTS: *****
0190: 59 1A STPBIT * $1A59 =Y
0200: 60 1A TEMP * $1A60 =X
0210: 62 1A CHA * $1A62 Acc=FF
0220: 7C 1A BRKT * $1A7C N=x
0230: 80 1A PAD * $1A80 Z=x
0240: 82 1A PBD * $1A82 V=0
0250: C=0
0260: 6E 0C PRNTER * $0C6E *****
0270: 12 13 DELBIT * $1312
0280:
0290: 1334 PRCHA ORG $1334
0300:
0310: 1334 8D 62 1A STA CHA :save character
0320: 1337 A5 9F LDA PRFLAG :printer or video?
0330: 1339 D0 06 BNE VIDEO :
0340: 133B AD 62 1A LDA CHA :
0350: 133E 4C 6E 0C JMP PRNTER :printer!
0360: 1341 8E 60 1A VIDEO STX TEMP :save index X
0370: 1344 18 CLC :
0380: 1345 20 6E 13 JSR ONEBIT :transmit startbit (logical 0)
0390: 1348 A2 08 LDXIM $08 :set up for 8 databits
0400: 134A 4E 62 1A PRA LSR CHA :get next bit and send it
0410: 134D 20 6E 13 JSR ONEBIT :if C=0, bit is 0; if C=1, bit is 1
0420: 1350 CA PRB DEX :set up for next bit
0430: 1351 D0 F7 BNE PRA :all bits transmitted?
0440: 1353 AE 59 1A LDX STPBIT :amount of stopbits
0450: 1356 38 PRD SEC :
0460: 1357 20 6E 13 JSR ONEBIT :send stopbit (logical 1); Acc=FF
0470: 135A CA DEX :
0480: 135B D0 F9 PRD BNE PRD :all stopbits?
0490: 135D 2C 80 1A END BIT PAD :test for break (PAD=X000.0000)
0500: 1360 10 04 BPL BRKTST :N=0 if break key NV-- ---- Z
0510: 1362 AE 60 1A LDX TEMP :restore index X
0520: 1365 60 RTS :C and V flags are always cleared
0530: 1366 2C 80 1A BRKTST BIT PAD :key released?
0540: 1369 10 FB BPL BRKTST :
0550: 136B 6C 7C 1A JMI BRKT :jump to an user selectable vector
0560: 136E AD 82 1A ONEBIT LDA PBD :
0570: 1371 90 03 BCC $1376 :
0580: 1373 09 01 BIT-1 ORAIM $01 :send a logical 1
0590: 1375 2C 29 FE BIT-0 BIT $FE29 : (ANDIM $29) send a logical 0
0600: 1378 8D 82 1A STA PBD :
0610: 137B 4C 12 13 JMP DELBIT :delay 1 bit time (and RTS); Acc=FF
0620: :end of patch.

```

PROGRAMMEREN VAN 2764

Hoger Langeveld
Laan van Middenburg 26 a
2275 CC Voorburg

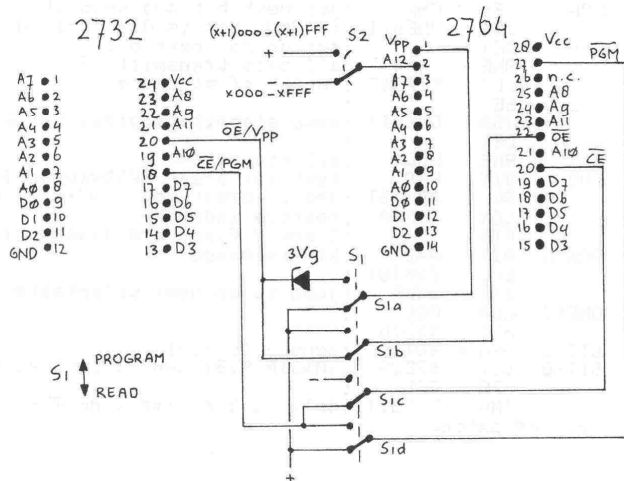
Aan de tekentafel:
Fridus Jonkman

Nu Elektuur met zijn 64K geheugenkaart is uitgekomen en velen al in het bezit zijn van de Eeprom-programmer van Elektuur voor het programmeren van 2716 en 2732, terwijl de prijs van de Eeproms 2764 ten opzichte van twee Eeproms 2732 gunstig ligt, ben ik gaan uitzoeken hoe de 2764 met bovengenoemde Eeprom-programmer geprogrammeerd kan worden.

In de bijlage zijn de aansluitingen van de 2764 te vinden. De 2732 geeft aan hoe de programmer moet worden geschakeld. Op stand 2732 dus.

Werkwijze:

- Stel C000 - DFFF moet in Eeprom komen (totaal 8K). We gaan ervan uit dat de Eeprom leeg is. We roepen de programmeerfunctie (P) aan voor het eerste deel: C000 - CFFF.
- Nu S2 op X000 - XFFF (X is de positie waar de Eeprommer geselecteerd staat).
S1 op PROGRAM.
- Op de Eeprommer S2 op off (programmeerspanning aanwezig).
Druk op toetsenbord de (M) in om de data te verhuizen.
Na verloop van tijd verschijnt de bekende melding DATA MOVED.
- Op de Eeprommer S2 dicht en S1 op READ.
Controleer de inhoud door (V) in te toetsen.
Als de melding DATA COMPARED verschijnt, is alles goed gedaan en is de eerste helft van de 2764 Eeprom gevuld met de door de (P)-functie opgegeven data.
- Voor de tweede helft van de 2764 veranderen we (P) en S2 - de schakelaar van A12 - omzetten. We herhalen dan de procedure als bovengenoemd.



BOUW EENS EEN MILJONOR !

tot 2 miljoen bytes voor Uw JUNIOR.

Door: J.H. Vernimmen
Van IJsendijkstraat 128
1442 CS Purmerend

De gedachte meer dan 64 K bytes te adresseren is al oud. De dwang het probleem op te lossen kwam zo'n half jaar geleden, toen ik 64 K dynamische RAM in m'n hand had en hiervan minimaal 8 K - en na aanschaf van EPROM's, 16 K of zelfs 24 K - ongebruikt zou moeten laten. Daar koop je ze niet voor!

Grotere systemen lossen dit probleem op d.m.v. Virtual Memory: een geheugenruimte die je ziet als behorend tot jouw geheugen, maar er niet direkt is. Anders gezegd: je werkt in een relatief kleine ruimte; zodra deze vol is wordt de inhoud gedumpt in een groot achtergrondgeheugen, jouw virtuele deel, (bijvoorbeeld op disk) en heb je weer een lege ruimte. Uiteraard wordt dit alles verzorgd door de systeemsoftware die tevens de administratie bijhoudt waar alles blijft.

Bestudering van deze systemen brachten mij op het idee niet alleen de inhoud weg te zetten, maar het gehele geheugen zelf, compleet met inhoud; m.a.w.: verwissel een vol blok met leed; verwissel een gebruikte blok (bijvoorbeeld textgenerator invoer programmatext) tegen een nog te gebruiken blok (Pascal-compiler of zoiets); enz. Dit alles natuurlijk softwarematig en niet door met je vingers IC's te verwisselen, of schakelaartjes om te moeten zetten.

Verwisselen van geheugenblokken betekent hardwarematig: het veranderen (hercoderen) van de adreslijnen die verantwoordelijk zijn voor het enable/disable signaal van de betreffende geheugen IC's.

Verdelen wij nu de beschikbare 64 K geheugenruimte in blokken van 8 K bytes, dan zijn steeds de adreslijnen A13, A14, A15 verantwoordelijk voor het geselecteerde blok. Een verdeling in blokken van 8 K leek mij het meest logische, daar dit hardwarematig nogal in de bouw van de JUNIOR verscholen ligt, terwijl veel verkrijgbare software (compilers, interpreters) 8 K omvatten of meervouden daarvan zijn en omdat ik dacht eenvoudige IC's te kunnen kopen om dit alles te realiseren. (Dit laatste heb ik geweten!).

De eerste eis was (en is): het hercoderen van de adreslijnen mag geen extra tijd (klokpulsen) kosten. Deze eis is dwingend en compromisloos.

Tweede vereiste: het moet betaalbaar blijven (rekbaar...).

Derde eis: de IC's moeten verkrijgbaar zijn.

Het basisidee ging uit van een IC type 82S12 of 82S112. Dit idee bleek niet aan m'n derde eis te voldoen ... ! Wie nog ergens zo'n IC heeft liggen weet nu waar hij hem voor gebruiken kan. In mijn optimistische dacht ik dit IC te kunnen nabootsen met een 7489 en wat tristate buffertjes. MIS! De 7489 heeft open-collector-outputs. Dit ontdekte ik na weken vergeefs experimenteren (in mijn Philips Data Handboek 1976 staat er echt niets over in). Bovendien kan dit IC 375 mW verwerken, en echt, hij vreet ze, alle 375! Je verbrandde je vingers en het soldeer werd er haast zacht van. Maar, doorwerken deed hij en betrouwbaar en snel als een oude stoomtram. Mijn RAM-metjes en EPROMs wilden nog wel meewerken, maar m'n moderne dynamische RAM's met toegangstijd van 150 nSec lieten het afweten, naar achteraf bleek, vanwege de adreslijnmultiplexer die een adres multiplexte dat nog niet klaar stond.

Het IC dat tenslotte uitkomst bracht, was een 74S189, volledig pin-compatible met de 7489, echter zonder open-collector-uitgangen en met een propagation delay (= signaalvertraging), die in de praktijk voldeed (met dank aan de heer Wim van Elburg, die half Nederland heeft afgebeld om er een te pakken te krijgen. Dit lukte voor ca. Fl. 35,- bij groothandel Malgus te Vlaardingen).

De 74S189 heeft ingaand 4 adreslijnen en 4 datalijnen, een CE-lijn (chip enable indien laag) en een WE-lijn (write indien laag). Uitgaand zijn er 4 datalijnen welke de geïnverteerde data bevatten. De algehele oozet is nu als volgt:
Direkt adresseerbaar zijn en blijven 64 K bytes. Door de adreslijnen A13, A14, A15 aan te sluiten aan de adresingangen van de 74S189 kunnen we 8 woorden adresseren van 4 bits breed (voor het IC de adressen 0 t/m 7). De zo uit te lezen data vormen de 4 nieuwe adreslijnen A13, A14, A15, A16. Hiermee kunnen we nu ineens 16 blokken van 8 K bestrijken. Let wel: we hebben er 16, maar we kunnen er maar 8 gelijktijdig van gebruiken. Welke 8 dit zijn, kunnen we nu zelf kiezen door de inhoud van de 74S189 te programmeren. Bovendien kunnen we, midden in ons programma, als we dat zouden willen, de inhoud van de 74S189 wijzigen d.m.v. slechts 2 instructies:
LDA adres/bloknummer
STAABS Register
We kunnen nu dus 1 bank van 128 K bestrijken met als het ware 64 direct toegankelijk, met nog eens 64 K virtueel geheugen achter de hand. De adreslijnen A13 ... A16 poorten we uit met een 74154 (4 line naar 16 line decoder) voor de chip select lijnen. Dit zal verder wel duidelijk zijn; ieder van de 16 lijnen zal 1 blok van 8 K selecteren.

De 74S189 bevat echter nog een ongebruikte adres-ingang, en dus nog 8 ongebruikte 4 bit-woordjes. Door nu een schakelaartje te creëren dat de bovenste adreslijn 1 maakt (en houdt) kunnen we de blokregistertjes 8 t/m F adresseren. Als schakelaartje nemen we een software bedienbare latch: de 7475, een Quad bistable latch.
Wanneer we de IQ uitgang van de latch hoog maken (dus 1) zijn alleen de blokregisters 8 - F toegankelijk. Bevorderen we nu dezelfde uitgang (of beter IQ en dan via een inverterend buffertje) tot adreslijn A17, dan hebben we ineens weer 128 K erbij (aan adressen; het geheugen zelf zal nog wel eerst aangeschaft moeten worden). Omdat in het eerstgenoemde geval de IQ uitgang 0 moet zijn en in het tweede geval 1 zullen we verderop dit resp. Bank 0 en Bank 1 noemen. We moeten ons twee dingen heel goed realiseren:
- Geheugen uitgepoord, gebruikmakend van adreslijn A17, kan maar in een bank voorkomen. Het is niet te selecteren in de andere bank.
- Geheugen uitgepoord zonder A17 (de systeemsoftware in blok 0 bijv.) komt zonder pardon in beide banks voor (dan wel: is in beide banks te selecteren. Aardig voor uitwisselen van data).
Resumerend: Op dit moment hebben we 2 banks van elk 128 K: elk opgebouwd uit 16 blokken van 8 K. Komt blok 0 uit bank 0 fysiek ook in bank 1 voor, dan hebben we nog steeds 2 banks van elk 16 blokken, maar de som van deze twee geeft ineens 31 blokken van 8 K = totaal 248 K, wat aardig is om mee te beginnen.

En toen had de latch nog poortjes (en pootjes) over ... Geven we elk van de 4 uitgangen z'n eigen bank, dan hebben we dus 5 banks van elk 128 K = 640 K intern geheugen (o ja, geen een poot = bank 0). Duwen we deze 4 uitgangen echter weer door een 74154, dan hebben we 16 banks = $16 \times 128 = 2$ KK bytes (de uitdrukking 2 M bytes is niet genormaliseerd).

Resteren de volgende problemen:

- 1. Waar brengen we de schakeling aan; d.w.z. op welke plaats moeten de adreslijnen A13, A14, A15 overgaan in de nieuwe A13, ... A16?
- 2. Welke adressen kiezen we voor het blokregister en het bankregister?
- 3. We kunnen het blok met de IRQ, RESET en NMI- vectoren verwisselen. Hoe vinden we deze vectoren terug?
- 4. Hoe starten we het systeem op? (Bedenk dat het register een Random (willekeurige) inhoud heeft en dus bij oostarten blok 0 niet meer blok 0 zal zijn).

Het eerste probleem is vrij simpel: zo dicht mogelijk bij de processor. Een heel goede plaats is de busverbinding tussen de basiskaart (print 1) en de interfacekaart (verderop print 2 genoemd). Dit vooral omdat de selectie van de RAM, PIA en EPROM in blok 0 op print 2 plaatsvindt: het signaal EX of BK0. Bovendien moet nu A16 in het EX-signaal verwerkt worden.

Probleem 2 en 3 bleken in combinatie zeer goed oplosbaar met een schakeling, ongeveer zoals in de Halfgeleidergids van 1982 werd gepubliceerd, poorten we de adressen FFF8 - FFFF uit (een 13-inout NAND-gate verzorgt dit). FFF8 en FFF9 hebben we over. (in de systeemsoftware 1C00 - 1FFF worden 1FF8 en 1FF9 ook niet gebruikt) en de lokaties FFFA - FFFF laten we keihard naar 1FFA - 1FFF wijzen door het signaal EX laag te maken voor beide printen.

Voor de lokaties FFF8/9 foppen we de JUNIOR. Door de EX voor print 2 laag te maken, verwijst deze naar print 1. Door het signaal EX voor print 1 hoog te maken, verwijst deze naar print 2. En de waarheid zit dus verbonden aan de verbindingbus tussen deze twee. Het is duidelijk dat de verbinding tussen EX op print 1 en EX op print 2 moet worden onderbroken. Verder moet pin 8 van IC 13 op print 2, welke tot nu toe het EX-signaal verzorgde, van de print worden losgekoppeld. Het is aan te bevelen hier een draadje+busje aan te solderen, zodanig dat de oude situatie kan worden hersteld. Hetzelfde geldt voor de onderbroken adreslijnen: soldeer aan de ene kant draad-enden met bussen welke tevens aan het blok-switching systeem kunnen worden aangesloten, en aan de andere kant pinnen, zodat tijdens elke fase van ombouw en testen de oorspronkelijke situatie kan worden hersteld. Het even kijken of de oorspronkelijke versie nog in takt is, kan heel rustgevend zijn voor de gemoedsrust van de geachte ombouwer. Wanneer dit niet meer zo is: controleer dan eerst of met al dat gedoe de oorspronkelijke verbindingen tussen print 1 en print 2 nog allemaal in orde zijn (ik spreek uit ervaring!).

Probleem 4 is niet te onderschatten. De onder beschreven oplossing voldoet, maar naar mijn gevoel zijn er betere denkbaar; ik heb me het meest op het eigenlijke switchen geconcentreerd. Om blok 0 te kunnen adresseren, heb ik een schakelaar aangebracht waarmee met of zonder extern geheugen kan worden geadresseerd: zie JUNIOR computer boek 3 blz. 65. Helemaal feilloos zal dit niet werken: EX op print 1 moet immers hoog worden om de twee registers te kunnen adresseren. Daartoe wordt het punt D van print 1 niet naar aarde gelegd, maar verbonden met stuursignaal K9 dat wordt gemaakt in de nieuwe schakeling: zie figuur selectie-signalen. Dit signaal wordt alleen hoog indien FFF8 of FFF9 wordt geadresseerd. Deze schakelaar heeft van mij de naam "BOOT-Switch" meegekregen: alleen met deze schakelaar in de stand "BOOT" (= geen extern geheugen) kan het systeem worden gestart. Pas wanneer BRO (blokregister 0) naar 0 wijst, kan de switch worden omgezet. Denk nog wel aan de resetvectoren op \$1A7A t/m \$1A7F.

Het schema selectiesignalen:

Via IC1 poorten we de adressen FFF8 t/m FFFF uit. Als A1 en A2 laag zijn, hebben we FFF8 en FFF9. A0 = laag geeft FFF8 en maakt signaal BCE. Dit signaal blokkeert de 74S189 (blokswitchregister) en geeft de 74LS75 bankswitchregister vrij voor data, mits RW = 0 (= write data). Signaal B, dat ontstaat wanneer een der registers is geselecteerd en er een write-operatie plaatsvindt, geeft de datalijnen D4 ... D7 vrij en blokkeert de adreslijnen A13, A14, A15 vanaf print 1. Is A0 hoog, dan vormen adreslijnen D4, D5, D6 het adres van IC74S189 waarin dan D0 t/m D4 als data (= hardware bloknummer) worden weggeschreven. Pogingen om reeds geschreven data terug te lezen heb ik (voorlopig) opgegeven, zij het na weken uitvoerig experimenteren (oortredende bijverschijnselen: o.a. "rondzingende" data, vertraging van het beschikbaar komen van de adressen door extra IC's in de adreslij-

nen, etc.).
 Houdt nog rekening met het volgende:
 Bij disable van, of een write-actie naar, IC 74S189 worden de uitgangen TRI-STATE. Het beste is weerstandjes in de uitgaande lijnen op te nemen die deze naar 0 trekken en niet gaan zweven waardoor dubbele adressering kan ontstaan.
 De adreslijnen A13, A14, A15 gaan direkt en alleen naar buffers (IC9 print 2). Om A16 te bufferen is van een nog beschikbaar poortje van IC 4 (fig. 1) gebruik gemaakt.
 IC 74S189 invertteert de data. Om de zaak software-matig eenvoudig te houden, wordt de data daarom geïnverteerd, voordat deze wordt weggeschreven (D0 t/m D3).

Software-matig moeten we als volgt te werk gaan:

