

GEBRUIKERS CLUB een club van 6502 gebruikers **NEDERLAND**

Het doel van de vereniging is het bevorderen van de kennis uitwisseling tussen de gebruikers van 6502 computers zoals KIM, SYM, JUNIOR, AIM65, System-65, Acorn, PET en CBM, Apple, Atari, ITT2020, PC100, Ohio Scientific Challengers, etc.

Dit doel wordt onder andere gerealiseerd door vijf maal per jaar een clubbijeenkomst te houden op de derde zaterdag van oneven maanden uitgezonderd de maand juli. Ook wordt vijf maal per jaar het huisorgaan " DE 6502 KENNER " gepubliceerd, welke verschijnt op de derde zaterdag van de maanden februari, mei, augustus, oktober en december.

Naast deze vaste activiteiten kunnen de leden gebruik maken van club faciliteiten zoals de KIM-CLUB-KIM , de KIM-CLUB-JUNIOR en de cassette bibliotheek.

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

De samenstelling van het bestuur is thans als volgt :

Dagelijks bestuur	Voorzitter	Anton Müller
	Secretaris	Ruud Uphoff
	Penningmeester	Ted Schouten
	Accomodatie	Riche van Steen
	Hardware	Peter Visser
	Ledenadministratie	Bob van de Oudewetering
	Organisatie	Rinus Vleesch Dubois
	Redactie	Hans Otten
	Software	Sebo Woldringh
Het adres van de vereniging is	KIM GEBRUIKERS CLUB NEDERLAND	
	Voorburgpad 10	
	6843 EM ARNHEM	
	tel. 085 - 816935 op werkdagen van 19.00 tot 20.00	

Penningmeester : giro nummer 3757649 tnv T.Schouten Junoplts 57 Haarlem
f 40,- per kalenderjaar

DE 6502 KENNER

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

Adres voor het in-
zenden van en re-
akties op artike-
len voor de 6502

KENNER:

Redaktiesekretaris

Hans Otten

Ottoborrengeod 33

3871 MJ HOEVELAKEN

Redactie 6502 KENNER:

Anton Müller

Hans Otten

Willem van Pelt

Geheel of gedeelte-
lijke overname van
de inhoud van de
6502 KENNER zonder
toestemming van het
bestuur is verboden.

Toepassen van gepu-
bliceerde programma's,
hardware etc. is al-
leen voor persoonlijk
gebruik toegestaan.

COPYRIGHT © 1981
KIM Gebruikers club
Nederland.

Verschijnt vijf maal
per jaar.

INHOUDSOPGAVE 6502 KENNER nr 19 DECEMBER 1981

1. Inhoudsopgave
2. Van het bestuur, Anton Müller
3. Van de redactie, Hans Otten
4. UITNODIGING CLUB BIJEENKOMST
6. Schrijf- en leesroutine voor de
JUNIOR, Dick Blok
15. Rapportcijfer, Rein Duyts
22. Display op oscilloscoop, C.Totté
28. Aanpassingen aan de Macro-
Assembler van C.W.Moser,
Van Nieuwenhove Koen
34. Patches op BASIC, Hans Otten
35. Lichtshow voor de JUNIOR en de KIM,
K.Kikke
38. 4 kolom printer, A.S. Hankel
40. Printer routine voor H14,
Paul de Beer en Hans Otten
43. BREAK voor JUNIOR en KIM,
Frans Mepschen
46. Hex dump , Frans Smeehuyzen
48. Cassette bibliotheek
49. Vraag en Aanbod , Agenda

6502 VAN HET BESTUUR

Deze keer wil het eens hebben over een activiteit die U misschien best aanspreekt. Een paar weken geleden kreeg ik een telefoontje van John van Sprang uit Krimpen aan de IJssel met het verzoek of wij hem wilden steunen met het opstarten van lokale club bijeenkomsten in Den Haag. Na overleg met het dagelijks bestuur is hieruit het volgende naar voren gekomen :

Er wordt een up to date ledenlijst verstrekt aan de initiatiefnemer ; de kosten voor het produceren en verzenden van de uitnodiging voor de eerste bijeenkomst komen voor rekening van de KIM club, alsmede de zaalhuur voor de eerste bijeenkomst, een en ander binnen zekere voor het bestuur aanvaardbare grenzen. De eerste bijeenkomst in Den Haag is inmiddels geweest en was een geweldig succes. De opkomst was 130 %, dat wil zeggen er werden 20 mensen uitgenodigd en er kwamen er 26. Er was van te voren geen programma vastgesteld. Naar ik heb vernomen komen allen volgendekeer weer. Voelt U zelf iets voor het starten van lokale of regionale bijeenkomsten neem dan eens contact op met ondergetekende of de secretaris. Het is overigens wel de bedoeling dat qua financieën dergelijke bijeenkomsten zichzelf bedruipen.

Dan nu even iets anders . Wij kunnen voor de redactie nog steeds medewerkers gebruiken die de beschikking hebben over een ontwikkel systeem met Micro Ade en/of ASM/TED van C.W. Moser voor het vertalen van de assembler sources en het produceren van listings op papier. Hoe meer wij het werk voor de 6502 KENNER kunnen verdelen hoe beter en des te meer aandacht kan er aan elk programma worden besteed, hetgeen de kwaliteit van de 6502 KENNER alleen maar ten goede kan komen.

Wat U minimaal nodig heeft is een systeem met 8K RAM en een beeldscherm of toetsenbord met video interface of iets dergelijks en de mogelijkheid om cassettes aan te kunnen maken en enige kennis van de 6502 assembly language. De rest komt vanzelf. Als we zo'n stuk of vijf medewerkers kunnen krijgen dan blijven de werkzaamheden beperkt tot zo'n 50 uur per jaar per medewerker en dat is toch niet teveel gevraagd. Ook kunnen we in de redactie nog wel mensen gebruiken die van een kladje of iets dergelijks een behoorlijk hardware schema kunnen tekenen, hetgeen zo nu en dan eens voorkomt. Heeft U interesse neem dan eens even contact op met de secretaris of ondergetekende. Behalve dat het nuttig werk voor de club is steek je er zelf nog wat van op in de vorm van contacten en kennis. Bovendien help je mee aan het in stand houden van een vereniging van zo'n ruim 400 leden ; een vereniging waar je graag lid van zou worden en dat waarschijnlijk om het clubblad : DE 6502 KENNER.

Anton Müller
Voorzitter

6502 KENNER 19 ligt hier voor U, het laatste nummer in 1981 en het laatste nummer waarin U mijn naam als medewerker zal tegenkomen aan de 6502 KENNER en het bestuur van de KIM club. Twee jaar lang heb ik mogen helpen met de club activiteiten en in die twee jaar de club wat leden aantal betreft zien verdrievoudigen zonder dat het eind van die groei in zicht is gelukkig. Aan die groei heb ik wat geholpen door wat contacten naar buiten te leggen voor de club maar drie personen wil ik noemen die het gepresteerd hebben alles in goede administratieve banen te leiden : Anton Müller , secretaris in 1980 en nu voorzitter en Ruud Uphoff , de huidige secretaris, de financieën werden uitstekend beheerd door Ted Schouten. Deze personen zijn van onschatbare betekenis geweest voor de club. Zelf stop ik nu met actieve deelneming aan het club gebeuren gedwongen door mijn drukke werkkring en de activiteiten die ik als redacteur van Radio Bulletin ontplooi. Wel hoop ik nog vele jaren als lid van de club samen met U onze hobby en ook mijn vak nu te mogen beoefenen.

Een van de fijnste ervaringen in de KIM club is geweest samen met mede hobbyisten te kunnen werken aan de 6502 soft- en hardware op een gelijke basis. Allen doen we ons werk voor de club in onze vrije tijd. Het clubblad wordt ook zonder enige commerciële binding of oogmerk gepubliceerd. We laten daarbij regelmatig een steek vallen maar hopen dat U dat ons niet kwalijk neemt. De 6502 KENNER is een blad gewijd aan de 6502 voor en door leden van de KIM club. De leden dragen zelf de artikelen etc aan waarmee het clubblad gevuld kan worden. Het resultaat mag gerust uniek worden genoemd, in Nederland maar ook internationaal. Voor een gevaar wil ik nog waarschuwen. De meeste beginnende computer hobbyisten denken op een volledig verkeerde manier over hobby computers . Hobby of personal computers zijn namelijk gewone computers en het schrijven van software verschilt nauwelijks met dat voor mini en mainframe computers. Luister en lees daarom de ervaringen van ervaren programmeurs en hun ideeën over gestructureerd programmeren : Vind niet opnieuw het wiel uit ! Tot ziens.

Hans Otten

6502 CLUB BIJeenKOMST

U I T N O D I G I N G K I M C L U B B I J E E N K O M S T = = = = =

Locatie : Forbo Krommenie
Krommenie - Assendelft

Tijdstip : 16 januari 1982 10.00 tot 17.00 uur

Kosten : f 20,- inclusief koffie, lunch en frisdrankje

Route : zie ommezijde

Programma :

9.30 Ontvangst met koffie .

10.00 Opening , welkomswoord door voorzitter en gastheer.

10.15 Lezing door Ted Schouten.

Onderwerp : Temperatuurmeting mbc AIM65.