```

A9 00      LDA #$00      nul in de ACC
8D F8 FF   STA $FFF8     in bankregister
A9 00      LDA #BR0.0    *) blokregister 0, inhoud 0
8D F9 FF   STA REGISTER  schrijf in $FFF9
A9 11      LDA #BR1.1    blokregister 1, blok 1
8D F9 FF   STA REGISTER
A9 7F      LDA #BR7.F    blokregister 7, blok F
8D F9 FF   STA REGISTER
  
```

*) NB: De geheugens FFF8/9 zijn WRITE ONLY. Daarom is na een STA FFF8/9 de DATA in de accu verminkt, dus opnieuw LDAIM \$00 ! Dit moet software-matig. Even laden via de SKIP-toets lukt niet. Het heeft nut om tenminste het laden van het bankregister met 00 en BR0 met 0 in de reset-routine op te nemen, plus een EPROM met reset-routines.

Een leuke mogelijkheid dient nog te worden vermeld. Elk (RAM-)blok is als blok 0 te adresseren. Dit geeft idem zoveel stack's. Bedenk dan wel dat de IRQ en NMI mogelijk bruikbaar moeten zijn, en dat onze RS 232, VIA en PIA heel ergens anders zitten.

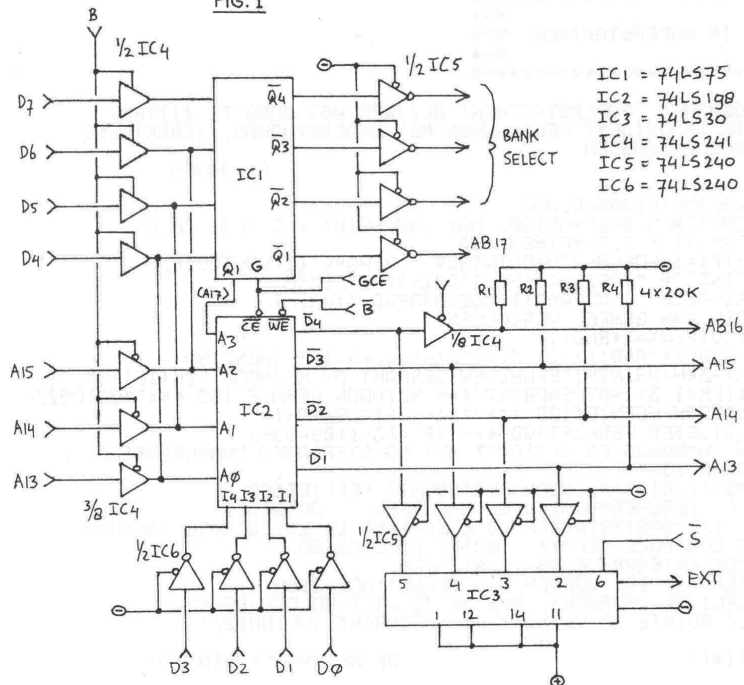
Met dynamische RAM's dient men voorzichtig te zijn. Het kritieke moment is het tijdstip waarop het multiplexen plaatsvindt. Eventueel kunnen de vrije poorten N43 en/of N44 tussen de punten A en B of C en D worden geschakeld. Bedenk dat dan de pull-up weerstanden R7 en/of R8 moeten worden aangebracht. Door de propagationdelay van deze poorten wordt het moment van multiplexen vertraagd. Trage dynamische RAM's kunnen nu problemen geven! Bedenk dat hoe groter de pull-up weerstand, hoe groter de propagation-delay. Die van het block-switching systeem ligt in orde grootte van 20 nSec. (genoemde oplossing m.b.v. N43 en N44 is niet door mij getest; was bij mij niet nodig).

Resumerend kunnen wij beschikken over maximaal

- 16 banks
- 16 blokken per bank
- 8 K bytes per blok (dus 128 K per bank)
- werkgebied 1 bank waarvan 8 blokken (64 K) als
 direkt toegankelijk werkgebied en
 8 blokken virtueel geheugen.

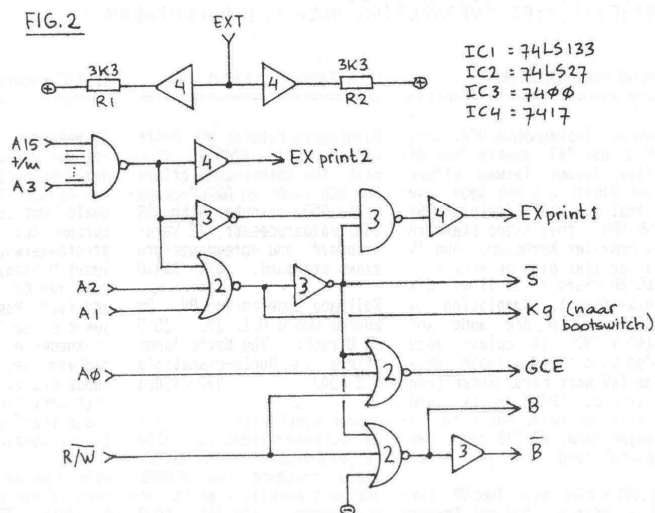
Schakeling gebouwd en getest in combinatie met een JUNIOR computer;
 systeemfrequentie: 1 MHz.

FIG. 1



Aan de tekentafel:
Fridus Jonkman

FIG. 2



DE 6502 KENNER

```

1 REM *****
2 REM ***
3 REM *** BUGS IN SUPERSTARTREK ***
4 REM ***
5 REM *****
6 REM
7 REM IN HET PROGRAMMA "SUPERSTARTREK" BLIJKEN WAT BUGS TE ZITTEN.
8 REM DE BUGS DIE IK ONTDEKT HEB KUNNEN MET ONDERSTAANDE, VERBETERDE
9 REM REGELS OPGELOST WORDEN.
10 REM
11 REM R. BANEN
1000 GOSUB4820:GOSUB5630:GOSUB3250
1350 PRINT"SPRONGMOTOREN BESCHADIGD. MAX.SNELHEID: 0.2":GOTO1010
1430 UF=0:FORI=1TOK3:IFK(I,3)=0THEN1460
1510 PRINTTAB(8):R1=I:GOSUB4370:PRINTG2$:" REPARATIE VOLTOOID."
1550 GOSUB4370:PRINTG2$:" BESCHADIGD.":PRINT:GOTO1590
1570 GOSUB4370:PRINTG2$:" REPARATIE GEVORDERD.":PRINT
1850 REM DEZE REGEL KAN GEHEEL VERVALLEN
2140 E=E-X:IFD(8) (OTHENX=X*RND(1)
2160 H=INT((H1/FND(0))*(RND(1)+2)/3):IF3*H).15*K(I,3)THEN2190
2190 K(I,3)=K(I,3)-3*H:PRINTH:"EENHEDEN GERAAKT OP KLINGON ":K(I,1):" ";
2200 PRINTK(I,2):IFK(I,3) (OTHENPRINT"*** KLINGON Vernietigd ***":GOTO2230
2410 PRINT"*** KLINGON Vernietigd ***":K3=K3-1:K9=K9-1
2500 PRINT"*** BASISSTER Vernietigd ***":B3=B3-1:B9=B9-1
2530 PRINT"VAN UW COMMANDO EN GESTRAFT MET 99 STERDAGEN DWANGARBEID"
2540 PRINT"OP CYGNUS 13."
2590 M3=M3-1:M9=M9-1:PRINT"KLINGON BASISCHIP Vernietigd."
2780 PRINT"KAMER." 'BESCHERMESCHILD ONVERANDERD.'":GOTO1010
2990 IFD(0)OTHENPRINT"BASISSTERSCHILD BESCHERM DE ENTERPRISE":RETURN
3070 PRINT"SCHADE CONTROLE MELDT:"G2$:" BESCHADIGD."
3110 PRINT:PRINT"DE ENTERPRISE IS Vernietigd."
3120 PRINT"DE FEDERATIE ZAL WORDEN VEROVERD.":GOTO3100
3170 PRINT"SDORTGELIJKE OPDRACHT. ALS JE JE WILT MELDEN DOE"
3220 PRINT"IN DEZE RUIMTE IS Vernietigd. OPDRACHT VOLTOOID."
3535 GOTO3600
3690 H8=0:G5=1:PRINT" DE MELKWEG":GOTO3730
3940 GOTO3960
3990 X$="":IFK3)1THENX$="S:"
4000 PRINT"VAN ENTERPRISE NAAR KLINGON":X$
4010 H8=0:FORI=1TOK3:IFK(I,3) (OTHEN4230
4320 IFLEN(A$) (3)THENGOSUB4770:A$=A$+" "
4770 IFUF=1THEN4720
4775 PRINT"BELANGRIJKE BOODSCHAP:"
4790 "KORTE AFSTAND SCANNER SIGNALEERT UFO'S IN DIT KWADRANT."
4815 UF=1
5090 PRINT"SCHILD VAN ENTERPRISE Vernietigd."
5430 K3=K3+1
5560 S3=INT(B(F1,F2)*1)-1000*M3-100*K3-10*B3
5590 PRINT"BASISSTER":F1:"";F2:"Vernietigd DOOR KLINGONENZWERM."
OK

```

TE KOOP AANGEBODEN:	NEWS FROM 6502 FRONT	NEWS FROM 6502 FRONT	TELEAC programma najaar 1984
5" Floppy-drives. Informatie te verkrijgen bij W. van Elburg. Telf.: 02984 - 1625 (na 21.00 uur).	Winmed Instruments Mfg. Corp. P.O. Box 217, Nankin Post Office, Taipei, Taiwan offers the MINTA, a 2 MHz 6502 personal computer featuring 64 KB RAM, full sized standard typewriter keyboard, and TV or monitor display unit with 40/80 chars x 24 lines display format. Resolution is 280 x 192 in B/W mode and 140 x 192 in colour mode. Add-in cards available include Z80 soft card, Eoson printer card, EPROM write card, 80-column card, PAL card, integer card, RS-232 card, language card and joy stick.	Robotronix/Fabulex BV, Delftweg 69 - 691, 2289 BA Rijswijk, The Netherlands offers the ACE 1000, an APPLE-compatible 6502 computer with 64K RAM, wordprocessor, 72 keys-keyboard and spreadsheetprograms standard. 015-134940	De bekende cursus Programmeren met Pascal, eerder uitgezonden najaar 1983, zal in het najaar 1984 worden herhaald. Het is een inleidende cursus tot gestructureerd programmeren. In de cursus leert U programmeren met behulp van de hogere programmeertaal Pascal. Bovendien leert U de toepassingsmogelijkheden en -onmogelijkheden van Pascal op verschillende computers. De serie telt achttien units, verdeeld over radio, televisie en cursusboek.
Sperry Univac Uniscope 100 Terminal, 12 x 80 char's. Werkt niet helemaal correct. (Display geheugen heeft kurren.) Incl. werkend toetsenbord met HAL - generatoren. Voor de knutselaar. Tegen Elk aanneemelijk bod. Th. den Exter, Enschede, 053-893923.	KIM-1 met 5K (1+4) geheugen, voeding en boeken. Het geheugen is uitbreidbaar tot 17K (1+16). Extra 2114L's. Voor Fl.160, Th. den Exter Enschede, 053 - 893923.	Neitraco Engineering BV, Beaurde Weerd O.Z. 39, 3513 AR Utrecht, The Netherlands offers the Apple-compatible NCS 6502. 030-715004	Wilt U op de hoogte blijven? Schrijf dan aan Teleac, Post bus 2414, 3500 BK Utrecht, of bel 030 - 956248/956911.
	COMAL*.KGN voor JUNIOR verkrijgbaar op band met bekende handleiding. In: redaktie.	Kempac Electronic Engineering De Houtkoperstraat 16, 3334 KD Zwijndrecht, The Netherlands presents the KEMPAC System 4 modularly built microcomputer, with Z80, 6808, 6809 or 6502. 078-100300	

PAGE 01

```

0010: PROGRAM CASSETTE 1
0020:
0030: (C) AD BROUWER
0040:
0050: In deze routines wordt de mode geselecteerd waarmee
0060: gelezen of geschreven wordt. Vanuit deze routines worden
0070: de echte lees/schrijf I/O routines aangeroepen. Echter
0080: als er een fout wordt gedetekteerd dan wordt er
0090: teruggekeerd naar deze routines.
0100: Er zijn twee foutmeldingen:
0110:
0120: - ERROR TAPE Deze melding volgt nadat de checksum niet
0130: kloopt of nadat er gebreukt is tijdens het
0140: lezen of (alleen bij LOAD) nadat er een
0150: niet toegelaten byte is ingelezen.
0160: Hierna wordt er gewacht op een toets. CR (OD) start
0170: het lezen opnieuw. Een andere toets stopt het lezen
0180: en er wordt geprint hoeveel er is gelezen. Hierna
0190: volgt return met Z=0 en C=0.
0200:
0210: - ERROR MEMORY Deze melding volgt nadat het maximale
0220: eindadres is overschreden.
0230: Er wordt hierna geprint hoeveel er is gelezen waarna
0240: return volgt met Z=0 en C=1.
0250:
0260: Als het lezen goed ging wordt er geprint hoeveel er
0270: gelezen is waarna return volgt met Z=1 en C=0.
0280:
0290: Als na de LOAD I/O routine de V vlag staat (bij de andere
0300: I/O routines wordt V altijd 0 gemaakt), betekent dit dat
0310: er een file is gevonden met een verkeerd label. LOAD
0320: print dit nummer waarna er opnieuw gestart wordt.
0330:
0340: De routines gebruiken een tweetal monitor subroutines
0350: waarbij vereist is dat het X en Y register hetzelfde
0360: blijven.
0370:
0380: De juiste lees schrijf mode wordt gekozen door in MODE
0390: een aantal bits te zetten.
0400:
0410: Indien nodig staat in TPBG het beginadres en in TPEN het
0420: eindadres.
0430: Er wordt geschreven en gelezen van TPBG tot en met TPEN.
0440:
0450: Bij JUNIOR lees/schrijf staat het label in LABL.
0460:
0470: Om een maximaal (DFFF?) eindadres bij het lezen te
0480: specificeren kan, voordat de eigenlijke routine begint,
0490: een routine LDEN worden aangeroepen die DFFF in TPEN
0500: plaatst.
0510:
0520: Aan het begin staan vier jumps naar deel 2 van het
0530: cassette programma. Dit deel 2 bevat de routines die de
0540: I/O verrichten. Hierna staan twee jumps naar monitor
0550: routines.
0560:
0570: Een tweetal routines zijn er voor JUNIOR formaat.
0580: De andere twee zijn er voor APPLE formaat indien men
0590: de cassette interface heeft die aan de VIA zit.
0600:
0610: Om een onderscheid te maken benoem ik de JUNIOR routines
0620: met LOAD en SAVE. Voor APPLE geldt READ en WRITE.
0630:
0640: Voor LOAD geldt:

```

PAGE 02

```

0650:      - hex      (mode 00) zoals standaard junior
0660:      - hex      (mode 40) negeert startadres van cassette
0670:      -          (mode 80) controleert max eindadres
0680:      - ascii (mode 80) leest asciifile
0690:      -          (mode 80) controleert max eindadres
0700:
0710: Voor SAVE geldt:
0720:      - hex      (mode 00) zoals standaard junior
0730:      - ascii (mode 80) schrijft ascii formaat
0740:
0750: Voor READ geldt:
0760:      -          (mode 00) READ controleert altijd het
0770:      -          (mode 00) max eindadres.
0780:      - stle      (mode C0) haalt eerst het startadres en
0790:      -          (mode C1) de lengte van cassette waarna
0800:      -          (mode C1) de (hex) file gelezen wordt.
0810:      - stle      (mode C1) gelijk mode C0 echter het startadres
0820:      -          (mode 80) van cassette wordt genegeerd
0830:      - leng      (mode 80) leest eerst de lengte en daarna
0840:      -          (mode 00) de (ascii) file.
0850:      - blok      (mode 00) leest een blok in waarbij de grenzen
0860:      -          (mode 00) in TPBG en TPEN staan.
0870:
0880: Voor WRITE geldt:
0890:      - stln      (mode C0) schrijft eerst het startadres en de
0900:      -          (mode 80) lengte weg waarna de (hex) file volgt.
0910:      - leng      (mode 80) schrijft lengte weg waarna de (ascii)
0920:      -          (mode 00) file volgt.
0930:      - blok      (mode 00) schrijft blok weg van TPBG t/m TPEN.
0940:
0950: Na afloop staat in TPBG het beginadres en in TPEN het
0960: eindadres. Voor LOAD geldt nog dat als er wordt geladen met
0970: LABEL = 00 of FF dat dan het gevonden label in LABEL staat.
0980:
0990: Om de adressen die in een ander programma staan (b.v. M.A.)
1000: niet bij elke verandering te hoeven wijzigen, starten alle
1010: routines vlak na een aantal jumps.
1020:
1030: Redenen om op apple formaat te lezen en te schrijven zijn
1040: de snelheid en de betrouwbaarheid. De snelheid is gemiddeld
1050: 1333 baud waarbij er synchroon wordt weggeschreven (geen
1060: start of stopbits). 8k weg schrijven duurt ongeveer 50 sec.
1070: De snelheid is niet absoluut gegeven daar de tijd die nodig
1080: is om een 1 weg te schrijven langer is dan voor een 0.
1090: De foutkans is afhankelijk van de cassette en recorder.
1100: Deze kans is mijns inziens klein.
1110:
1120: In CASSETTE 2 wordt er verder op in gegaan over de hardware
1130: die nodig is.
1140:
1150: Ik stel voor om het programma voorlopig op 4000 te laten
1160: starten en later in eorum te zetten op de plaats van de
1170: oude TM routines (0800) die dan overbodig zijn.
1180:
1190: Voor diegene die de apple routines niet willen hebben,
1200: kunnen dat stuk programma weglaten en dan de source
1210: opnieuw assembleren.
1220:
1230: 4000          ORG      $4000
1240:
1250:          FF DF  ENDAD  *      $DFFF  eindadres
1260:
1270:          F9 00  MODE   *      $00F9  mode
1280:          FA 00  TPBG   *      $00FA  beginadres

```

```

1290:      FC 00 TPEN *      $00FC eindadres
1300:      FE 00 LABL *      $00FE label
1310:
1320: 4000 4C 00 42 LOAD JMP $4200 load i/o lees junior
1330: 4003 4C 03 42 SAVE JMP $4203 save i/o schrijf junior
1340: 4006 4C 06 42 READ JMP $4206 read i/o lees apple
1350: 4009 4C 09 42 WRIT JMP $4209 writ i/o schrijf apple
1360:
1370: 400C 4C 0C F0 INCH JMP $F00C haal karakter van terminal
1380: 400F 4C 0F F0 OUTCH JMP $F00F zend karakter naar terminal
1390:
1400:      junior load
1410:      4012 A9 00 LDAIM $00 mode 00 hex
1420:      4014 F0 0C BEQ LODAA
1430:      4016 20 D3 40 JSR LDEN
1440:      4019 A9 40 LDAIM $40 mode 40 hex
1450:      401B D0 05 BNE LODAA
1460:      401D 20 D3 40 JSR LDEN
1470:      4020 A9 80 LDAIM $80 mode 80 ascii
1480:      4022 85 F9 LODAA STAZ MODE
1490:      4024 4C 8B 40 JMP LOD
1500:
1510:      junior save
1520:      4027 A9 00 LDAIM $00 mode 00 hex
1530:      4029 F0 02 BEQ SAVAA
1540:      402B A9 80 LDAIM $80 mode 80 ascii
1550:      402D 85 F9 SAVAA STAZ MODE
1560:      402F 4C 86 40 JMP SAV
1570:
1580:      apple read
1590:      4032 20 D3 40 JSR LDEN
1600:      4035 A9 C0 LDAIM $C0 mode C0 startadres
1610:      4037 D0 10 BNE REDK lengte
1620:      4039 20 D3 40 JSR LDEN
1630:      403C A9 C1 LDAIM $C1 mode C1 zie C0
1640:      403E D0 09 BNE REDK
1650:      4040 20 D3 40 JSR LDEN
1660:      4043 A9 80 LDAIM $80 mode 80 lengte
1670:      4045 D0 02 BNE REDK
1680:      4047 A9 00 LDAIM $00 mode 00 blok
1690:      4049 85 F9 REDK STAZ MODE
1700:      404B 4C 5F 40 JMP RED
1710:
1720:      apple write
1730:      404E A9 C0 LDAIM $C0 mode C0 startadres
1740:      4050 D0 06 BNE WRTK lengte
1750:      4052 A9 80 LDAIM $80 mode 80 lengte
1760:      4054 D0 02 BNE WRTK
1770:      4056 A9 00 LDAIM $00 mode 00 blok
1780:      4058 85 F9 WRTK STAZ MODE
1790:
1800:      schrijft een file weg volgens apple formaat:
1810:
1820:      405A 20 09 40 JSR WRIT start WRITE
1830:      405D 50 1F BVC REDR
1840:
1850:      leest een file in volgens apple formaat:
1860:
1870:      405F A5 FC RED LDAZ TPEN start READ
1880:      4061 48 PHA bewaar eind
1890:      4062 A5 FD LDAZ TPEN +01
1900:      4064 48 PHA
1910:      4065 20 06 40 JSR READ
1920:      4068 B0 0F BCS REDMER memory error

```

```

1930: 406A F0 10      BEQ  REDRD
1940: 406C 20 12 41   JSR  PRERTP error tape
1950: 406F D0 0B      BNE  REDRD stop lezen
1960: 4071 68         PLA
1970: 4072 85 FD      STAZ TPEN haal eind terug
1980: 4074 68         PLA
1990: 4075 85 FC      STAZ TPEN
2000: 4077 B0 E6      BCS  RED
2010: 4079 20 23 41   REDMER JSR  PRERME error memory
2020: 407C 68         REDRD PLA haal stack terug
2030: 407D 68         PLA
2040: 407E 20 EB 40   REDR JSR  CRLF
2050: 4081 20 2E 41   JSR  PRBGEN print kop
2060: 4084 F0 3C      BEQ  LDT
2070:
2080: schrijft een file weg volgens junior formaat:
2090:
2100: 4086 20 03 40   SAV  JSR  SAVE start SAVE
2110: 4089 50 2C      BVC  LODD
2120:
2130: leest een file in volgens junior formaat:
2140:
2150: 408B A2 02      LOD  LDXIM $02 start LOAD
2160: 408D B5 FC      LDBB LDAZX TPEN zet eind en
2170: 408F 48         PHA label op stack
2180: 4090 CA         DEX
2190: 4091 10 FA      BPL  LDBB
2200: 4093 20 00 40   LDB  JSR  LOAD laad
2210: 4096 50 06      BVC  LDD
2220: 4098 20 E3 40   JSR  PRHA print verkeerd label in A
2230: 409B 4C 93 40   JMP  LDB
2240: 409E B0 11      LDD  BCS  LODC memory overflow
2250: 40A0 F0 12      BEQ  LODDB
2260: 40A2 20 12 41   JSR  PRERTP error tape
2270: 40A5 D0 0D      BNE  LODDB geen CR ) stop
2280: 40A7 A2 FD      LDXIM $FD haal eind en label terug
2290: 40A9 68         LDBC PLA
2300: 40AA 95 FF      STAZX TPEN +03
2310: 40AC E8         INX
2320: 40AD D0 FA      BNE  LDBC
2330: 40AF F0 DA      BEQ  LOD
2340: 40B1 20 23 41   LODC JSR  PRERME error memory
2350: 40B4 68         LODDB PLA haal stack terug
2360: 40B5 68         PLA
2370: 40B6 68         PLA
2380: 40B7 20 EB 40   LODD JSR  CRLF
2390: 40BA 20 2A 41   JSR  PRLABL print kop
2400: 40BD A5 FE      LDAZ LABL print label
2410: 40BF 20 E3 40   JSR  PRHA
2420: 40C2 A2 FA      LDT  LDXIM TPBG print begin
2430: 40C4 20 DC 40   JSR  PRHB
2440: 40C7 A2 FC      LDXIM TPEN print eind
2450: 40C9 20 DC 40   JSR  PRHB
2460: 40CC 20 EB 40   JSR  CRLF
2470: 40CF 98         TYA zet status terug
2480: 40D0 48         PHA
2490: 40D1 28         PLP
2500: 40D2 60         RTS eind alle routines
2510:
2520: gebruikte subroutines
2530:
2540: 40D3 A9 FF      LDEN LDAIM ENDAD om eindadres te
2550: 40D5 85 FC      STAZ TPEN specificeren
2560: 40D7 A9 DF      LDAIM ENDAD /

```

```

2570: 40D9 85 FD      STAZ  TPEN  +01
2580: 40DB 60          RTS
2590:
2600: 40DC B5 01      PRHB  LDAZX $01  print adres
2610: 40DE 20 F8 40    JSR  PRHEXA
2620: 40E1 B5 00      LDAZX $00
2630: 40E3 20 F8 40    PRHA  JSR  PRHEXA
2640: 40E6 A9 20      OUTSPA LDAIM $20  print soatie
2650: 40E8 4C OF 40    JMP  OUTCH
2660:
2670: 40EB 48          CRLF  PHA      carriage return
2680: 40EC A9 0D      LDAIM $0D  line feed
2690: 40EE 20 OF 40    JSR  OUTCH
2700: 40F1 A9 0A      LDAIM $0A
2710: 40F3 20 OF 40    JSR  OUTCH
2720: 40F6 68          PLA
2730: 40F7 60          RTS
2740:
2750: 40F8 48          PRHEXA PHA      print hex A
2760: 40F9 4A          LSRA
2770: 40FA 4A          LSRA
2780: 40FB 4A          LSRA
2790: 40FC 4A          LSRA
2800: 40FD 20 01 41    JSR  PRNIBA
2810: 4100 68          PLA
2820: 4101 29 OF      PRNIBA ANDIM $0F
2830: 4103 C9 0A      CMPIM $0A
2840: 4105 90 02      BCC  PRNA
2850: 4107 69 06      ADCIM $06
2860: 4109 69 30      PRNA  ADCIM '0
2870: 410B 4C OF 40    JMP  OUTCH
2880:
2890: 410E A2 00      ERROR  LDXIM $00  error
2900: 4110 F0 1E      BEQ
2910: 4112 20 0E 41    PRERTP JSR  ERROR  tape
2920: 4115 A2 0A      LDXIM $0A
2930: 4117 20 30 41    JSR  PRINT
2940: 411A 20 0C 40    JSR  INCH  wacht toets
2950: 411D 20 EB 40    JSR  CRLF
2960: 4120 C9 0D      CMPIM $0D  om oonieuw te
2970: 4122 60          RTS      lezen
2980: 4123 20 0E 41    PRERME JSR  ERROR  memory
2990: 4126 A2 0F      LDXIM $0F
3000: 4128 D0 06      BNE  PRINT
3010: 412A A2 18      PRLABL LDXIM $18  1b
3020: 412C D0 02      BNE  PRINT
3030: 412E A2 1B      PRBGEN LDXIM $1B  beq end
3040: 4130 BD 3C 41    PRINT LDAAX TABL
3050: 4133 F0 06      BEQ  PRRT
3060: 4135 20 OF 40    JSR  OUTCH
3070: 4138 E8          INX
3080: 4139 D0 F5      BNE  PRINT
3090: 413B 60          RTS
3100: 413C 0D          PRRT  TABL  = $0D  error
3110: 413D 0A          = $0A
3120: 413E 07          = $07  bell
3130: 413F 45          = 'E
3140: 4140 52          = 'R
3150: 4141 52          = 'R
3160: 4142 4F          = 'O
3170: 4143 52          = 'R
3180: 4144 20          =
3190: 4145 00          = $00
3200: 4146 54          = 'T  0A tape

```

```

3210: 4147 41      =      'A
3220: 4148 50      =      'P
3230: 4149 45      =      'E
3240: 414A 00      =      $00
3250: 414B 4D      =      'M      OF memory
3260: 414C 45      =      'E
3270: 414D 4D      =      'M
3280: 414E 4F      =      'O
3290: 414F 52      =      'R
3300: 4150 59      =      'Y
3310: 4151 0D      =      $0D
3320: 4152 0A      =      $0A
3330: 4153 00      =      $00
3340: 4154 4C      =      'L      1B 1b
3350: 4155 42      =      'B
3360: 4156 20      =      '
3370: 4157 20      =      '      1B beg end
3380: 4158 42      =      'B
3390: 4159 45      =      'E
3400: 415A 47      =      'G
3410: 415B 20      =      '
3420: 415C 20      =      '
3430: 415D 45      =      'E
3440: 415E 4E      =      'N
3450: 415F 44      =      'D
3460: 4160 07      =      $07      bell
3470: 4161 0D      =      $0D
3480: 4162 0A      =      $0A
3490: 4163 00      =      $00
3500:

```

MICRO-Ade Assembler/Disassembler/Texteditor for 6502 (8K)

SYSTEM DESCRIPTION

The Micro-ADE system is designed for use with any 6502 microcomputer and consist of three major programs as well as a number of utility programs. The major programs are an assembler, a disassembler, and a text editor.

The assembler is used to create machine executable code for the 6502 from a symbolic input source program. Small programs can be created and tested directly in memory. Larger programs may be written using cassette tapes for source input and object output.

The disassembler is used to list executable 6502 machine code in the symbolic assembler source format. Symbols are generated if they are defined in the symbol table.

The text editor is used to create source programs in the format required for the assembler. It contains the necessary routines for easy manipulation of text data in memory or from cassette files.

The minimum system configuration for full use of all Micro-ADE features consists of a 6502 CPU, 8K of random access memory, 2 cassette recorders with start/stop control, and an ASCII input/output device. It is possible to use all parts of the system in a restricted way with less memory and a single manually operated cassette recorder.

For some months past our club has obtained copyrights on Micro-ADE of Peter Jennings from Micro-Ware Ltd. We are allowed to place Micro-ADE at the disposal of the members of our club. The original 4K Micro-Ade has been patched up to 8K by Sebo Woldrinch, with commands like Append Source Files, Insert Lines, Set/Reset Form Feed Flag, Chosse Lines, List, Move Lines, Load ASCII-format Files, Set/Reset Page Mode, List Used Memory, Duplicate Files, Save Source Files, Print Symbol/XREF-tables, Set/Reset Page-per-File/EJECT Flag, Search Line, Print/Change String, Assemble/Execute, Set Lines/Screen and Lines/Page, Disassemble, List without line numbers, Save chosen Lines, Save ASCII-format Files, etc.

From these patches W.L. van Pelt composed an 8K source-listing, complete with comments. Manual+Command Review: Fl. 35.00, Source Listing 8K-version: Fl. 65.00. Original 4K Source-Listing: Fl. 30.00. Cassettes in KIM/JUNIOR hypertext: 4K KIM = Fl. 12.50, 8K KIM = Fl. 12.50, 8K JUNIOR = Fl. 12.50. To be paid with Eurocheque, or on the postal account 841433 from W.L. van Pelt, Krimpen a.d. IJssel, The Netherlands. No other payments allowed.


```

4020
4025 ; schuiven of springen
4030 *****
4035
4040
4045 assembler deel 2
4050
4055
4060
4065 .ba $24cd
4070 .mc $94cd
4075 .os
4080 .ct
4085 text2 .de $01 ; hob (text)
4090 free1 .de $05 ; lege plaats
4095 free2 .de $06 ; vaste idem
4100 who1 .de $07 ; wie speelt
4105 who2 .de $08 ; zet idem
4110 work1 .de $0b ; algem. buffer
4115 work2 .de $0c ; algem. buffer
4120 input .de $0e ; inputcoord.
4125 copyvd .de $10 ; werkspeelveld
4130 savecp .de $1a ; save werkveld
4135 sava .de $1f ; bewaar accu
4140 savx .de $2f ; bewaar x-reg.
4145 kern .de $37 ; middenpunt veld
4150 savy .de $3f ; bewaar y-reg.
4155 savf .de $4f ; bewaar flag-reg.
4160 limit .de $79 ; midden vidio
4165 chastr .de $7a ; alfa num. vidio
4170 piostr .de $7b ; rasterwaarde vid
4175 totlin .de $7c ; lijn of tot. vid
4180 lincom .de $7d ; count/comp vidio
4185 aria .de $7e ; begrenzer vidio
4190 cursv .de $80 ; data cursor vert
4195 cursh .de $85 ; idem horizontaal
4200 zpoint .de $a1 ; zetten
4205 prcha .de $1334 ; output char.
4210 ;
4215 clean .de $234e ; sr wis scherm
4220 home .de $2364 ; sr cursor home
4225 text .de $2368 ; sr plaats tekst
4230 texta .de $236a ; sr plaats tekst
4235 vidsp .de $239f ; sr soatie
4240 crlf .de $23a4 ; sr cr+lf
4245 ;
4250 ;
4255 ;
4260 ; toon verplaatsing pion op scherm
4265 ;
4270 vidzet jsr home ; cursor home
4275 lda #h, piohi ; hob textdata
4280 sta *text2
4285 lda #$04 ; counter .van.
4290 sta *work2 ; naar .hier.
4295 vipa lda *free2 ; coord. lege
4300 vipb pha ; plaats
4305 and #$0f
4310 tay
4315 ldx *cursh,y ; cursordata voor
4319
4320 pla ; horz. beweging
4325 lsr a
4330 lsr a
4335 lsr a
4340 lsr a
4345 tay
4350 lda cursv,y ; cursordata voor
4355 tay ; vert. beweging
4360 dec *work2 ; counter oneven=
4365 ; plaats tekst .van.
4370 lda *work2 ; counter even=
4375 ; plaats tekst .hier.
4380 bmi vipc ; ff=einde
4385 and #$01
4390 beq vipd

```

24F4-	20	3A	25	4395	jsr fromz	; tekst .van.
24F7-	F0	05		4400	beq vipe	
24F9-	20	12	25	4405 vipd	jsr tozet	; tekst .hier.
24FC-	F0	DA		4410	beq vipa	
24FE-	A5	05		4415 vipe	lda *free1	; zetcoördinaat
2500-	10	D8		4420	bpl vipb	
2502-	20	41	25	4425 vipc	jsr posit	; curs. plaatsen
2505-	20	9F	23	4430	jsr vidsp	; spatie
2508-	A9	2B		4435	lda #h,piocl	; lege plaats
250A-	A0	B5		4440	ldy #1,piocl	; aanduiden
250C-	20	68	23	4445	jsr text	
250F-	4C	64	23	4450	jmp home	; cursor home
2512-	20	41	25	4455 tozet	jsr posit	; curs. richten
2515-	A0	B0		4460	ldy #1,piohi	; tekst .hier.
2517-	20	6A	23	4465 toza	jsr texta	
251A-	A0	B5		4470	ldy #1,piocl	; tekst ver-
251C-	20	6A	23	4475 tozb	jsr texta	; wijderen
251F-	E6	0B		4480	inc *work1	; counter +1
2521-	30	01		4485	bmi tozc	; pl. pion terug
2523-	60			4490	rts	
2524-	A5	07		4495 tozc	lda *whol	; soeelzin
2526-	10	06		4500	bpl tozd	
2528-	A5	0E		4505	lda *inout	; wiens zetbe-
252A-	10	06		4510	bpl tozf	; weging?
252C-	30	08		4515	bmi tozg	
252E-	A5	0E		4520 tozd	lda *input	
2530-	10	04		4525	bpl tozg	
2532-	A0	C2		4530 tozf	ldy #1,uoppio	; plaats .!.
2534-	D0	E6		4535	bne tozb	
2536-	A0	C6		4540 tozg	ldy #1,dwnpio	; plaats .*.
2538-	D0	E2		4545	bne tozb	
253A-	20	41	25	4550 fromz	jsr posit	; curs. richten
253D-	A0	CA		4555	ldy #1,fropio	; tekst .van.
253F-	D0	D6		4560	bne toza	
2541-	A9	0A		4565 posit	lda #\$0A	; lf. volgens data
2543-	20	34	13	4570	jsr prcha	
2546-	88			4575	dex	
2547-	D0	F8		4580	bne posit	
2549-	A9	09		4585 posht	lda #\$09	; ht. volgens data
254B-	20	34	13	4590	jsr prcha	
254E-	CA			4595	dex	
254F-	10	F8		4600	bpl posht	
2551-	CA			4605	dex	
2552-	86	0B		4610	stx *work1	; counter=fe
2554-	60			4615	rts	
				4619		
				4620		
				4625	zoek 9 mogelijke zetten	
				4630	t.o.v. lege plaats	
				4635		
2555-	20	76	26	4640 zet	jsr utosa	; save registers
2558-	A2	08		4645 zepass	ldx #\$08	; counter
255A-	86	0C		4650	stx *work2	
255C-	A9	FF		4655	lda #\$ff	
255E-	95	A1		4660 zenxt	sta *zpoint,x	; set buffers
2560-	CA			4665	dex	
2561-	10	FB		4670	bpl zenxt	
2563-	A6	05		4675	ldx *free1	; zetcoördinaat
2565-	A0	01		4680	ldy #\$01	; counter
2567-	CA			4685 vnxt	dex	
2568-	20	DF	25	4690	jsr zero	; rij links
256B-	F0	03		4695	beq tloa	
256D-	20	01	26	4700	jsr contr	; aanwezig?
2570-	C0	00		4705 tloa	cpy #\$00	; hoger of lager
2572-	D0	12		4710	bne tloc	
2574-	CA			4715	dex	
2575-	20	DF	25	4720	jsr zero	; rij links
2578-	F0	03		4725	beq tlob	
257A-	20	01	26	4730	jsr contr	; aanwezig?
257D-	E8			4735 tlob	inx	
257E-	20	DF	25	4740	jsr zero	; linkser
2581-	F0	03		4745	beq tloc	
2583-	20	C2	25	4750	jsr srwho	; paardespr. links
2586-	E8			4755 tloc	inx	
2587-	E8			4760	inx	
2588-	20	E9	25	4765	jsr one	; rij rechts
258B-	10	03		4770	bpl tlod	
258D-	20	01	26	4775	jsr contr	; aanwezig?

```

2590- C0 00      4780 tlod      cpy #$00
2592- D0 12      4785      bne tlof      ; hoger of lager
2594- E8          4790      inx
2595- 20 E9 25    4795      jsr one      ; rij rechts
2598- 10 03      4800      bol tloe
259A- 20 01 26    4805      jsr contr   ; aanwezig?
259D- CA          4810 tloe      dex
259E- 20 E9 25    4815      jsr one      ; rechtser
25A1- 10 03      4820      bol tlof
25A3- 20 C2 25    4825      jsr srwho   ; paardspr. rechts
25A6- CA          4830 tlof      dex
25A7- C0 00      4835      cpy #$00
25A9- F0 14      4840      beq gbrb     ; einde?
25AB- A5 08      4845      lda *who2    ; soelrichting
25AD- D0 0B      4850      bne wjmp
25AF- 20 EF 25    4855      jsr two     ; omhoog
25B2- 10 0B      4860      bol gbrb
25B4- 20 01 26    4865 gbra      jsr contr   ; aanwezig?
25B7- 88          4870      dey      ; verlaat rij
25B8- F0 AD      4875      beq vnxt
25BA- 20 F9 25    4880 wjmp      jsr three    ; omlaag
25BD- 10 F5      4885      bol gbra
25BF- 4C 83 26    4890 gbrb      jmp ufrsa    ; haal req. op
25C2- A5 08      4895 srwho      lda *who2    ; soelrichting
25C4- D0 0D      4900      bne dbra
25C6- 20 EF 25    4905      jsr two     ; omhoog
25C9- 10 03      4910      bol dbrb
25CB- 20 01 26    4915      jsr contr
25CE- 20 F9 25    4920      jsr three    ; omlaag
25D1- 10 0B      4925      bol dbrd
25D3- 20 F9 25    4930 dbra      jsr three    ; omlaag
25D6- 30 03      4935      bmi dbrb
25D8- 20 01 26    4940      jsr contr
25DB- 20 EF 25    4945      jsr two     ; omhoog
25DE- 60          4950      rts
25DF- 8A          4955 zero      txa      ; naar links
25E0- 29 0F      4960      and #$0f
25E2- C9 0F      4965      cmp #$0f
25E4- F0 02      4970      beq heok
25E6- C9 0E      4975      cmp #$0e
25E8- 60          4980 heok      rts
25E9- 8A          4985 one       txa      ; naar rechts
25EA- 29 0F      4990      and #$0f
25EC- C9 05      4995      cmp #$05
25EE- 60          5000      rts
25EF- 8A          5005 two       txa      ; naar omlaag
25F0- 18          5010      clc
25F1- 69 10      5015      adc #$10
25F3- AA          5020      tax
25F4- 29 F0      5025      and #$f0
25F6- C9 50      5030      cmp #$50
25F8- 60          5035      rts
25F9- 8A          5040 three     txa      ; naar omhoog
25FA- 38          5045      sec
25FB- E9 10      5050      sbc #$10
25FD- AA          5055      tax
25FE- 29 F0      5060      and #$f0
2600- 60          5065      rts
2601- A5 08      5070 contr      lda *who2    ; wie soeelt
2603- F0 04      5075      beq vwha
2605- A9 26      5080      lda #$26    ; pion waarde
2607- D0 02      5085      bne vwbb
2609- A9 23      5090 vwba      lda #$23
260B- D5 10      5095 vwbb      cmp *copyvd,x ; is zet wel
260D- F0 01      5100      beq vwbc    ; aanwezig
260F- 60          5105      rts
2610- 8A          5110 vwbc      txa      ; zoja, plaats zet-
2611- A6 0C      5115      ldx *work2  ; coord. in buffer
2613- 95 A1      5120      sta *zpoint,x
2615- AA          5125      tax
2616- C6 0C      5130      dec *work2  ; counter -1
2618- 60          5135      rts
2619- 20 76 26    5140      ;
261C- A2 05      5145      ; copie van ram-veld naar werkveld
261E- D0 0E      5150      ;
2619- 20 76 26    5155 copy      jsr utosa    ; bewaar registers
261C- A2 05      5160      ldx #$05    ; pointer .from.
261E- D0 0E      5165      bne coya

```

```

5170 :
5175 :   copie van werkveld naar bewaar-veld
5180 :
2620- 20 76 26 5185 copsave   jsr utosa
2623- A2 00      5190      ldx #$00      ; pointer .from.
2625- A0 0A      5195      ldy #$0A      ; pointer .to.
2627- D0 07      5200      bne copyb
5205 :
5210 :   herstel werkveld vanuit bewaar-veld
5215 :
5219 :
2629- 20 76 26 5220 restcop   jsr utosa
262C- A2 0A      5225      ldx #$0A      ; pointer .from.
262E- A0 00      5230 cpya      ldy #$00      ; pointer .to.
2630- A9 04      5235 copyb     lda #$04      ; counter kolom
2632- 85 0C      5240      sta *work2
2634- B5 10      5245 cpyc      lda *copyvd,x      ; copieer
2636- 99 10 00    5250      sta copyvd,y
2639- E8          5255      inx      ; pointer kolom +1
263A- C8          5260      iny
263B- C6 0C      5265      dec *work2      ; counter kolom -1
263D- 10 F5      5270      bpl cpyc
263F- 8A          5275      txa
2640- 18          5280      clc      ; volgende rij .from.
2641- 69 0B      5285      adc #$0B
2643- AA          5290      tax
2644- 98          5295      tya
2645- 18          5300      clc      ; volgende rij .to.
2646- 69 0B      5305      adc #$0B
2648- A8          5310      tay
2649- 29 F0      5315      and #$F0
264B- C9 50      5320      cmp #$50      ; einde copie?
264D- D0 E1      5325      bne copyb
264F- 4C 83 26    5330      jmp ufrsa      ; haal req. op
5335 :
5340 :   haal tegenzetten voor beweging van
5345 :   onder naar boven
5350 :
2652- A0 23      5355 putup     ldy #$23
2654- D0 02      5360      bne putbne
5365 :
5370 :   haal tegenzetten voor beweging van
5375 :   boven naar onder
5380 :
2656- A0 26      5385 putdwn   ldy #$26
2658- 20 76 26 5390 putbne   jsr utosa
265B- 48          5395      pha      ; reset zetcoördinaat
265C- AA          5400      tax      ; in copyveld
265D- A9 00      5405      lda #$00
265F- 95 10      5410      sta *copyvd,x
2661- 98          5415      tya
2662- A6 05      5420      ldx *free1      ; plaats volgens
2664- 95 10      5425      sta *copyvd,x      ; richting op leeg
2666- 68          5430      pla      ; veld pionwaarde
2667- 85 05      5435      sta *free1
2669- A9 00      5440      lda #$00      ; richting bepalen
266B- C0 26      5445      cpy #$26      ; voor tegenzetten
266D- F0 02      5450      beq puta
266F- 49 FF      5455      eor #$FF
2671- 85 08      5460 puta      sta *who2
2673- 4C 58 25    5465      jmp zepass      ; haal tegenz. op
2676- 85 1F      5470 utosa     sta *sava      ; bewaar registers
2678- 86 2F      5475      stx *savx
267A- 84 3F      5480      sty *savy
267C- 08          5485      php
267D- 68          5490      pla
267E- 85 4F      5495      sta *savf
2680- A5 1F      5500      lda *sava
2682- 60          5505      rts
2683- A5 4F      5510 ufrsa     lda *savf      ; haal reg. op
2685- 48          5515      pha
5519 :
2686- 28          5520      plp
2687- A5 1F      5525      lda *sava
2689- A6 2F      5530      ldx *savx
268B- A4 3F      5535      ldy *savy

```

```

268D- 60      5540      rts
                5545      ;
                5550      ; op scherm brengen bovenste regel,
                5555      ; of het ganse speelbord tonen
                5560      ;
268E- A9 01    5565      rasta      lda #$01      ; 1 lijn op scherm
2690- D0 05    5570      bne rasc
2692- 20 4E 23 5575      rastb      jsr clean
2695- A9 10    5580      lda #$10      ; volledig raster
2697- 85 7C    5585      rasc      sta *totlin ; tot. lijn(en)
2699- A9 FF    5590      lda #$ff
269B- 85 7E    5595      sta *aria      ; cijferplaatsen
269D- 85 79    5600      sta *limit      ; middenpunt rast.
269F- A9 0E    5605      lda #$0e
26A1- 85 7D    5610      sta *lincom ; vergelijker
26A3- A9 41    5615      lda #$41
26A5- 85 7A    5620      sta *chastr ; letterwaarde
26A7- A9 2E    5625      lda #$2e
26A9- 85 7B    5630      sta *piostr ; char.waarde
                5635      ;
                5640      ; 3 spaties en/of cursor home of
                5645      ; cijfer + 2 spaties
                5650      ;
26AB- A2 03    5655      rastd      ldx #$03      ; counter
26AD- A5 7E    5660      lda *aria
26AF- F0 0F    5665      beq rasfia ; 00= 3 spaties
26B1- 10 05    5670      bpl rasfib ; pos.=cijfer +
                5675      ; twee soatie
26B3- 20 64 23 5680      jsr home ; curs. home+3 sp.
26B6- 50 08    5685      bvc rasfia
26B8- A5 7A    5690      rasfib      lda *chastr ; plaats cijfer
26BA- 20 34 13 5695      jsr prcha
26BD- E6 7A    5700      inc *chastr ; volgend cijfer
26BF- CA      5705      dex
26C0- 20 9F 23 5710      rasfia      jsr vidsp ; soatie
26C3- CA      5715      dex
26C4- D0 FA    5720      bne rasfia
26C6- A0 05    5725      ldy #$05 ; counter
26C8- A9 2E    5730      rassa      lda #$2e
26CA- 20 34 13 5735      jsr prcha ; plaats punt
26CD- A5 7A    5740      lda *chastr
26CF- C9 46    5745      cmp #$46 ; einde letters
26D1- 30 04    5750      bmi rassb
26D3- A9 31    5755      lda #$31 ; cijferwaarde
26D5- D0 0B    5760      bne rassc
26D7- C9 41    5765      rassb      cmp #$41 ; reeds cijfers?
26D9- 30 09    5770      bmi rasse
26DB- 20 34 13 5775      jsr prcha ; plaats letter
26DE- E6 7A    5780      inc *chastr ; volgende letter
26E0- D0 05    5785      bne rassd
26E2- 85 7A    5790      rassc      sta *chastr ; zet cijfer weg
26E4- 20 9F 23 5795      rasse      jsr vidsp ; soatie
26E7- A2 03    5800      rassd      ldx #$03 ; counter
26E9- 20 41 27 5805      rassf      jsr vidsp ; welk char.?
26ED- CA      5810      dex
26ED- D0 FA    5815      bne rassf
                5819
26EF- A5 79    5820      lda *limit ; reeds midden?
26F1- 30 12    5825      bmi rasma
26F3- C0 04    5830      cpy #$04 ; reeds over helft
26F5- F0 0A    5835      beq rasmb
26F7- 10 0C    5840      bpl rasma
26F9- 49 FF    5845      eor #$ff
26FB- 85 79    5850      sta *limit ; set limiet
26FD- A9 2A    5855      lda #$2a ; char.waarde (*)
26FF- D0 02    5860      bne rasmc
2701- A9 20    5865      rasmb      lda #$20 ; char.waarde ( )
2703- 85 7B    5870      rasmc      sta *piostr
2705- 88      5875      rasma      dey ; afgewerkt?
2706- D0 C0    5880      bne rassa
2708- A9 2E    5885      lda #$2e ; eindpunt rij
270A- 20 34 13 5890      jsr prcha
270D- C6 7C    5895      dec *totlin ; tot. afgewerkt?
270F- D0 03    5900      bne rasend
2711- 4C 64 23 5905      jmp home ; cursor home
2714- 20 A4 23 5910      rasend      jsr crlf ; volgende rij
2717- A5 7D    5915      lda *lincom ; vergelijk met

```

2719-	C5 7C	5920	cmp #totlin	; lijncounter
271B-	F0 0B	5925	beq raslin	
271D-	A9 00	5930	lda #\$00	; begin rij met
271F-	85 7E	5935	sta *aria	; 3 spaties
2721-	A9 20	5940	lda #\$20	; char.waarde ()
2723-	85 7B	5945	sta *piostr	
2725-	4C AB	5950	jmp rastd	; rij afwerken
2728-	85 7E	5955	sta *aria	; cijfer + 3 sp.
272A-	38	5960	sec	
272B-	E9 03	5965	sbx #\$03	; vergelijker -3
272D-	85 7D	5970	sta *lincom	
272F-	A5 7C	5975	lda *totlin	; helft afgewerkt
2731-	C9 08	5980	cmp #\$08	
2733-	30 08	5985	bmi rasmi	
2735-	D0 02	5990	bne raspl	
2737-	85 79	5995	sta *limit	; middenste rij
2739-	A9 21	6000	lda #\$21	; pionwaarde (!)
273B-	D0 E6	6005	bne rasnxt	
273D-	A9 2A	6010	lda #\$2a	; pionwaarde (*)
273F-	D0 E2	6015	bne rasnxt	
2741-	A5 7C	6020	lda *totlin	
2743-	C9 10	6025	cmp #\$10	; op deze lijnen
2745-	F0 14	6030	beq rasno	; dienen alleen
2747-	C9 0D	6035	cmp #\$0d	; punten geplaatst
2749-	F0 10	6040	beq rasno	
274B-	C9 0A	6045	cmp #\$0a	
274D-	F0 0C	6050	beq rasno	
274F-	C9 07	6055	cmp #\$07	
2751-	F0 08	6060	beq rasno	
2753-	C9 04	6065	cmp #\$04	
2755-	F0 04	6070	beq rasno	
2757-	C9 01	6075	cmp #\$01	; op andere spatie
2759-	D0 04	6080	bne rasnoa	; + pionwaarde
275B-	A9 2E	6085	lda #\$2e	
275D-	D0 0A	6090	bne rasnob	
275F-	E0 02	6095	coy #\$02	
2761-	D0 04	6100	bne rasnoc	
2763-	A5 7B	6105	lda *piostr	
2765-	D0 02	6110	bne rasnob	
2767-	A9 20	6115	lda #\$20	; char.waarde ()
		6119		
2769-	20 34 13	6120	jsr prcha	
276C-	4C 9F 23	6125	jmp vidsp	; spatie
		6130		
		6135		
		6140	tekst- en databuffer	
		6145		
276F-	00 02 05	6150	coord	.by \$00 \$02 \$05 \$08 \$0b \$0e
2772-	08 0B 0E			
2775-	06 0E 16	6155		.by \$06 \$0e \$16 \$1e \$26
2778-	1E 26			
277A-	21 11 12	6160		.by \$21 \$11 \$12 \$13
277D-	13			
277E-	20 73 63	6165	setxt	.by 'schuiven of springen'
2781-	68 75 69			
2784-	76 65 6E			
2787-	20 6F 66			
278A-	20 73 70			
278D-	72 69 6E			
2790-	67 65 6E			
2793-	0D 0A	6170		.by \$0d \$0a
2795-	20 2D 2D	6175		.by '-----'
2798-	2D 2D 2D			
279B-	2D 2D 2D			
279E-	2D 2D 2D			
27A1-	2D 2D 2D			
27A4-	2D 2D 2D			
27A7-	2D 2D 2D			
27AA-	2D			
27AB-	0D 0A	6180		.by \$0d \$0a
27AD-	6D 65 6E	6185		.by 'men brengt een pion'
27B0-	20 62 72			
27B3-	65 6E 67			
27B6-	74 20 65			
27B9-	65 6E 20			
27BC-	70 69 6F			
27BF-	6E 20			
27C1-	6E 61 61	6190		.by 'naar de enige lege plaats.'

27C4-	72	20	64		
27C7-	65	20	65		
27CA-	6E	69	67		
27CD-	65	20	6C		
27D0-	65	67	65		
27D3-	20	70	6C		
27D6-	61	61	74		
27D9-	73	2C			
27DB-	0D	0A	6D	6195	.by \$0d \$0a 'met keuze uit'
27DE-	65	74	20		
27E1-	6B	65	75		
27E4-	7A	65	20		
27E7-	75	69	74		
27EA-	20	74	77	6200	.by ' twee mogelijkheden.'
27ED-	65	65	20		
27F0-	6D	6F	67		
27F3-	65	6C	69		
27F6-	6A	6B	68		
27F9-	65	64	65		
27FC-	6E	2E			
27FE-	0D	0A	6E	6205	.by \$0d \$0a 'namelijk '
2801-	61	6D	65		
2804-	6C	69	6A		
2807-	6B	20			
2809-	73	63	68	6210	.by 'schuiven: het stuk '
280C-	75	69	76		
280F-	65	6E	3A		
2812-	20	68	65		
2815-	74	20	73		
2818-	74	75	6B		
281B-	20				
281C-	77	6F	72	6215	.by 'wordt over 1 veldstrook'
281F-	64	74	20		
2822-	6F	76	65		
2825-	72	20	31		
2828-	20	76	65		
282B-	6C	64	73		
282E-	74	72	6F		
2831-	6F	6B			
2833-	20	76	65	6220	.by ' vertikaal,' \$0d \$0a
2836-	72	74	69		
2839-	6B	61	61		
283C-	6C	2C	0D		
283F-	0A				
2840-	68	6F	72	6225	.by 'horizontaal '
2843-	69	7A	6F		
2846-	6E	74	61		
2849-	61	6C	20		
284C-	6F	66	20	6230	.by 'of diagonaal verschoven.'
284F-	64	69	61		
2852-	67	6F	6E		
2855-	61	61	6C		
2858-	20	76	65		
285B-	72	73	63		
285E-	68	6F	76		
2861-	65	6E	2E		
2864-	0D	0A		6235	.by \$0d \$0a
2866-	6F	66	77	6240	.by 'ofwel springen,net zoals'
2869-	65	6C	20		
286C-	73	70	72		
286F-	69	6E	67		
2872-	65	6E	2C		
2875-	6E	65	74		
2878-	20	7A	6F		
287B-	61	6C	73		
287E-	20	69	6E	6245	.by ' in het schaakspel.'
2881-	20	68	65		
2884-	74	20	73		
2887-	63	68	61		
288A-	61	6B	73		
288D-	70	65	6C		
2890-	2E				
2891-	0A	0D		6250	.by \$0a \$0d
2893-	6E	6F	6F	6255	.by 'nooit mag men een '
2896-	69	74	20		
2899-	6D	61	67		
289C-	20	6D	65		
289F-	6E	20	65		

28A2- 65 6E 20		
28A5- 73 74 75	6260	.by 'stuk naar achter '
28A8- 6B 20 6E		
28AB- 61 61 72		
28AE- 27 61 63		
28B1- 68 74 65		
28B4- 72 20		
28B6- 76 65 72	6265	.by 'verplaatsen!' \$0d \$0a
28B9- 70 6C 61		
28BC- 61 74 73		
28BF- 65 6E 21		
28C2- 0D 0A		
28C4- 76 65 72	6270	.by 'verliezer is degene '
28C7- 6C 69 65		
28CA- 7A 65 72		
28CD- 20 69 73		
28D0- 20 64 65		
28D3- 67 65 6E		
28D6- 65 20		
28D8- 64 69 65	6275	.by 'die geen geldige zet'
28DB- 20 67 65		
28DE- 65 6E 20		
28E1- 67 65 6C		
28E4- 64 69 67		
28E7- 65 20 7A		
28EA- 65 74		
28EC- 20 6D 65	6280	.by ' meer kan plaatsen!'
28EF- 65 72 20		
28F2- 6B 61 6E		
28F5- 20 70 6C		
28F8- 61 61 74		
28FB- 73 65 6E		
28FE- 21		
28FF- 0D 0A	6285	.by \$0d \$0a
2901- 67 65 65	6290	.by 'geef nu spelniveau in a.u.b.'
2904- 66 20 6E		
2907- 75 20 73		
290A- 70 65 6C		
290D- 6E 69 76		
2910- 65 61 75		
2913- 20 69 6E		
2916- 20 61 2E		
2919- 75 2E 62		
291C- 2E		
291D- 0D 0A	6295	.by \$0d \$0a
291F- 61 3D 20	6300	.by 'a= grootmeester' \$0d \$0a
2922- 67 72 6F		
2925- 6F 74 6D		
2928- 65 65 73		
292B- 74 65 72		
292E- 0D 0A		
2930- 62 3D 20	6305	.by 'b= meester' \$0d \$0a
2933- 6D 65 65		
2936- 73 74 65		
2939- 72 0D 0A		
293C- 63 3D 20	6310	.by 'c= modaal' \$0d \$0a
293F- 6D 6F 64		
2942- 61 61 6C		
2945- 0D 0A		
2947- 64 3D 20	6315	.by 'd= hobby' \$0d \$0a
294A- 68 6F 62		
294D- 62 79 0D		
2950- 0A		
2951- 65 3D 20	6320	.by 'e= beginnening' \$0d \$0a
2954- 62 65 67		
2957- 69 6E 6E		
295A- 65 6C 69		
295D- 6E 67 0D		
2960- 0A		
2961- 67 65 65	6325	.by 'geef niveau a.u.b. :?'
2964- 66 20 6E		
2967- 69 76 65		
296A- 61 75 20		
296D- 61 2E 75		
2970- 2E 62 2E		
2973- 20 3A 3F		
2976- 03	6330	.by \$03
2977- 0D 0A 0A	6335	.by \$0d \$0a \$0a

297A- 77 65 6C	6340	.by 'welk soort tegenstrever'
297D- 6B 20 73		
2980- 6F 6F 72		
2983- 74 20 74		
2986- 65 67 65		
2989- 6E 73 74		
298C- 72 65 76		
298F- 65 72		
2991- 20 77 65	6345	.by 'wenst u ?' \$0d \$0a
2994- 6E 73 74		
2997- 20 75 20		
299A- 3F 0D 0A		
299D- 61 3D 20	6350	.by 'a= aanvallend' \$0d \$0a
29A0- 61 61 6E		
29A3- 76 61 6C		
29A6- 6C 65 6E		
29A9- 64 0D 0A		
29AC- 62 3D 20	6355	.by 'b= verdedigend' \$0d \$0a
29AF- 76 65 72		
29B2- 64 65 64		
29B5- 69 67 65		
29B8- 6E 64 0D		
29BB- 0A		
29BC- 74 6F 65	6360	.by 'toets soort in :?' \$03
29BF- 74 73 20		
29C2- 73 6F 6F		
29C5- 72 74 20		
29C8- 69 6E 20		
29CB- 3A 3F 03		
29CE- 0D 0A 0A	6365	.by \$0d \$0a \$0a
29D1- 6D 6F 65	6370	.by 'moeilijkheidsgraad '
29D4- 69 6C 69		
29D7- 6A 6B 68		
29DA- 65 69 64		
29DD- 73 67 72		
29E0- 61 61 64		
29E3- 20		
29E4- 69 6E 73	6375	.by 'instelbaar in '
29E7- 74 65 6C		
29EA- 62 61 61		
29ED- 72 20 69		
29F0- 6E 20		
29F2- 64 61 6C	6380	.by 'dalende lijn met toetsen'
29F5- 65 6E 64		
29F8- 65 20 6C		
29FB- 69 6A 6E		
29FE- 20 6D 65		
2A01- 74 20 74		
2A04- 6F 65 74		
2A07- 73 65 6E		
2A0A- 0D 0A 30	6385	.by \$0d \$0a '0 t.e.m. 9' \$0d \$0a
2A0D- 20 74 2E		
2A10- 65 2E 6D		
2A13- 2E 20 39		
2A16- 0D 0A		
2A18- 67 65 65	6390	.by 'geef nu moeilijkgraad'
2A1B- 66 20 6E		
2A1E- 75 20 6D		
2A21- 6F 65 69		
2A24- 6C 69 6A		
2A27- 6B 67 72		
2A2A- 61 61 64		
2A2D- 20 69 6E	6395	.by ' in a.u.b. :?' \$03
2A30- 20 61 2E		
2A33- 75 2E 62		
2A36- 2E 20 3A		
2A39- 3F 03		
2A3B- 0D 0A 0A	6400 tetxt	.by \$0d \$0a \$0a
2A3E- 62 69 6A	6405	.by 'bij invoer coördinaten'
2A41- 20 69 6E		
2A44- 76 6F 65		
2A47- 72 20 63		
2A4A- 6F 6F 72		
2A4D- 64 69 6E		
2A50- 61 74 65		
2A53- 6E		
2A54- 20 69 73	6410	.by ' is wijzigen moeilijk'
2A57- 20 77 69		

2A5A-	6A	7A	69		
2A5D-	67	65	6E		
2A60-	20	20	6D		
2A63-	6F	65	69		
2A66-	6C	69	6A		
2A69-	68				
2A6A-	68	65	69	6415	.by 'heidgraad' \$0d \$0a
2A6D-	64	67	72		
2A70-	61	61	64		
2A73-	0D	0A			
				6419	
2A75-	6D	6F	67	6420	.by 'mogelijk!' \$0d \$0a
2A78-	65	6C	69		
2A7B-	6A	6B	21		
2A7E-	0D	0A			
2A80-	74	6F	65	6425	.by 'toets op spatiebalk.'
2A83-	74	73	20		
2A86-	6F	70	20		
2A89-	73	70	61		
2A8C-	74	69	65		
2A8F-	62	61	6C		
2A92-	6B	2E			
2A94-	0D	0A	0A	6430	.by \$0d \$0a \$0a 'druk '
2A97-	64	72	75		
2A9A-	6B	20			
2A9C-	22	6A	22	6435	.by '"j" wanneer u wenst'
2A9F-	20	77	61		
2AA2-	6E	6E	65		
2AA5-	65	72	20		
2AAB-	75	20	77		
2AAB-	65	6E	73		
2AAE-	74				
2AAF-	20	74	65	6440	.by ' te beginnen,dan'
2AB2-	20	62	65		
2AB5-	67	69	6E		
2AB8-	6E	65	6E		
2ABB-	2C	64	61		
2ABE-	6E				
2ABF-	20	73	70	6445	.by ' soeelt u'
2AC2-	65	65	6C		
2AC5-	74	20	75		
2AC8-	20	77	65	6450	.by ' wel van ' \$0d \$0a
2ACB-	6C	20	76		
2ACE-	61	6E	20		
2AD1-	0D	0A			
2AD3-	62	6F	76	6455	.by 'boven naar onder. anders '
2AD6-	65	6E	20		
2AD9-	6E	61	61		
2ADC-	72	20	6F		
2ADF-	6E	64	65		
2AE2-	72	2E	20		
2AE5-	61	6E	64		
2AE8-	65	72	73		
2AEB-	20				
2AEC-	22	6E	22	6460	.by '"n" drukken. : ? ' \$03
2AEF-	20	64	72		
2AF2-	75	6B	6B		
2AF5-	65	6E	2E		
2AF8-	20	20	20		
2AFB-	3A	20	3F		
2AFE-	20	03			
2B00-	20	76	65	6465 txtlost	.by ' verloren! ' \$03
2B03-	72	6C	6F		
2B06-	72	65	6E		
2B09-	21	20	20		
2B0C-	20	03			
2B0E-	20	70	72	6470 txtwin	.by ' proficiat! u hebt'
2B11-	6F	66	69		
2B14-	63	69	61		
2B17-	74	21	20		
2B1A-	20	20	75		
2B1D-	20	68	65		
2B20-	62	74			
2B22-	20	67	65	6475	.by ' gewonnen! ' \$03
2B25-	77	6F	6E		
2B28-	6E	65	6E		
2B2B-	21	20	20		

```

2B2E- 03
2B2F- 20 73 6F 6480 txtong .by ' sorry,zet ongel'
2B32- 72 72 79
2B35- 2C 7A 65
2B38- 74 20 6F
2B3B- 6E 67 65
2B3E- 6C
2B3F- 64 69 67 6485 .by 'dig! ' $03
2B42- 21 20 03
2B45- 73 63 68 6490 fitxt .by 'schuiven of soringen'
2B48- 75 69 76
2B4B- 65 6E 20
2B4E- 6F 66 20
2B51- 73 70 72
2B54- 69 6E 67
2B57- 65 6E
2B59- 0D 0A 6495 .by $0d $0a
2B5B- 2A 2A 2A 6500 .by '*****'
2B5E- 2A 2A 2A
2B61- 2A 2A 2A
2B64- 2A 2A 2A
2B67- 2A 2A 2A
2B6A- 2A 2A 2A
2B6D- 2A 2A
2B6F- 0D 0A 0A 6505 .by $0d $0a $0a
2B72- 63 6F 70 6510 .by 'copyright : de pree'
2B75- 79 72 69
2B78- 67 68 74
2B7B- 20 3A 20
2B7E- 64 65 20
2B81- 70 72 65
2B84- 65
2B85- 74 65 72 6515 .by 'tere r. / belgie'
2B88- 65 20 72
2B8B- 2E 20 2F
2B8E- 20 62 65
2B91- 6C 67 69
2B94- 65
2B95- 0D 0A 0A 6520 .by $0d $0a $0a $0a $0a
2B98- 0A 0A
2B9A- 74 6F 6F 6525 .by 'toon u het speelbord.'
2B9D- 6E 20 75
2BA0- 20 68 65
2BA3- 74 20 73
2BA6- 70 65 65
2BA9- 6C 62 6F
2BAC- 72 64 2E
2BAF- 03 6530 .by $03
2BB0- 68 69 65 6535 piohi .by 'hier' $03
2BB3- 72 03
2BB5- 08 08 08 6540 piocl .by $08 $08 $08 $08 $20
2BB8- 08 20
2BBA- 20 20 20 6545 .by $20 $20 $20 $08 $08 $08
2BBD- 08 08 08
2BC0- 08 03 6550 .by $08 $03
2BC2- 21 20 1C 6555 uppio .by $21 $20 $1c $03
2BC5- 03
2BC6- 2A 20 1C 6560 dwnpio .by $2a $20 $1c $03
2BC9- 03
2BCA- 76 61 6E 6565 fropio .by 'van ' $03
2BCD- 20 03
6570 ;
6575 ; buffer=opslagbuffer van 1 pagina lang
6580 ; van resultaten diepgang tegenzetten
6585 ;
6590 buffer nop
6595 ;
6600 ;
6605 ;
6610 ; einde deel 2.
2BCF- EA

```

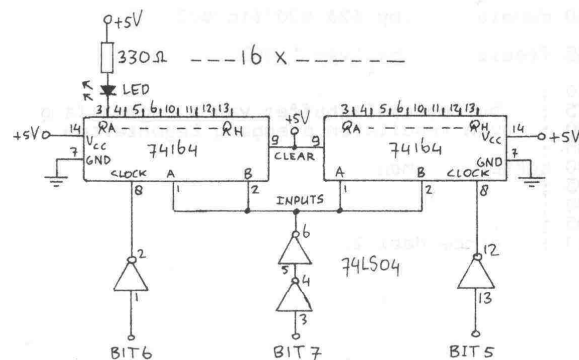
LEDS

PAGE 01

```

0010: *****
0020: LED'S OP JUNIOR
0030: *****
0040:
0050:
0060: DOOR : H. BUURMAN
0070: VAN DER FUYCKSTRAAT 136
0080: 3232 AS BRIELLE.
0090:
0100: 0400 LEMS ORG $0400
0110:
0120: 00 01 TEMPX * $0100
0130: 01 01 DATA * TEMPX +01
0140: 02 01 LEDA * $0102
0150: 03 01 LEDB * LEDA +01
0160:
0170: 82 1A PBD * $1A82
0180: 83 1A PBDD * $1A83
0190:
0200: 0400 8D 02 01 OUTA STA LEDA : REDT A EN X
0210: 0403 8E 00 01 STX TEMPX : A BLIJFT BEWAARD
0220: 0406 A2 A0 LDXIM $A0 : IN LEDA
0230: 0408 8E 82 1A STX PBD : ZET DR EN DDR
0240: 040B 8E 83 1A STX PBDD : BITS 5 EN 7
0250: 040E 20 2A 04 JSR OUT : VERSTUUR A
0260: 0411 AD 02 01 LDA LEDA : EN HERSTEL 'M
0270: 0414 60 RTS :
0280: 0415 8D 03 01 OUTB STA LEDB : REDT A EN X
0290: 0418 8E 00 01 STX TEMPX : A BLIJFT IN LEDB
0300: 041B A2 C0 LDXIM $C0 : ZET DR EN DDR
0310: 041D 8E 82 1A STX PBD : BITS 6 EN 7
0320: 0420 8E 83 1A STX PBDD :
0330: 0423 20 2A 04 JSR OUT : VESTUUR A
0340: 0426 AD 03 01 LDA LEDB : EN HERSTEL
0350: 0429 60 RTS :
0360: 042A 8D 01 01 OUT STA DATA : SLA DATA OP
0370: 042D A2 08 LDXIM $08 : X = 8
0380: 042F AD 01 01 NEXT LDA DATA : REPEAT
0390: 0432 29 80 ANDIM $80 : VERZEND BIT 7
0400: 0434 49 E0 EORIM $E0 :
0410: 0436 8D 82 1A STA PBD :
0420: 0439 0E 01 01 ASL DATA : SCHUIF DATA
0430: 043C 49 60 EORIM $60 :
0440: 043E 8D 82 1A STA PBD : EN GEEF CLOCKPULS
0450: 0441 49 60 EORIM $60 : OP BIT 5 EN 6
0460: 0443 8D 82 1A STA PBD : (SELECTIE DOOR DDR)
0470: 0446 CA DEX : X = X-1
0480: 0447 D0 E6 BNE NEXT : UNTIL X = 0
0490: 0449 8E 83 1A STX PBDD : ALLE BITS INGANG
0500: 044C AE 00 01 LDX TEMPX : HERSTEL X
0510: 044F 60 RTS :

```



Aan de tekentafel:
Fridus Jonkman

```
*****
**          PRINT ROUTINES VOOR EEN  STAR DP 510  PRINTER          **
** (of iedere andere printer met een Centronics interface)      **
** Auteur : S.T. Woldringh                                       **
**          Klievering 619                                         **
**          1104 KC Amsterdam ZO                                   **
**          *****                                              **
*****
```

De besturing wordt gedaan met twee PIA's, waarbij een PIA (PIAA) de data "opstuurt" en de andere de besturing ter hand neemt.
In het voorbeeld is PIAB-3 verbonden aan de STROBE (pin 1)
PIAB-4 verbonden aan de ACK (pin 10)
PIAB-5 verbonden aan de BUSY (pin 11)
De PAPER-END en SELECTED en ERROR zijn niet gebruikt (pin 12, 13 en 32)

Er zijn vier routines:

- PINIT : Initialiseren van de PIA's
- Initialiseren van de printer
- PFFFEED : Opsturen van een Form-Feed
- PCRLF : Opsturen van een CR-LF
- POUT : Opsturen van een character (dat zich in de 'accu' bevindt)
- PSTOP : Stoppen van de printer
- PIA's resetten

Opm. : Om de PIA te beschermen zijn in alle uitgaande PIA-bits inverters (74LS...) opgenomen; in de coding is hiervan uitgegaan.

***** STAR DP510 printerroutine *****

```
PINIT  LDAIM $00          SET PIAA AS OUTPUT
        STA  PIAA
        LDAIM $FF
        STA  PIAA+01
        LDA  PIAB          SET PIAB AS INPUT (4.5)
        ORAIM $08          AND AS OUTPUT (3)
        STA  PIAB
        LDA  PIAB+01
        ANDIM $CF          (11001111)
        ORAIM $08          (00001000)
        STA  PIAB+01
        LDAIM $1B          Initialize printer by an
        JSR  POUT          Esc ("adestaartie")
        LDAIM $40
        JSR  POUT
        RTS              READY

PFFFEED LDAIM $0C          GIVE A FORM-FEED
        JSR  POUT
        RTS

PCRLF   LDAIM $0D          GIVE A CR
        JSR  POUT
        LDAIM $0A          AND A LF
        JSR  POUT
        RTS

POUT     EORIM $FF          FLIP THE BITS
        STA  PIAA          PUT IT ON THE LINE
POUT1    LDA  PIAB          CHECK FOR BUSY AND ACKNOWLEDGE
```

```

ANDIM $30
CMPIM $10
BNE POUT1      HE'S NOT READY
LDA PIAB       GIVE STROBE SIGNAL
ANDIM $F7
STA PIAB
NOP
LDA PIAB       WAIT 1 mSec
ORAIM $08      STOP STROBE SIGNAL
STA PIAB
RTS           READY

PSTOP LDAIM $0D      GIVE ONE LAST CR
      JSR POUT       TO CLEAR PRINTER-BUFFER
      LDA PIAB+01    SET PIAB BACK AS INPUT
      ANDIM $F7      (11110111)
      STA PIAB+01
      LDAIM $00      AND PIAA TOO
      STA PIAA+01
      RTS           READY

```

Noot van de redactie:

Wij kunnen ons voorstellen dat, indien men voor het eerst een printer aan de computer wil hangen, er eerst aan de redactie gevraagd wordt of voor die specifieke printer een routine bestaat. Een enkel lid heeft tot nu toe een dergelijke routine ingeleverd, waarmee wij in de gelegenheid zijn aan de behoefte tegemoet te komen. Er zijn natuurlijk veel meer toepassingen met andere printers dan de OKI 80, STAR DPS10 (er is in de copybuffer een routine waarmee vanuit Micro-Ade de Tandy printer VII over een VIA wordt aangestuurd; deze wacht op een mogelijkheid tot publikatie).

Wij roepen eenieder op printer-routines voor andere dan de twee eerstgenoemde in te zenden aan de redactie. Op die manier kunnen we andere leden van dienst zijn. Bovendien komen ook deze inzendingen in de eerder genoemde copybuffer terecht.

Wellicht is het goed, voor de nieuwe leden vooral, nog even te verwijzen naar de volgende publikaties:

H.J.C. Otten	RS232 Printer Output Routine.	De KIM Kenner 13
H.J.C. Otten en	Printer Output H14 met hand-	De 6502 Kenner 19
P.G.J. de Beer	shake: uitgebreide versie van RS232 Printer Output Routine.	
Fr. Smeehuysen	Printe	De 6502 Kenner 23
	- een routine om de printer softwarematig aan- en uit te schakelen	
	- een routine om de printer te kunnen sturen	
	- een routine om de print-control te verzorgen	
W.L. van Pelt,	Printen op OKI Microline 80	De 6502 Kenner 26
J.F. van Sprang,	Printer: een nieuwe routine voor de JUNIOR-computer	
K. van Nieuwenhove en	(adres \$1334 verplaatst naar adres \$1634)	
R. Langeveld	Expanding Micro-Ade with printer on/off command	De 6502 Kenner 28

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: 9102
0440:
0450:
0460:
0470:
0480:
0490:
0500:
0510:
0520:
0530:
0540:
0550: 9102 20 7E 91
0560: 9105 A9 00
0570: 9107 20 64 C3
0580: 910A 4C 8A C2
0590:
0600: 913D
0610:
0620:
0630:
0640:
0650:
0660:
0670:
0680:
0690: 913D A5 EE
0700: 913F C9 DD
0710: 9141 F0 08
0720: 9143 C9 99
0730: 9145 F0 08
0740: 9147 20 5F C3
0750: 914A 60
0760: 914B 20 AF 91
0770: 914E 20 61 C3
0780: 9151 60
0790: 9152 4C 48 99
0800:

UITBREIDINGEN OP DE SYM-1 BASIC VOOR KIM-1 EN JUNIOR

AUTOMATISCHE REGELNUMMERING

AUTEUR: F. J. M. SNEEHUIJZEN
LIPPEDAL 19
2904 CL CAPELLE AAN DEN IJSSEL
TEL: 010-512507

HET PROGRAMMA "AUTOMATISCHE REGELNUMMERING" HEEFT TOT DOEL OM BIJ HET INVOEREN VAN BASIC PROGRAMMAREGELS VOORAF AUTOMATISCH EEN REGELNUMMER TE GENEREREN ZODAT DIT NIET STEEDS BEHOEFT TE WORDEN INGETOETST. NA HET INTOETSEN VAN HET KOMMANDO "#AUTO" (ZIE BESCHRIJVING VAN HET TOEVOEGEN VAN NIEUWE BASIC KOMMANDO'S) VERSCHIJNT OP HET SCHERM DE VRAAG "REGNR ?".

HIEROP KAN WORDEN GEANTWOORD MET EEN RETURN OF EEN GETAL VAN VIER CIJFERS INCLUSIEF EVENTUELE VOORLOOPNULLEN. WANNEER NIJ EEN RETURN WORDT INGETOETS WORDT AUTOMATISCH GESTART MET REGELNUMMER 10 (VOORLOOPNULLEN WORDEN ONDERDRUKT!). WANNEER EEN GETAL VAN VIER CIJFERS IS INGETOETS ZAL MET DIT GETAL ALS EERSTE REGELNUMMER WORDEN BEGONNEN (VOORLOOPNULLEN WORDEN OOK HIER ONDERDRUKT!).

IN BEIDE GEVALLEN WORDT VERVOLGENS HET REGELNUMMER STEEDS MET 10 OPGEHOOGD.

OM TE BEPALEN OF ER AUTOMATISCH EEN REGELNUMMER MOET WORDEN AFGEDRUKT WORDT ER GEBRUIK GEMAAKT VAN EEN VLAGREGISTER OP ADRES \$00EE. OM DIT REGISTER TE TESTEN DIENT DE VOLGENDE WIJZIGING IN BASIC TE WORDEN AANGEBRACHT:

OP ADRES \$C287 STAAT - JSR \$C35F -
DIT DIENT GEWIJZIGD TE WORDEN IN - JSR \$913D -

HIERNA VOLGEN EEN DRIETAL PROGRAMMA'S T.W.:

- AUTO - HIERMEE WORDT DE AUTO-NUMBER VLAG AANGEZET
- FLAG - HIERMEE WORDT DE AUTO-NUMBER VLAG GETEST
- BASREG - DIT IS HET EIGENLIJKE AUTO-NUMBER PROGRAMMA

AUTO ORG \$9102

ROUTINE OM NA HET INTOETSEN VAN EEN #AUTO KOMMANDO EEN KOPPELING TE LEGGEN TUSSEN BASIC EN DE AUTOMATISCHE REGELNUMMER-ROUTINE.

JSR BASREG ; SPRING NAAR AUTO-NUMBER ROUTINE
LDAM \$00
JSR \$C364 ; BASIC INPUT ROUTINE
JMP \$C28A ; VERWERKEN VAN DE INGEVOERDE REGEL

FLAG ORG \$913D

ROUTINE OM TE TESTEN OF DE AUTO-NUMBER VLAG EN/OF DE DELETE-VLAG IS GEZET.

LDAX \$EE ; HAAL HET AUTO-NUMBER VLAG-
CMPIM \$DD ; REGISTER OP
BEQ AUTONU ; IS DE VLAG GEZET ?
CMPIM \$99 ; IS DE DELETE-VLAG GEZET ?
BEQ DEL
JSR \$C35F ; TERUG NAAR BASIC INPUT-ROUTINE
RTS
AUTONU JSR \$91AF ; NAAR AUTO-NUMBER ROUTINE
JSR \$C361 ; EN TERUG NAAR BASIC INPUT-ROUTINE
RTS
DEL JMP \$9948 ; NAAR REGEL-DELETE ROUTINE

DE 6502 KENNER

```

0810: 917E      BASREG ORG   $917E
0820:
0830:      *****
0840:
0850:      DEZE ROUTINE VERZORGT VOOR BASIC DE AUTOMATISCHE REGEL-NUMMERING
0860:      DOOR STEEDS 10 BIJ HET VOORGAANDE REGELNUMMER OP TE TELLLEN, TE
0870:      BEGINNEN BIJ HET INGETOETSTE REGELNUMMER.
0880:      ALS REGELNUMMER DIENEN VIER CIJFERS TE WORDEN INGETOETST, WAARBIJ
0890:      ALS LAATSTE CIJFER ALTIJD EEN 0 ZAL WORDEN AFGEDRUKT ONDANKS HET
0900:      FEIT DAT ER EEN ANDER CIJFER IS INGETOETST.
0910:      INDIEN DIREKT NA DE MELDING "REG-NR?" EEN RETURN WORDT INGETOETST
0920:      WORDT AUTOMATISCH MET REGELNUMMER 10 BEGONNEN.
0930:      BIJ HET PRINTEN VAN DE REGELNUMMERS WORDEN DE VOORLOOPNULLEN
0940:      ONDERDRUKT.
0950:      OM DE AUTOMATISCHE REGELNUMMERING WEER UIT TE SCHAKELN DIEN
0960:      CONTROL-B INGETOETST TE WORDEN.
0970:
0980:      *****
0990:
1000:
1010: 917E      OUTCH *      $1EA0
1020: 917E      GETKEY *     $1ESA
1030:
1040: 917E A9 DD      LDAIM $DD      ; ZET DE AUTO-NUMBER
1050: 9180 85 EE      STAZ $EE      ; VLAG AAN
1060: 9182 A9 30      LDAIM $30      ; ZET HET REGELNUMMER-
1070: 9184 85 E5      STAZ $E5      ; REGISTER OP DE
1080: 9186 85 E6      STAZ $E6      ; BEGINWAARDE 0010
1090: 9188 A9 31      LDAIM $31
1100: 918A 85 E7      STAZ $E7
1110: 918C A2 07      LDXIM $07      ; PRINTEN "REG-NR ? "
1120: 918E BD FC 91  LOOP      LDAAX TEKST
1130: 9191 20 A0 1E    JSR      OUTCH
1140: 9194 CA          DEX
1150: 9195 10 F7      BPL      LOOP
1160: 9197 A2 01      LDXIM $01      ; INVOEREN VAN EEN
1170: 9199 CA          DEX          ; REGELNUMMER VIA
1180: 919A 20 5A 1E  DEX KEY     JSR      GETKEY      ; HET TOETSENBORD
1190: 919D C9 08      CMPIM $08      ; IS HET EEN BACK-SPACE ?
1200: 919F F0 F8      BEQ      DEX      ; ZO JA, TERUG IN INPUTBUFFER
1210: 91A1 C9 0D      CMPIM $0D      ; IS HET EEN RETURN ?
1220: 91A3 F0 07      BEQ      IN      ; ZO JA, INGETOETSTE REGELNUMMER AFDrukKEN
1230: 91A5 95 E5      STAZ $E5      ; INGETOETSTE REGELNUMMER
1240: 91A7 E8          INX          ; IN REGELNUMMER-REGISTER
1250: 91A8 E0 05      CPXIM $05      ; PLAATSEN (MAX 4 POSITIES)
1260: 91AA D0 EE      BNE      KEY
1270: 91AC 20 04 92  IN          JSR      CRLF
1280: 91AF A2 00      LDXIM $00
1290: 91B1 EA          NOP          ; HIER STOND EEN EXTRA CRLF
1300: 91B2 EA          NOP
1310: 91B3 EA          NOP
1320: 91B4 A5 E5      LDZ $E5      ; DEZE SUBROUTINE ONDERDRUKT
1330: 91B6 C9 30      CMPIM $30      ; DE NUL OP DE PLAATS VAN
1340: 91B8 F0 08      BEQ      100T   ; DE DUIZENDTALLEN
1350: 91BA 20 F5 91    JSR      HULP   ; PRINTEN EN IN BASIC INPUTBUFFER PLAATSEN
1360: 91BD A5 E6      LDZ $E6      ; HAAL HONDERDTAL OP
1370: 91BF 4C C8 91    JMP      100TAL  ; DRUK HONDERDTAL AF
1380: 91C2 A5 E6      LDZ $E6      ; DEZE SUBROUTINE ONDERDRUKT
1390: 91C4 C9 30      CMPIM $30      ; DE NUL OP DE PLAATS VAN
1400: 91C6 F0 03      BEQ      10T    ; DE HONDERDTALLEN
1410: 91C8 20 F5 91  100TAL     JSR      HULP   ; PRINTEN EN IN BASIC INPUTBUFFER PLAATSEN
1420: 91CB A5 E7      LDZ $E7      ; HAAL DE TIJNTALLEN OP
1430: 91CD 20 F5 91  10T        JSR      HULP   ; PRINTEN EN IN BASIC INPUTBUFFER PLAATSEN
1440: 91D0 A9 30      LDAIM $30      ; DE EENHEDEN ZIJN ALTIJD NUL
1450: 91D2 20 F5 91  10T        JSR      HULP   ; PRINTEN EN IN BASIC INPUTBUFFER PLAATSEN
1460: 91D5 A9 20      LDAIM $20      ; EEN SPATIE ACHTER HET REGELNUMMER
1470: 91D7 20 F5 91  10T        JSR      HULP   ; PRINTEN EN IN BASIC INPUTBUFFER PLAATSEN
1480: 91DA E6 E7      INCZ $E7      ; VERHOOG DE TIJNTALLEN MET 1
1490: 91DC A5 E7      LDZ $E7      ; EN KONTROLEER OF HET HOGER
1500: 91DE C9 3A      CMPIM $3A      ; IS DAN 90
1510: 91E0 D0 12      BNE      RETURN ; ZO JA, MAAK ER DAN
1520: 91E2 A9 30      LDAIM $30      ; WEER 0 VAN EN
1530: 91E4 85 E7      STAZ $E7      ; VERHOOG DE HONDER-
1540: 91E6 E6 E6      INCZ $E6      ; TALLEN MET 1
1550: 91E8 A5 E6      LDZ $E6      ; EN KONTROLEER OF
1560: 91EA C9 3A      CMPIM $3A      ; HET HOGER IS DAN 900
1570: 91EC D0 06      BNE      RETURN ; ZO JA, MAAK ER DAN
1580: 91EE A9 30      LDAIM $30      ; WEER 0 VAN EN
1590: 91F0 85 E6      STAZ $E6      ; VERHOOG DE DUIZEND-
1600: 91F2 E6 E5      INCZ $E5      ; TALLEN MET 1
1610: 91F4 60          RETURN RTS

```

```

1620: 91F5 95 1E HULP STAZ $1E ; VULLEN VAN HET BASIC INPUTBUFFER
1630: 91F7 20 A0 1E JSR OUTCH ; PRINTEN VAN HET REGELNUMMER
1640: 91FA E8 INX
1650: 91FB 60 RTS
1660: 91FC 20
1670: 91FD 3F TEKST = '
1680: 91FE 52 = 'R
1690: 91FF 4E = 'N
1700: 9200 20 = '
1710: 9201 47 = 'G
1720: 9202 45 = 'E
1730: 9203 52 = 'R
1740: 9204 A9 0A CRLF LDAM $0A
1750: 9206 20 A0 1E JSR OUTCH
1760: 9209 60 RTS

```

```

0010: 9000 BASKEY ORG $9000
0020:
0030: *****
0040:
0050: OM DE AUTO-NUMBER VLAG WEER AF TE ZETTEN
0060: DIENT BASIC TE REAGEREN OP DE DAARVOOR
0070: BESTEMDE TYPE-IN. GEKOZEN IS HIER VOOR DE
0080: RUB-OUT OF DELETE TOETS.
0090: HIERVOOR MOET OP ADRES $C38F DE JSR $1E5A
0100: (KIM GETKEY ROUTINE) GEWIJZIGD WORDEN
0110: IN EEN JSR $9000.
0120:
0130: *****
0140:
0150: 9000 OUTCH * $1EA0
0160: 9000 GETKEY * $1E5A
0170: 9000 WARM * $C27E
0180:
0190: 9000 20 5A 1E JSR GETKEY : SPRING NAAR KIM GETKEY-ROUTINE
0200: 9003 C9 7F CMPIM $7F : IS HET EEN RUB-OUT ?
0210: 9005 D0 03 BNE BACK
0220: 9007 20 6F 94 JSR RESET : SPRING NAAR DE BASIC RESET-ROUTINE
0230: 900A 60 BACK RTS
0240:
0250: 946F RESET ORG $946F
0260:
0270: *****
0280:
0290: ROUTINE OM DE AUTO-NUMBER VLAG AF TE ZETTEN.
0300:
0310: *****
0320:
0330: 946F A9 CC LDAM $CC ; ZET DE AUTO-NUMBER VLAG AF
0340: 9471 85 EE STAZ $EE ; EN GA TERUG
0350: 9473 60 RTS ; NAAR BASIC
0360:

```

MISSING DATA IN JUNIOR-BOOK.

In JUNIOR - book 4, p.203 from adress \$0B3E on line 0671 the source-listing of the TM-moni- tor is not complete. Our member Marcel van Dorp informed us. Here are the missing data.	0B66 85 FA STAZ POINTL 0B68 20 F3 0B JSR RDBYT	0B98 D0 01 BNE RDTAP
	0B6B 20 4B 0C JSR CHKSUM 0B6E 85 FB STAZ POINTH	0B9A 60 RTS 0B9B 4C 02 0B RDTAP JMP RDTAPE
	0B70 20 F3 0B JSR RDBYT 0B73 30 8D BMI RDTAPE 0B75 F0 13 BEQ CHK 0B77 20 4B 0C JSR CHKSUM	0B9E AD 79 1A LDA ID 0BA1 C9 00 CMPIM \$00 0BA3 F0 BB BEQ CHKR 0BA5 C9 FF CMPIM \$FF
0B43 D0 F1 BNE TENSYN	0B7A A0 00 LDYIM \$00 0B7C 91 FA STAIY POINTL 0B7E E6 FA INCZ POINTL	0BA7 D0 F2 BNE RDTAP 0BA9 20 F3 0B JSR RDBYT 0BAC 20 4B 0C JSR CHKSUM
0B45 20 36 0C TENSJSR RDCH 0B48 20 5D 0C JSR CHARVU 0B4B C9 2A CMPIM \$2A 0B4D F0 06 BEQ TENS 0B4F C9 16 CMPIM \$16 0B51 F0 F2 BEQ TENS 0B53 D0 AD BNE RDTAPE	0B80 D0 02 BNE CHKS 0B82 E6 FB INCZ POINTH	0BB2 20 4B 0C JSR CHKSUM 0BB5 AD 70 1A LDA SAL 0BB8 85 FA STAZ POINTL
0B55 20 5D 0C TENSJSR CHARVU 0B58 20 F3 0B JSR RDBYT 0B5B CD 79 1A CMP ID 0B5E D0 3E BNE RDTAP	0B84 20 64 0C CHKS JSR VU 0B87 4C 70 0B JMP FILMEM	
0B60 20 F3 0B CHKRJSR RDBYT 0B63 20 4B 0C JSR CHKSUM	0B8A 20 F3 0B CHK JSR RDBYT 0B8D CD 6E 1A CMP CHKL 0B90 D0 09 BNE RDTAP 0B92 20 F3 0B JSR RDBYT 0B95 CD 6F 1A CMP CHKH	

Did you ever found slips in such
books, in Elektor's Paperware or
elsewhere, don't forget to inform
Mr. W.L. van Pelt, editorial ma-
nager, c/Jacob Jordaensstraat 15,
2923 CK, Krimpen a.d. IJssel, The
Netherlands.

```

1 REM
10 PRINT : GOSUB 1200
20 PRINT"WISKUNDE PROGRAMMA":SPC(25);"V. 1.0"          ":PRINT
30 PRINT"DE MOGELIJKHEDEN VAN DIT PROGRAMMA ZIJN ALS VOLGT : "
40 PRINT"1) DETERMINANTEN BEREKENEN"
50 PRINT"2) KWADRATISCHE VERGELIJKINGEN OPlossen"
60 PRINT"3) ..... "
70 PRINT"4) ..... "
80 PRINT"5) ..... "
90 PRINT"6) ..... "
100 PRINT"7) N BOVEN K UITREKENEN"
110 PRINT"8) BINOMIALE KANSEN BEREKENEN"
120 PRINT"9) CUMULATIEVE BINOMIALE KANSEN BEREKENEN"
130 PRINT"10) ..... "
140 INPUT"TYP UW KEUZE ":Y
150 IF NOT( Y<1 OR Y>10 OR Y<>INT(Y) ) THEN170
160 GOTO 10
170 ON Y GOTO190,420,540,550,560,570,580,730,870,1050
180 GOSUB 1180:GOTO 10
190 GOSUB1200
200 PRINT"DETERMINANTEN BEREKENEN"
210 PRINT"2- OF 3-DIMENSIONAAL":
220 INPUT Y
230 IF Y<3 OR Y<2 OR INT(Y)<>Y THEN210
240 ON Y GOTO.250,330
250 PRINT"GEEF GETALLEN UIT EERSTE KOLOM ( A.B ) "
260 INPUTA1,A2
270 PRINT"GEEF GETALLEN UIT TWEEDE KOLOM ( C.D ) "
280 INPUTA3,A4
290 PRINT"DE UITKOMST IS":A1*A4-A2*A3
300 GOSUB 1060
310 IF Y THEN 190
320 GOTO 10
330 PRINT"GEEF GETALLEN UIT EERSTE KOLOM ( A.B.C ) "
340 INPUTA1,A2,A3
350 PRINT"GEEF GETALLEN UIT TWEEDE KOLOM ( D.E.F ) "
360 INPUTA4,A5,A6
370 PRINT"GEEF GETALLEN UIT DERDE KOLOM ( G.H.I ) "
380 INPUTA7,A8,A9
390 PRINT"DE UITKOMST IS":
400 PRINTA1*(A5*A9-A6*A8)+A2*(A6*A7-A4*A9)+A3*(A4*A8-A5*A7)
410 GOTO 300
420 GOSUB 1200:PRINT"KWADRATISCHE VERGELIJKINGEN OPlossen"
430 PRINT: PRINT"DE ALGEMENE VORM IS : A*X^2 + B*X + C = 0"
440 INPUT "GEEF A ":A
450 INPUT "GEEF B ":B
460 INPUT "GEEF C ":C
470 IF NOT(B^2-4*A*C<0 OR A=0) THEN 490
480 PRINT"ER ZIJN GEEN OPLOSSINGEN": GOTO 510
490 PRINT"X = ":(-1*B-SQR(B^2-4*A*C))/(2*A):
500 PRINT"OF X = ":(-1*B+SQR(B^2-4*A*C))/(2*A)
510 GOSUB 1060
520 IF Y THEN420
530 GOTO 10
540 GOTO180
550 GOTO180
560 GOTO180
570 GOTO180
580 GOSUB 1200: PRINT"N BOVEN K UITREKENEN"
590 PRINT : INPUT"GEEF N ":N
600 INPUT"GEEF K ":K
610 GOSUB 630
620 GOTO 690
630 U=1
640 IF K=0 THEN 680
650 FOR I=1TOK
660 U=U*((N-I+1)/(K-I+1))
670 NEXT I
680 RETURN
690 PRINT"DE UITKOMST IS : ":U
700 GOSUB 1060
710 IF Y THEN 580
720 GOTO 10

```

```

730 PRINT : GOSUB 1200
740 PRINT"BINOMIALE KANSSEN BEREKENEN"
750 PRINT:PRINT"BEREKEND WORDT DE KANS OP K SUCCESSEN ."
760 PRINT"BIJ EEN EXPERIMENT VAN N BERNOUILLI-EXPERIMENTEN ."
770 PRINT"MET EEN AFZONDERLIJKE KANS VAN P ."
780 PRINT : INPUT"GEEF N ":N
790 INPUT"GEEF K ":K
800 INPUT"GEEF P ":P
810 GOSUB 630
820 PRINT"DE KANS IS : ": (U*P^K)*(1-P)^(N-K)
830 PRINT"1 MIN DE KANS IS : ":1-(U*P^K)*(1-P)^(N-K)
840 GOSUB 1060
850 IF Y THEN 730
860 GOTO 10
870 PRINT:GOSUB 1200
880 PRINT"CUMULATIEVE BINOMIALE KANSSEN BEREKENEN":PRINT
890 PRINT"BEREKEND WORDT DE KANS OP X OF MINDER SUCCESSEN ."
900 PRINT"BIJ EEN EXPERIMENT VAN N BERNOUILLI-EXPERIMENTEN ."
910 PRINT"MET EEN AFZONDERLIJKE KANS VAN P"
920 INPUT"GEEF N ":N
930 INPUT"GEEF X ":X
940 INPUT"GEEF P ":P
950 U1=0
960 FOR K=0 TO X
970 GOSUB 630
980 U1=U1+(U*P^K)*(1-P)^(N-K)
990 NEXT K
1000 PRINT"DE KANS IS : ":U1
1010 PRINT"1 MIN DE KANS IS : ":1-U1
1020 GOSUB 1060
1030 IF Y THEN 870
1040 GOTO 10
1050 GOTO 180
1060 RESTORE
1070 FOR I=0 TO 8
1080 READ A
1090 POKE 512+I,A
1100 NEXT I
1110 DATA &"0020", &"00AE", &"0012", &"00AB", &"00A9", &"0000"
1120 DATA &"004C", &"004C", &"00D1"
1130 PRINT"NOG EENS ? ( J/N ) ":
1140 A=USR(512,0)
1150 Y=0
1160 IF A=74 THEN Y=-1
1170 RETURN
1180 PRINT"NOG NIET WERKEND":FOR W=1 TO 1500 :NEXT W
1190 RETURN
1200 PRINT CHR$(12):FOR W=1 TO 100 :NEXT W
1210 RETURN

```

Informatiepakket Disk Operating System van Koen van Nieuwenhove, Leuven België, voor JUNIOR met 8" of 5 1/4"

In editie 27 van augustus 1983 publiceerde Koen van Nieuwenhove het artikel "Koppeling van JUNIOR aan 8" floppy". Het operating system, dat ook geschikt is voor 5" floppy, en evenals de systemen van OHIO en Proton in single density werkt, is ontleend aan artikelen uit BYTE en het boek "A floppy Disk Operating System for the 8080" van Kenneth B. Welles.

Het informatiepakket bevat een groot aantal onderdelen, w.o.: schema's, de twee artikelen uit BYTE, data sheets van de FD 1771-01 floppy disk formatter en controller, aanpassingen aan de PRCHA-routine, User Guide FDOS 1.2, Basis-software, aanpassingen aan KB-9 Basic voor JUNIOR, Demo-programma's voor Basic, aanpassingen aan assembler van Moser, Hypersort voor RAE, aanpassingen aan Pascal, en nog een aantal zaken.

Het informatiepakket is te verkrijgen door overschrijven van Fl. 97,50 op postrekening 841433 t.n.v. W.L. van Pelt te Kriepen a.d. IJssel. Een Eurocheque, mits volledig ingevuld, mag ook. Of storten op postgiro 3050 van de AMRO-bank te Kriepen a.d. IJssel, ten gunste van rekening 44.11.06.471 op naam van W.L. van Pelt.

Men dient aan de hand van de gegevens zelf een controlerkaart te maken. De genoemde controller 1771 is verkrijgbaar bij DIL Elektronica te Rotterdam.

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 van de band willen halen en er op willen schrijven.

Stel: U heeft een Basic-programma waarmee U een adressen bestand kunt samenstellen. Op de Basic-cassette nr. 2 staat zo'n programma: CLAIRE. Als U het programma laadt, is de lijst weer leed en moet alles weer opnieuw ingevoerd. Met behulp van Save and Load kunt U de ingevoerde adressen op de band bewaren.

Het programma is voorzien van een demonstratieprogramma en is voor Fl. 5,- verkrijgbaar.

```

00001: *****
00002: *** LICHTKRANT BEELDSCHEM LINTOERN ***
00003: *** SCHERT LICHTKRANT BEELDSCHEM L ***
00004: *** BEELDSCHEM LINTOERN BEELDSCHEM ***
00005: *****
00006:
00007: DOOR : TUD UNN RIJSEL
00008:         LEVENLICH 10
00009:         5020 40 EINDHOVEN
00010:
00011:
00012:
00013:
00014:
00015:
00016:
00017:
00018:
00019:
00020:
00021:
00022:
00023:
00024:
00025:
00026:
00027:
00028:
00029:
00030:
00031:
00032:
00033:
00034:
00035:
00036:
00037:
00038:
00039:
00040:
00041:
00042:
00043:
00044:
00045:
00046:
00047:
00048:
00049:
00050:
00051:
00052:
00053:
00054:
00055:
00056:
00057:
00058:
00059:
00060:
00061:
00062:
00063:
00064:
00065:
00066:
00067:
00068:
00069:
00070:
00071:
00072:
00073:
00074:
00075:
00076:
00077:
00078:
00079:
00080:
00081:
00082:
00083:
00084:
00085:
00086:
00087:
00088:
00089:
00090:
00091:
00092:
00093:
00094:
00095:
00096:
00097:
00098:
00099:
00100:
00101:
00102:
00103:
00104:
00105:
00106:
00107:
00108:
00109:
00110:
00111:
00112:
00113:
00114:
00115:
00116:
00117:
00118:
00119:
00120:
00121:
00122:
00123:
00124:
00125:
00126:
00127:
00128:
00129:
00130:
00131:
00132:
00133:
00134:
00135:
00136:
00137:
00138:
00139:
00140:
00141:
00142:
00143:
00144:
00145:
00146:
00147:
00148:
00149:
00150:
00151:
00152:
00153:
00154:
00155:
00156:
00157:
00158:
00159:
00160:
00161:
00162:
00163:
00164:
00165:
00166:
00167:
00168:
00169:
00170:
00171:
00172:
00173:
00174:
00175:
00176:
00177:
00178:
00179:
00180:
00181:
00182:
00183:
00184:
00185:
00186:
00187:
00188:
00189:
00190:
00191:
00192:
00193:
00194:
00195:
00196:
00197:
00198:
00199:
00200:
00201:
00202:
00203:
00204:
00205:
00206:
00207:
00208:
00209:
00210:
00211:
00212:
00213:
00214:
00215:
00216:
00217:
00218:
00219:
00220:
00221:
00222:
00223:
00224:
00225:
00226:
00227:
00228:
00229:
00230:
00231:
00232:
00233:
00234:
00235:
00236:
00237:
00238:
00239:
00240:
00241:
00242:
00243:
00244:
00245:
00246:
00247:
00248:
00249:
00250:
00251:
00252:
00253:
00254:
00255:
00256:
00257:
00258:
00259:
00260:
00261:
00262:
00263:
00264:
00265:
00266:
00267:
00268:
00269:
00270:
00271:
00272:
00273:
00274:
00275:
00276:
00277:
00278:
00279:
00280:
00281:
00282:
00283:
00284:
00285:
00286:
00287:
00288:
00289:
00290:
00291:
00292:
00293:
00294:
00295:
00296:
00297:
00298:
00299:
00300:
00301:
00302:
00303:
00304:
00305:
00306:
00307:
00308:
00309:
00310:
00311:
00312:
00313:
00314:
00315:
00316:
00317:
00318:
00319:
00320:
00321:
00322:
00323:
00324:
00325:
00326:
00327:
00328:
00329:
00330:
00331:
00332:
00333:
00334:
00335:
00336:
00337:
00338:
00339:
00340:
00341:
00342:
00343:
00344:
00345:
00346:
00347:
00348:
00349:
00350:
00351:
00352:
00353:
00354:
00355:
00356:
00357:
00358:
00359:
00360:
00361:
00362:
00363:
00364:
00365:
00366:
00367:
00368:
00369:
00370:
00371:
00372:
00373:
00374:
00375:
00376:
00377:
00378:
00379:
00380:
00381:
00382:
00383:
00384:
00385:
00386:
00387:
00388:
00389:
00390:
00391:
00392:
00393:
00394:
00395:
00396:
00397:
00398:
00399:
00400:
00401:
00402:
00403:
00404:
00405:
00406:
00407:
00408:
00409:
00410:
00411:
00412:
00413:
00414:
00415:
00416:
00417:
00418:
00419:
00420:
00421:
00422:
00423:
00424:
00425:
00426:
00427:
00428:
00429:
00430:
00431:
00432:
00433:
00434:
00435:
00436:
00437:
00438:
00439:
00440:
00441:
00442:
00443:
00444:
00445:
00446:
00447:
00448:
00449:
00450:
00451:
00452:
00453:
00454:
00455:
00456:
00457:
00458:
00459:
00460:
00461:
00462:
00463:
00464:
00465:
00466:
00467:
00468:
00469:
00470:
00471:
00472:
00473:
00474:
00475:
00476:
00477:
00478:
00479:
00480:
00481:
00482:
00483:
00484:
00485:
00486:
00487:
00488:
00489:
00490:
00491:
00492:
00493:
00494:
00495:
00496:
00497:
00498:
00499:
00500:
00501:
00502:
00503:
00504:
00505:
00506:
00507:
00508:
00509:
00510:
00511:
00512:
00513:
00514:
00515:
00516:
00517:
00518:
00519:
00520:
00521:
00522:
00523:
00524:
00525:
00526:
00527:
00528:
00529:
00530:
00531:
00532:
00533:
00534:
00535:
00536:
00537:
00538:
00539:
00540:
00541:
00542:
00543:
00544:
00545:
00546:
00547:
00548:
00549:
00550:
00551:
00552:
00553:
00554:
00555:
00556:
00557:
00558:
00559:
00560:
00561:
00562:
00563:
00564:
00565:
00566:
00567:
00568:
00569:
00570:
00571:
00572:
00573:
0057
```


44

HEXADUMP:
 0 1 2 3 4 5 6 7 8 9 a b c d e f
 0350: 0a 0a 47 45 45 46 20 45 45 4a 20 4b 41 52 41 4b
 0360: 54 45 52 20 4f 5b 3a 20 03 0d 0a 0a 48 4f 52 49
 0370: 5a 4f 4e 54 41 41 4e 20 4f 40 20 50 45 52 54 49
 0380: 40 41 41 4e 3f 20 3e 48 2f 50 20 03 0d 0a 0a
 0390: 43 4f 4e 54 43 4e 55 45 3f 20 3e 4a 2f 4e 28
 03a0: 03 0d 0a 0a 54 43 4b 20 55 57 20 54 45 4b 53 54
 03b0: 20 43 4a 20 20 41 40 53 4e 55 43 54 45 4e 20 4d
 03c0: 43 54 20 43 52 3a 20 03 7a

JUNIOR

ASCII DUMP:

0 1 2 3 4 5 6 7 8 9 a b c d e f
 0350: T E R R O R E E N K A R A K
 0360: Z N T A L D F U H O R I
 0370: K A N T A L H V U
 0380: C O N T I N U E Z C J N S
 0390: I N T A P S L U I T E N M
 03a0: E T C R

SYMBOL TABLE 2000 2100

PAGE 01

TABLE	0000	TABLE	0001	CHAR	0002	CHAR	0003
BSP	0000	TEMP	0005	COLUMN	0006	SMUL	0007
RDN	0008	SHIFT	0009	SHIFT	0000	LENGTH	0000
INDEX	000C	CNTROW	000D	INIT	0200	KRNT	0200
VERT	0238	CNTIN	024A	MORE	0250	ONES	0261
TEXTIN	0268	HIGH	0270	HIGH	0280	START	0280
NECHN	0288	PRTX	02A1	GETCH	02A5	NCRD	0288
PRCOL	02B0	SHFT	02C9	SPACE	02CF	DOT	02D8
PRDUT	02DA	TEST	02E2	BACK	02E2	LF	030A
PRDUT	02DA	TEST	02E2	CLSC	0325	RTTIME	032F
PRDUT	02DA	TEST	02E2	PRSP	11F3	PRSP	11F3
PRDUT	02DA	TEST	02E2	LABJUN	105F	LABJUN	105F
PRDUT	02DA	TEST	02E2	PRCOW	133A	PRCOW	133A
PRDUT	02DA	TEST	02E2	BKNT	147C	BKNT	147C
PRDUT	02DA	TEST	02E2	CNTD	147F	CNTD	147F

HEXADUMP:
 0 1 2 3 4 5 6 7 8 9 a b c d e f
 0500: 41 18 24 22 24 18 20 50 68 88 18 68 88 42 fe 32
 0510: 92 32 0c 10 68 88 10 68 88 10 43 7c 92 92 82 44
 0520: 70 68 88 68 68 70 44 fe 82 82 82 7c 10 68 88
 0530: 68 68 88 10 43 fe 32 32 82 18 68 88 10 68 88
 0540: 40 4e 12 12 12 82 18 68 88 10 68 88 47 7c
 0550: 02 40 12 12 12 82 18 68 88 10 68 88 40 10
 0560: 10 68 88 10 68 88 10 68 88 10 68 88 20 20
 0570: 20 20 20 20 4a 40 80 80 80 7a 00 80 80 00
 0580: 88 70 4b fe 10 20 44 82 80 50 40 80 80 80
 0590: fe 80 80 80 80 80 80 80 80 18 4d fe 04 18
 05a0: 04 fe 88 d8 48 48 88 88 4e fe 08 10 20 fe 88
 05b0: 88 48 48 98 88 88 4f 7c 82 82 82 7c 70 88 88
 05c0: 88 70 50 fe 12 12 12 8c 10 88 88 10 80 80 80
 05d0: 51 7c 82 a2 42 bc 70 88 88 88 48 30 68 52 fe 12
 05e0: 32 8c 10 88 88 10 40 30 88 53 4c 32 32 04
 05f0: 70 88 88 70 54 82 92 fe 82 82 18 20 20
 0600: 20 20 20 20 40 40 80 80 80 80 80 80 80 80
 0610: 70 50 3a 40 80 40 3a 88 88 88 88 50 20 52 fe
 0620: 40 30 40 fe 88 88 88 48 48 88 50 c0 28 10 28
 0630: 40 88 88 50 20 88 88 55 00 88 10 88 88 88 88
 0640: 50 20 20 20 3a c2 a2 32 8a 95 18 08 10 20 40
 0650: 80 18 20 80 80 80 80 80 80 80 80 80 80 21
 0660: 00 00 00 00 20 20 20 20 20 20 20 20 10 30 7c
 0670: 10 10 00 20 18 20 20 20 20 20 20 20 10 10 10 00
 0680: 00 00 18 00 00 2a 00 00 00 00 00 00 00 00
 0690: 00 00 20 3a 80 80 20 00 00 00 00 20 00 00
 06a0: 30 25 25 25 00 00 18 00 18 00 00 31 04 92
 06b0: 70 88 18 20 80 80 80 80 80 80 80 80 80 80
 06c0: 70 88 88 88 88 88 88 88 88 88 88 88 88 88
 06d0: 20 20 70 32 c4 a2 32 8c 70 88 88 30 40 80
 06e0: 18 32 42 92 3a 00 18 08 10 30 88 88 70 3a 30
 06f0: 28 24 20 18 30 50 50 18 10 10 35 4a 8a 8a
 0700: 72 18 80 18 88 88 70 30 78 94 32 32 32 38 40
 0710: 80 18 88 88 70 37 82 c2 12 8a 85 18 08 18 28 40
 0720: 40 40 30 6c 32 32 52 c2 70 88 88 70 88 70 39
 0730: 8c 32 32 52 3a 70 88 88 70 88 10 40 2c 00 80 00
 0740: 00 00 00 00 00 20 20 40 ff

JUNIOR

TUTOR

PAGE 01

```

0010: 0200          TUTOR ORG $0200
0020:
0030: MORSE-TRAININGSPROGRAMMA. KAN ALLES WAT HET GELIJKNAMIGE
0040: DATONGAPARAAT KAN. PLUS NOG EEN BEETJE MEER. NAAR HET
0050: PROGRAMMA "CODE TEST" VAN STAN OCKERS. STERK VEREENVOU-
0060: DIGDE VERSIE.
0070: HET PROGRAMMA LEVERT EEN AUDIO-SIGNAAL OP PB7 VAN DE PIA.
0080: BIJ DE IETS UITGEBREIDE JUNIOR IS DIT DE OUTPUTPOORT VAN
0090: DE CASSETTE-ROUTINE. HET PROGRAMMA SCHAKELT DE SAVE-RE-
0100: CORDER OOK IN ZODAT EENVOUDIG 'LESMATERIAAL' TE GENE-
0110: REN IS. VOOR DE BASIS-JUNIOR IS EEN SCHAKELINGETJE TE
0120: GEBRUIKEN ZOALS BESCHREVEN IN JUNIOR-BOEK DEEL 2. BLZ 38.
0130: HET GEGENEREERDE SIGNAAL BESTAAT UIT GROEPJES VAN VIJF
0140: MORSETEKENS. GESCHIEDEN DOOR EEN SPATIE.
0150: DE SNELHEID WORDT BEPAALD DOOR DE WAARDE IN SPEED: $33 =
0160: 16 WPM, $44 = 12 WPM, ETC. (VIA $201).
0170: HET AANTAL TEKENS WAARUIT WORDT GEKOZEN WORDT BEPAALD
0180: DOOR DE WAARDE IN MAX (VIA $205).
0190: DE RELATIEVE FREQUENTIE VAN MOEILIJKE TEKENS KAN VER-
0200: GROOT WORDEN DOOR DAT TEKEN MEERDERE MALEN IN DE TABEL
0210: OP TE NEMEN. MAAR PAS OP: MAX MAX IS 127 ($7F).
0220:
0230: VOOR BEGINNERS KAN HET MAKKELIJK ZIJN OM DE LETTER- EN
0240: WOORDSPATIES WAT TE VERGROTEN. ZOALS BV BIJ DE BEKENDE
0250: DARC-CURSUSSEN GEBEURT. DIT KAN DOOR DE INHOUD VAN ADRES-
0260: SEN $21F EN $22C TE VERGROTEN.
0270:
0280: DE TOONHOOGTE VAN HET AUDIOSIGNAAL KAN WORDEN VERANDERD
0290: MET DE INHOUD VAN ADRES $270.
0300:
0310: DO 00 LSTRND * $00D0 DOOR: TH.G. DRIEVER
0320: D6 00 CHRLFT * $00D6 LANKFORST 53 - 49
0330: D7 00 TEMP * $00D7 6538 KM NIJMEGEN
0340: D8 00 TEMPX * $00D8
0350: D9 00 SPEED * $00D9
0360: DA 00 MAX * $00DA
0370: DB 00 MASK * $00DB
0380: F7 1A TIMER * $1AF7
0390: 82 1A PBD * $1A82
0400: 83 1A PBDD * $1A83
0410:
0420: 0200 A9 44 START LDXIM $44 : SNELHEID 12 WPM
0430: 0202 85 D9 STA SPEED :
0440: 0204 A9 28 LDXIM $28 : GEBRUIK DE HELE TABEL
0450: 0206 85 DA STA MAX :
0460:
0470: 0208 A2 07 LDXIM $07 : BEPAAL ZO KLEIN MOGELIJK
0480: 020A 85 DB STA MASK : MASKER
0490: 020C 4A SHIFT LSRA :
0500: 020D 05 DB ORA MASK :
0510: 020F CA DEX :
0520: 0210 D0 FA BNE SHIFT :
0530: 0212 85 DB STA MASK :
0540:
0550: 0214 A9 47 IO LDXIM $47 :
0560: 0216 A2 FF LDXIM $FF :
0570: 0218 8D 82 1A STA PBD : DISPLAY GEDOOFD. OUTP. REC. AAN
0580: 021B 8E 83 1A STX PBDD : POORT B: ALLES OUTPUT
0590:
0600: 021E A2 04 GRUP LDXIM $04 : LENGTE WOORDSPATIE - LENGTE SPATIE
0610: 0220 20 78 02 JSR SPACE :
0620: 0223 A9 06 LDXIM $06 : LENGTE GROEP +1
0630: 0225 85 D6 STA CHRLFT :
0640: 0227 C6 D6 CHAR DEC CHRLFT :

```

TUTOR

PAGE 02

```

0650: 0229 F0 F3      BEQ  GRUP  : NA 5 TEKENS NIEUWE GROEP
0660: 022B A2 03      LDXIM $03 :
0670: 022D 20 78 02   JSR  SPACE :
0680: 0230 20 89 02   NUMB JSR  RND   : KIES RANDOMNUMMER
0690: 0233 25 DB      AND  MASK  : MAAK )=0. (=127 OF 63 OF 31 ETC.
0700: 0235 C5 DA      CMP  MAX   : BINNEN SELEKTIE ?
0710: 0237 10 F7      BPL  NUMB  : NEEN ? DAN OPNIEUW
0720: 0239 AA        TAX        : INDEX VOOR TABEL
0730: 023A BD 9D 02   LDAAX TABEL : HAAL CODE VOOR CHAR
0740: 023D 85 D7      STA  TEMP  :
0750: 023F 06 D7      BITS ASL  TEMP :
0760: 0241 F0 E4      BEQ  CHAR  : LEEG GESCHOVEN ? DAN KLAAR
0770: 0243 B0 0D      BCS  DASH  : CARRY GESET ? DAN STREEP..
0780: 0245 A2 01      LDXIM $01  : ..ANDERS PUNT
0790: 0247 20 5A 02   JSR  MARK  :
0800: 024A A2 01      LDXIM $01  :
0810: 024C 20 78 02   JSR  SPACE :
0820: 024F 18        CLC        :
0830: 0250 90 ED      BCC  BITS  : UNC JUMP
0840: 0252 A2 03      DASH LDXIM $03 :
0850: 0254 20 5A 02   JSR  MARK  :
0860: 0257 18        CLC        :
0870: 0258 90 F0      BCC  SPAC  : UNC JUMP
0880:
0890: 025A 86 D8      MARK STX  TEMPX : SAVE AANTAL MARKS
0900: 025C A5 D9      TIMM LDA  SPEED :
0910: 025E 8D F7 1A   TOGGLE STA  TIMER : START TELLER
0920: 0261 A9 80      TOGGLE LDAIM $80 :
0930: 0263 4D 82 1A   EOR  PBD   : TOGGLE PB7
0940: 0266 8D 82 1A   STA  PBD   :
0950: 0269 A2 20      LDXIM $20  : HALVE PERIODE TOON
0960: 026B CA        FREQ DEX      :
0970: 026C D0 FD      BNE  FREQ  :
0980: 026E 2C F7 1A   BIT  TIMER  : UITGETELD ?..
0990: 0271 10 EE      BPL  TOGGLE : ..ZO NIET VERDER PIEPEN..
1000: 0273 C6 D8      DEC  TEMPX : ..ZO JA MARK KLAAR
1010: 0275 D0 E5      BNE  TIMM   : DRIE MARKS = EEN STREEP
1020: 0277 60      RTS        :
1030:
1040: 0278 86 D8      SPACE STX  TEMPX :
1050: 027A A5 D9      TIMS LDA  SPEED :
1060: 027C 8D F7 1A   STA  TIMER  : START TELLER
1070: 027F 2C F7 1A   HOLD BIT  TIMER : WACHT TOT TELLER KLAAR IS
1080: 0282 10 FB      BPL  HOLD  :
1090: 0284 C6 D8      DEC  TEMPX : EEN SPACE GEHAD
1100: 0286 D0 F2      BNE  TIMS   : NOG MEER ?
1110: 0288 60      RTS        :
1120:
1130:
1140:
1150:
1160: 0289 18      RND CLC        :
1170: 028A D8      CLD        :
1180: 028B A5 D1      LDA  LSTRND +01 :
1190: 028D 65 D4      ADC  LSTRND +04 :
1200: 028F 65 D5      ADC  LSTRND +05 :
1210: 0291 85 D0      STA  LSTRND :
1220: 0293 A2 04      LDXIM $04  :
1230: 0295 B5 D0      ROLL LDAZX LSTRND :
1240: 0297 95 D1      STAZX LSTRND +01 :
1250: 0299 CA      DEX        :
1260: 029A 10 F9      BPL  ROLL  :
1270: 029C 60      RTS        :
1280:

```

RANDOM GENERATOR NAAR J. BUTTERFIELD
KIM USER NOTES VOL.1.#1

TUTOR

PAGE 03

1290:	029D	60	TABEL	=	\$60	A
1300:	029E	88		=	\$88	B
1310:	029F	A8		=	\$A8	C
1320:	02A0	90		=	\$90	D
1330:	02A1	40		=	\$40	E
1340:	02A2	28		=	\$28	F
1350:	02A3	D0		=	\$D0	G
1360:	02A4	08		=	\$08	H
1370:	02A5	20		=	\$20	I
1380:	02A6	78		=	\$78	J
1390:	02A7	80		=	\$80	K
1400:	02A8	48		=	\$48	L
1410:	02A9	E0		=	\$E0	M
1420:	02AA	A0		=	\$A0	N
1430:	02AB	F0		=	\$F0	O
1440:	02AC	68		=	\$68	P
1450:	02AD	D8		=	\$D8	Q
1460:	02AE	50		=	\$50	R
1470:	02AF	10		=	\$10	S
1480:	02B0	C0		=	\$C0	T
1490:	02B1	30		=	\$30	U
1500:	02B2	18		=	\$18	V
1510:	02B3	70		=	\$70	W
1520:	02B4	98		=	\$98	X
1530:	02B5	B8		=	\$B8	Y
1540:	02B6	C8		=	\$C8	Z
1550:	02B7	FC		=	\$FC	0
1560:	02B8	7C		=	\$7C	1
1570:	02B9	3C		=	\$3C	2
1580:	02BA	1C		=	\$1C	3
1590:	02BB	0C		=	\$0C	4
1600:	02BC	04		=	\$04	5
1610:	02BD	84		=	\$84	6
1620:	02BE	C4		=	\$C4	7
1630:	02BF	E4		=	\$E4	8
1640:	02C0	F4		=	\$F4	9
1650:	02C1	56		=	\$56	.
1660:	02C2	CE		=	\$CE	?
1670:	02C3	32		=	\$32	/
1680:	02C4	8C		=	\$8C	

LOAD YOUR SYM-Basic Tapes into KB-9

DUBBELADRESSERING VIA 6522

When loading tapes of the cassette library doesn't succeed because of the forward links before each line doesn't fit, the problem is easy to solve. Take care of loading the program with ID=\$FF on address \$78/9. After loading remember the contents of POINT (\$FA/B) store contents of \$FA in to \$7A, \$7C, \$7E, contents of \$FB into \$7B, \$7D, \$7F (these are the pointers Start of simple variable table, Start of array table and First location unused by program). Hereafter return back to Basic via RECOMP. The last one computes the links.

```

0200- A5 78 0190 READSYM LDA $78 FIRST LOC. OF PGM
0202- BD 70 1A 0200 STA $1A70
0205- A5 79 0210 LDA $79
0207- BD 71 1A 0220 STA $1A71
020A- A9 FF 0230 LDA $FF
020C- BD 79 1A 0240 STA $1A79 ID=FF
020F- 20 02 0B 0250 JSR $0B02 LOAD PROGRAM
0212- 20 BC 14 0260 JSR $14BC RESTORE OUTPUT
0215- A5 FA 0270 LDA $FA CHANGE POINTERS
0217- B5 7A 0280 STA $7A
0219- B5 7C 0290 STA $7C
021B- B5 7E 0300 STA $7E
021D- A5 FB 0310 LDA $FB
021F- B5 7B 0320 STA $7B
0221- B5 7D 0330 STA $7D
0223- B5 7F 0340 STA $7F
0225- 4C F1 23 0350 JMP $23F1 RECOMPUTE LINKS
0360 .EN

```

J. Buurman, Van der Fuyckstraat 136, 3232 AS Brielle

In DE 6502 KENNER nr.25 geeft Koen van Nieuwenhove een oplossing voor de dubbeladressering van de 6522 in de JUNIOR. Een andere oplossing is de volgende:

- 1) Onderbreek de verbinding tussen K6 en CS2 (den 23). Deze verbinding wordt onder de 6522 van de ene kant van de print naar de andere gevoerd. Onderbreken op deze plaats levert gelijk een vrij gat aan CS2.
- 2) Verbindt CS2 met AB8, bijv. den 17 van een van de 2114 RAM's.

Hierdoor wordt de 6522 geselecteerd als:
- het signaal VIA hoog is (1800-19FF), en
- AB8 laag is.
Dit wil zeggen op 1800 - 18FF.

J. Buurman, Van der Fuyckstraat 136, 3232 AS Brielle

TELEAC cursus C.A.D./C.A.M.

Doel van deze cursus is het geven van inzicht in het gebruik van computers en computer-gestuurde apparatuur bij het ontwerpproces en het productieproces. Computer Aided Design en Computer Aided Manufacturing bestaat uit zestien modules, waarin veel praktijkvoorbeelden in de vorm van 'case-studies' zullen worden gegeven. Een praktische cursus dus, minder hobby-gericht, maar gezien de aard van door veel van onze leden uitgeoefende beroepen, voldoende aanleiding om deze cursus te vermelden.

USING FORTH WITH THE 6502

DOOR: FRIDUS JONKMAN
STIJN STREUVELSLAAN 9
5242 GD ROSMALEN
TEL.: 04192-16146

ALS AANVULLING OP EERDERE ARTIKEL EN OVER FORTH, EN MET
NAME HET VERHAAL VAN GERARD VAN ROEKEL IN DE 6502-KENNER
NR. 28 WORDEN HIER EEN AANTAL TOELICHTINGEN EN PATCHES GE-
GEVEN OM DAT MAKKELIJKER IN FORTH JE WEG TE KUNNEN VINDEN.

DICTIONARY EN VOCABULARIES

IN FORTH WORDT GESPOKEN OVER DE DICTIONARY: DEZE BEVAT ALLE GEDEFINIËRDE WOORDEN, ONGEACHT DE VOCABULARY, WAARTOE ZE BEHOREN. DE BASIS-VOCABULARY HEET FORTH. HIERAAR KUNNEN EVENTUEEL ANDERE VOCABULARIES WORDEN GEKOPPELD: DE EDITOR EN DE ASSEMBLER ZIJN HET MEEST VOOR DE HAND LIGGEND. DE VOCABULARIES ZIJN VIA EEN LINK-ADRES (DE LAATSTE TWEE BYTES VAN DE DEFINITIE VAN EEN VOCABULARY) AAN ELKAAR GEKEND: HET LINK-ADRES VAN FORTH IS ALTIJD \$0000, DE TWEE-GEDEFINIËRDE VOCABULARY (BV. EDITOR) HEEFT ALS LINKADRES HET ADRES, WAAROP HET LINKADRES VAN FORTH IS TE VINDEN, NL. \$2FBE, ENZ.

HET ADRES WAAROP HET LINKADRES VAN DE LAATSTE GEDEFINIËRDE VOCABULARY STAAT WORDT IN DE SOURCE-LISTING AANGEDUID ALS DE INITIAL VOCABULARY LINK POINTER EN DEZE DIENT TEvens ALS PARAMETER OP ADRES \$2020 DE JUISTE WAARDE TE KRIJGEN. DIT MOET GEBEUREN BIJ HET CREËREN VAN NIEUWE VOCABULARIES OF BIJ HET VERWIJDEREN VAN GEHELE VOCABULARIES.

DIT GEBEURT OP DE VOLGENDE WIJZE:
 FORTH DECIMAL ' XXX 6 + 32 +ORIGIN !
 XXX IS HIERBIJ DE LAATSTE VOCABULARY IN JE FORTH-SYSTEEM.

HET TOEVOEGEN VAN WOORDEN AAN EEN BESTAANDE VOCABULARY, WAARBIJ DEZE IN HET SYSTEEM "GELOCKED" MOETEN WORDEN, GAAT ALS VOLGT:

```

XXX DEFINITIONS DECIMAL
LATEST 12 +ORIGIN !
HERE 28 +ORIGIN !
HERE 30 +ORIGIN !
HERE FENCE !

```

XXX IS DE VOCABULARY, WAARIN HET LAATST GEDEFINIEERDE
WOORD IS OPGENOMEN.

WANNEER EEN AANTAL WOORDEN DIE EERDER ZIJN "GELOCKED" IN
HET SYSTEEM, MOETEN WORDEN VERWIJDERD, DAN MOET HET VOLGEN-
WORDEN UITGEVOERD:

```

WEN UITGEVOERD:
XXX DECIMAL
' AAA NFA 12 +ORIGIN !
YYY DEFINITIONS
' BBB NFA 28 +ORIGIN !
' BBB NFA 30 +ORIGIN !
' BBB NFA FENCE !
' ZZZ 6 + 32 +ORIGIN !
FORGET BBB

```

WAARIN: AAA HET LAATSTE WOORD IS DAT IN HET SYSTEEM BLIJFT
BBB HET EERSTE WOORD IS DAT UIT HET SYSTEEM VER-
DWIJNT

XXX DE VOCABULARY IS WAARIN AAA VOORKOMT (DEZE RE-
(GEL IS UITERAARD ALLEEN MODIG INDIEN DE BE-
DE BESTAANDE CONTEXT NIET OVEREENKOMT MET DE
VOCABULARY. WAARIN AAA VOORKOMT)

YYY DE VOCABULARY IS WAARIN BBB VOORKOMT

(DEZE REBEL IS ALLEEN NODIG INDIEN BBB VAN EEN
ANDERE VOCABULARY DEEL UITMAAKT DAN AAA: HET
WOORD DEFINITIONS IS ALLEEN NODIG INDIEN FORTH
VOLGENS DE 79-STANDAARD WORDT GEBRUIKT)

ZZZ DE LAATSTE GEDEFINIEERDE VOCABULARY IS DIE IN
HET SYSTEEM BLIJFT (DEZE REGEL IS ALLEEN NODIG
WANNEER EEN OF MEER VOCABULARIES GEHEEL VER-
DWIJNEN).

WEL DIENT BEDACHT TE WORDEN DAT ALLEEN EEN OF MEER WOORDEN,
DIE EEN AANEENSLUITENDE REEKS VORMEN EN DIE HET LAATST AAN
HET SYSTEEM ZIJN TOEGEVOEGD OP DEZE WIJZE VERWIJDERD KUNNEN
WORDEN.

TERMINAL-ECHO I.P.V FORTH-ECHO

Wanneer je met FORTH wilt werken zonder dat FORTH elke ingedrukte toets onmiddellijk terugzendt naar de terminal, maar je de echo van het toetsenbord zelf wilt laten komen, dan moet je de inhoud van de volgende adressen wijzigen:

\$2B9E D7 WORDT 19
\$2B9E 22 WORDT 25

M.A.W.: IN DE DEFINITIE VAN HET WOORD EXPECT KOMT HET WOORD
EMIT VOOR: DIT WOORD WORDT VERVANGEN DOOR HET WOORD DROP.

[illegible]

FORTH VOOR JUNIOR IS VERKRIJGBAAR OP BAND DOOR FL. 12.50
OVER TE SCHRIJVEN OP POSTREKENING 841433 T.N.V W.L. VAN PELT
TE KRIMPEN A.D. IJSSEL OP DOOR BIJSCHRIJVING OP BANKREKE-
NING 44.11.06.471 TEN GUNSTE VAN W.L. VAN PELT BIJ DE AMRO-
BANK TE KRIMPEN A.D. IJSSEL. EEN EURO-CHEQUE MAG EVENEENS.

HEEFT U PROGRAMMA'S ONTWIKKELD IN FORTH, OF NIEUWE WOORDEN,
STUUR HET UW REDAKTIE!

STARTING FORTH: AN INTRODUCTION TO THE FORTH LANGUAGE AND
OPERATING SYSTEM FOR BEGINNERS AND PROFESSIONALS. written by
LEO BRODIE, FORTH INC., WITH A FOREWORD BY CHARLES H. MOORE.
PRINTED IN THE UNITED STATES OF AMERICA. ISBN 0-13-842930-8.

XX

FORTH O.K.

EEN NEDERLANDSTALIG BOEK GESCHREVEN OM GEBRUIKT TE WORDEN IN
COMBINATIE MET EEN FORTH-SYSTEEM.

AUTEURS: DRS. F. J. MEIJER EN F. LEMAIRE.

UITG. : WOLFKAMP, AMSTERDAM.

ISBN 90-70556-06-7 PRIJS: ca. Fl. 25.-.

DE 6502 KENNER

JUNIOR AAN 8" OF 5" FLOPPY

In editie 27 van augustus 83 maakte Koen van Nieuwenhove uit Leuven (Belgie) middels een artikel bekend dat zijn disk operating system met de controller FD 1771 klaar was voor zijn 8" drives en ook voor 5". Het systeem, dat in single density draait, evenals dat van OHIO en PROTON, maakt gebruik van de geheugenplaatsen \$1900-\$1904. Deze adressen kunnen worden vrijgemaakt zoals hij al bekend maakte in editie 25 van februari 1983. De hard- en software is ontleend aan artikelen uit BYTE van mei en juni 1981 en het boek 'A floppy disk operating system for the 8080' van Kenneth B. Welles, uitgegeven door BYTE Books. Cassettebesturing zal mogelijk blijven, terwijl de DOS al draait met de assembler van Moser, de KB9-Basic en Pascal. Koen hoopt in samenwerking met clubleden ook Micro-Ade, Forth etc. aangepast te krijgen.

Het 8" DOS maakt gebruik van de configuratie 77 tracks, 26 sectoren per track en 128 bytes per sector. Het 5" DOS heeft 40 tracks. IBM-compatible dus.

Om een indruk te geven van de bevelen, hier een opsomming: Save, Run, Limits, Go, Quit, Execute, Copy, Delete, Directory, Exit, Free, List, Read, Rename, Trace, Store, Format, Random, Undelete, Memtest, Validate, Backup, Key (sleutel toekennen aan disk), Lock (beveiligen van een file).

Het gepubliceerde artikel is verkrijgbaar door storting van Fl. 10,- op postrekening 841433 t.n.v. W.L. van Pelt te Kripen a.d. IJssel. Een Eurocheque, mits volledig ingevuld, mag ook. Of overmaken op de AMRO-bank te Kripen a.d. IJssel t.g.v. bankrekening 44.11.06.471 t.n.v. W.L. van Pelt.

FORTH voor JUNIOR is op band verkrijgbaar voor Fl. 12,50 over te schrijven.

VERKRIJGBAARHEID BASIC

Microsoft Basic KB9 of KB9 P

Johnson Computer
P.O. Box 523
Medina, OHIO 44256

Itty Bitty Computers
P.O. Box 23189
San Jose, CA 95153

MSB-Verlag
Moangoldstrasse 10
D 7778 Markdorf
West Duitsland

Basic Compiler CMB 8032 en
Commodore 64

Gesellschaft fur Math. Beratung & Software Dr. August C. Quint a.b.h.
Kaiser-Friedrich Ring 55
D 6200 Wiesbaden
Tel.: (06121)842686

5.5K zelf-ontwikkelde Basic

J.J. Janssen
Gerardsweg 30
6525 RT Nijmegen

ASSEMBLER MOSER VERKRIJGBAAR

KIM-1 versie op band

Ing.buro Schroder
Echternachlaan 161
5625 KC Eindhoven
Tel.: 040 - 421821

VERKRIJGBAARHEID PASCAL

Pascal op floppy voor C 64

Micro-Delta Holland BV
Kaarder 100
1625 TN Hoorn
Tel.: 02290 - 41555

VERKRIJGBAARHEID FOCAL

Fast Focal, een Basic-achtige interpreter met veel wiskundige functies, afkomstig van The 6502 Program Exchange, 2920 West Moana, Reno, NV 89509, America, bij:

J.J. Janssen
Gerardsweg 30
6525 RT Nijmegen.
VERMELD DE NAAM VAN UW CLUB!

CASSETTES VOOR JUNIOR

NR.1 FL. 12,50

Elektuur Eprova Programmer + Aanvulling John van Sorang. 4-Kolom printer, Conversie Dag naar week, Zenuwslag, Music Box, Dokatimer, Automaat Register Uitlezing, Lotto.

NR.2 FL. 12,50

Frequentieteller met 6522, List Mnemonics (7-seg. disp) Shift, Basicode met de SC/MP interface, RTTY, Easy Editing Supporter (PME), Single Step Debug, Wallbreaker

NR.3 FL. 1250

Renumber SYM-Basic voor de Junior, Append SYM-Basic v. Junior, Basic Compressor v. Junior, NPRCHA (nieuwe print routine op \$1634), Adress (Micro-Ade verplaatsen), Mas termind R. Visser, Outclk (Klok met 8 schakelpoorten).

Programma's in de cassettebibliotheek zijn afkomstig van onze clubleden of bewerkingen 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 alleen maar kostendekkend en niet bedoeld om de programma's te kopen. DE CLUB IS EN WORDT GEEN HANDEL!

SYM-BASIC CASSETTES v JUNIOR

ook in te lezen met KB9

NR.1 FL. 12,50

A. Bruto-netto/Netto-bruto loonberekening B. Hypotheekberekening (M. van Hintum) C. Kalender D. Eeuwkalender E. Magische vierkanten F. Hoofdrekenen G. Bunny, H. Fruitautomaat (G.v. Roekel) I. Olifant J. Zeegevecht K. Poker L. Maanlander M. Afstandsspel N. Bioritmes

NR.5 FL. 12,50

Service Rapporten Bestand.

SYM-BASIC CASSETTES v JUNIOR

NR.2 FL. 12,50

A. Maanlanding B. Olifant C. Getal Raden D. Zeegevecht E. Claire (adresbestandpr.) F. Poker G. Vier op een rij H. Zoek het woord I. Woordsoel J. Faktuur (Fr. Smeehuijzen; zie hiervoor ook de publikatie van Ruud Uhoff: (editie 28).

NR.3 FL. 17,50

A. Superstartrek (nederlands) met handleiding

NR.4 FL. 1250

A. Othello (cursorbestuurd) B. Afstandsspel C. Safari D. Letters schuiven (M. van Hintum) E. Letters/cijferssoel F. Getal verwisselen G. Autodoelhof H. "Dammen" J. X-spel K. Vier op een rij L. Puzzel M. Uitgebreide dis assembler (Gerard v. Roekel)

PATCHES ON KB-9 BASIC

In the editions 22 until 26 Koen van Nieuwenhove published his patches for the KB-9 Basic (JUNIOR-version). The patches included a routine to shift itself to the pages 2 and further. Now we can offer you the same patches excluding a shift-routine. In some situations these are more comfortable (f.i.: the club-DOS of Koen uses those pages).

The patches contain:
Trace, Step, Reset, Printer Page, Printer, Video, Auto-numbering, Renumber, Data-Save, Edit and Append. The Modem in / Modem out-command is prepared.
Price: Fl. 7,50
To be paid to Mr. W.L.v. Pelt at Kripen a.d. IJssel on the postal account 841433 or by Eurocheque.

Routine to Save & Load Basic Data Tables for JUNIOR com.

Listing available for Fl.5,00