11.00 Koffie pauze

11.15 Lezing door Hans Otten.

Onderwerp : De 6809 microprocessor hard en software.

Programmeer model en instructieset en belang

daarvan voor reentrant position indepent programme-
ren , het OS/9 operating system

12.00 Lunchpauze

12.45 Deel 2 van de lezing over de 6809 door Hans Otten

13.30 Inleiding gestructureerd programmeren met voorbeelden in

Basic en Assembler door Fred Harthoorn.

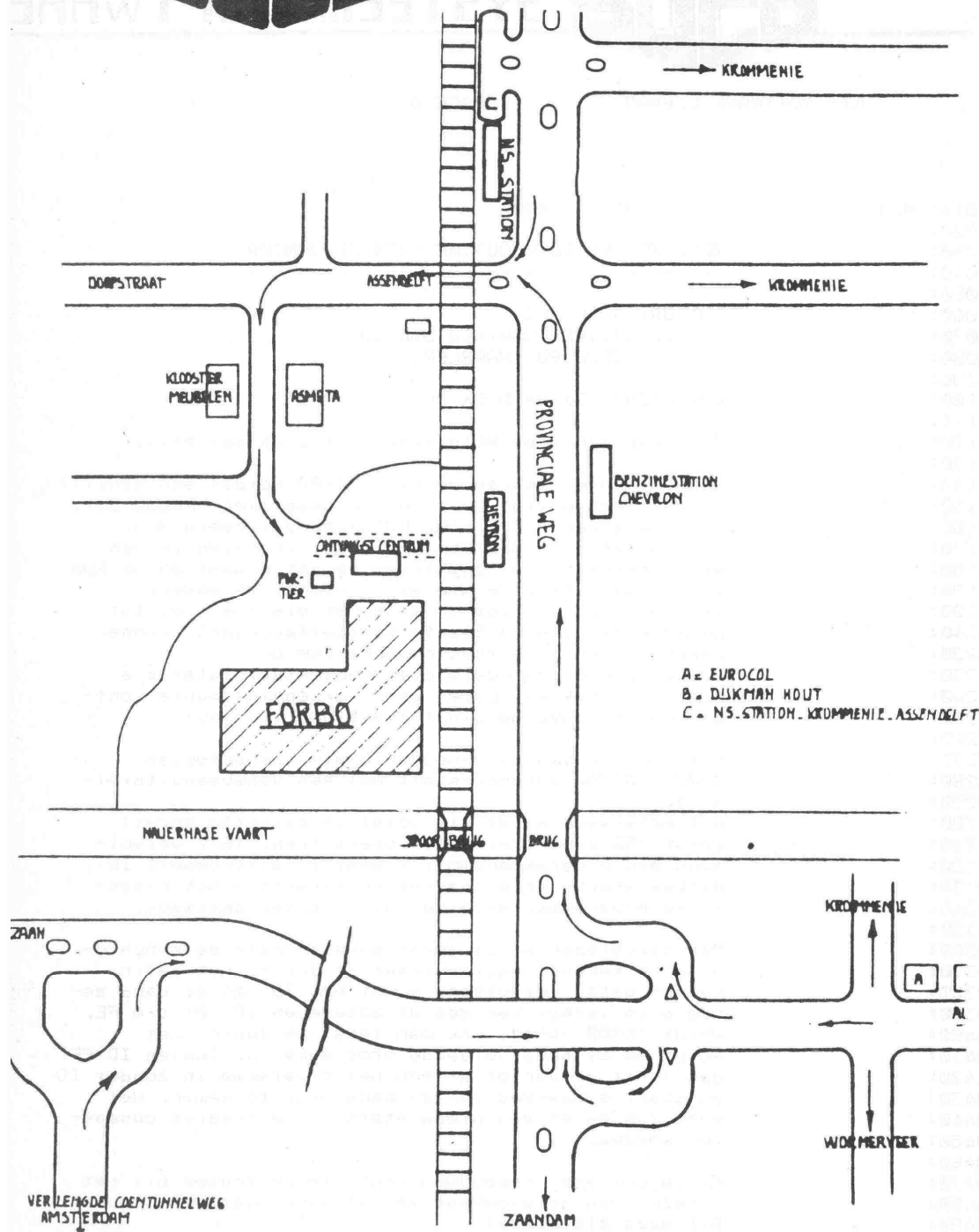
14.15 Theepauze, forum en markt

15.00 Behandeling instructie set van de 6502

15.30 tot 17.00 Informele discussies demonstraties etc

6502

CLUB BIJeenkomst



6502 SYSTEEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 01

0010: 0200

ORG \$0200

0020:

SCHRIJF- EN LEESROUTINE VOOR DE JUNIOR

0040:

0050:

0060:

AUTEUR: DICK BLOK

0070:

..... RIJKSSTRAATWEG 594 2H

0080:

2026 RD HAARLEM

0090:

0100:

COPYRIGHT (C) BY DICK BLOK

0110:

0120:

Met dank aan Sebo Woldringh voor zijn adviezen.

0130:

0140:

0150:

0160:

0170:

0180:

0190:

0200:

0210:

0220:

0230:

0240:

0250:

0260:

0270:

0280:

0290:

0300:

0310:

0320:

0330:

0340:

0350:

0360:

0370:

0380:

0390:

0400:

0410:

0420:

0430:

0440:

0450:

0460:

0470:

0480:

0490:

0500:

In KIM-Kenner 14 van december 1980 werden een schrijf- en een leesroutine voor de veelbesproken Kansas City interface van M.C.P. (Nu CHIP) gepubliceerd door Sebo Woldringh. Daarmee heeft hij voorzien in een grote behoefte, al ontbraken op dat moment enige kwaliteitsaspecten. Om hieraan tegemoet te komen, speciaal met het oog op diegenen die nog niet tot de aanschaf van de Elektuur-interfacekaart (kunnen) overgaan, is deze routine ontwikkeld. Diegenen die inmiddels over de Elektuur-interfacekaart beschikken zullen veel herkenningpunten ontdekken. (Bronvermelding: zie einde listing).

Het gebruik van de geheugenruimte, de adressen 0200 - 03E5, veronderstelt wel een geheugenuitbreiding.

Het programma wordt als volgt op cassette gezet: eerst 256 synchronisatie tekens (hex. 16), vervolgens een programmanummer (identificatienummer: ID), daarna start- en eindadres en tenslotte het eigenlijke programma, gevolgd door 2 bytes checksum.

Met teruglezen wordt eerst gekeken naar de synchronisatie tekens, daarna zoekt de Junior het begin van de data, vervolgens wordt het ID van de band gehaald en vergeleken met de opgegeven ID, 01 t/m FE. Wordt ID=00 opgegeven, dan leest de Junior het eerst op de band volgende programma in. Indien ID=FF, dan leest de Junior alleen het programma in zonder ID en start/eindadres van de band over te nemen. Wel moet dan eerst een nieuw start- en eindadres opgegeven worden.

Er is ook een ingebouwde controle op fouten bij het inlezen van de gegevens van de band naar de computer. Dit gaat als volgt:

Tijdens het schrijven op de band wordt alle data bij

6502 SYSTEEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 02

510: elkaar opgeteld in een veld van 2 bytes en wordt be-
520: waard op adres 1A73 en 1A74, respectievelijk CHKL en
530: CHKH. Dit noemen we de checksum (controlegetal).
540: Bij het teruglezen wordt dit ook gedaan. Aan het eind
550: van het programma staat de checksum op de band. Als
560: alles is ingelezen wordt de checksum op de band verge-
570: leken met de berekende checksum. Stemt dit overeen,
580: dan wordt teruggesprongen naar de resetroutine en
590: op het display verschijnt, in tegenstelling tot het
600: eindadres van de elektuurroutine, het startadres van
610: het zojuist ingelezen programma. Er kan dan meteen
620: worden gestart.
630: Gaat er iets fout met het van de band lezen, dan wordt
640: teruggesprongen naar de monitor en verschijnt het
650: startadres van het leesprogramma op het display. Men
660: kan het inlezen dan opnieuw proberen.
670:
680: De volgende adressen zijn belangrijk om te onthouden:
690:
700: SAL = 1A75 (Start adres laag)
710: SAH = 1A76 (Start adres hoog)
720: EAL = 1A77 (Eind adres laag)
730: EAH = 1A78 (Eind adres hoog)
740: ID = 1A79 (Identificatie nummer)
750:
760: 0200 = Start address schrijfprogramma.
770: 02AE = Start address leesprogramma.
780:
790: Zet men de routines in een andere geheugenruimte, dan
800: moeten ook de adressen 0362 en 0365 worden aangepast.
810:
820: De opgegeven versie van het programma werkt op onge-
830: veer 100 Baud, zoals het programma uit KIM-Kenner 14.
840: Voor wijziging in 300 Baud zie het commentaar
850: bij de equates en timer.
860: Voor wijzigingen aan de cassette-interface en het ge-
870: bruik van een voorversterker, zie "Ervaringen met
880: de Junior" van Wim van Pelt en John van Sprang.
890:
900: Schrijven gebeurt via PB5.
910: Lezen gebeurt via PB6.
920:
930:
940:
950:
960:
970:
980:
990:
000:

6502 SYSTEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 03

```

1010:
1020:
1030:
1040: 0200
1050: 0200
1060: 0200
1070: 0200
1080: 0200
1090:
1100: 0200
1110: 0200
1120:
1130:
1140:
1150:
1160: 0200
1170: 0200
1180: 0200
1190: 0200
1200: 0200
1210: 0200
1220:
1230:
1240:
1250:
1260: 0200
1270: 0200
1280: 0200
1290: 0200
1300: 0200
1310: 0200
1320: 0200
1330:
1340:
1350:
1360:
1370: 0200
1380:
1390: 0200
1400:
1410: 0200
1420:
1430:
1440:
1450:
1460:
1470:
1480:
1490:
1500:

```

JUNIOR MONITOR DEFINITIES:

```

-----
PAD      *      $1A80      PIA A data register
PADD     *      PAD      +01 PIA A data direction register
PBD      *      PADD     +01 PIA B data register
PBDD     *      PBD      +01 PIA B data direction register
TIMER    *      $1AF7      Timer voor 100 Baud.
(Deze locatie wijzigen in 1AF6 voor 300 Baud).
RESET    *      $1C1D      Reset routine
SHOW     *      $1DCC      7 segment display routine

```

PAGINA NUL LOCATIES:

```

-----
TEMP     *      $00E0
INL      *      $00F8
INH      *      INL      +01
POINTL   *      INH      +01
POINTH   *      POINTL   +01
TEMPX    *      POINTH   +02

```

VASTE LOCATIES IN PAGINA 1A:

```

-----
CHKL     *      $1A73
CHKH     *      CHKL     +01
SAL      *      CHKH     +01
SAH      *      SAL      +01
EAL      *      SAH      +01
EAH      *      EAL      +01
ID       *      EAH      +01

```

E Q U A T E S:

```

-----
HALF     *      $0005      Halve bit tijd bij 100 Baud
(Wijzigen in $001A bij 300 Baud).
HEEL     *      $000A      Hele bit tijd bij 100 Baud
(Wijzigen in $0034 bij 300 Baud).
TWEETH   *      $0018      2 1/2 bit tijd bij 100 Baud
(Wijzigen in $007E bij 300 Baud).

```

6502 SYSTEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 04

```

510:                                     S C H R I J F R O U T I N E :
520:                                     -----
530:
540: 0200 A9 20      WRITE LDAIM $20      Zet data direction register
550: 0202 8D 83 1A  STA PBDD            PB5 op output
560: 0205 A9 00      LDAIM $00            Maak inhoud checksum schoon
570: 0207 8D 73 1A  STA CHKL
580: 020A 8D 74 1A  STA CHKH
590: 020D A2 FF      LDXIM $FF            Schrijf 256 SYNC tekens weg
600:
610: 020F A9 16      WSYNC LDAIM $16      (Synchronisatie teken)
620: 0211 86 FD      STX  TEMPX
630: 0213 20 6F 02  JSR  OBYTE
640: 0216 A6 FD      LDX  TEMPX
650: 0218 CA        DEX
660: 0219 D0 F4      BNE  WSYNC
670: 021B A9 2A      LDAIM $2A            Schrijf start data teken weg
680: 021D 20 6F 02  JSR  OBYTE
690: 0220 AD 79 1A  LDA  ID              Schrijf identificatie nummer weg
700: 0223 20 6F 02  JSR  OBYTE
710: 0226 AD 75 1A  LDA  SAL            Schrijf start adres weg
720: 0229 85 F8      STA  INL
730: 022B 20 6C 02  JSR  OCHKT
740: 022E AD 76 1A  LDA  SAH
750: 0231 85 F9      STA  INH
760: 0233 20 6C 02  JSR  OCHKT
770: 0236 AD 77 1A  LDA  EAL            Schrijf eind adres weg
780: 0239 20 6C 02  JSR  OCHKT
790: 023C AD 78 1A  LDA  EAH
800: 023F 20 6C 02  JSR  OCHKT
810:
820: 0242 A0 00      WDATA LDYIM $00      Schrijf data weg
830: 0244 B1 F8      LDAIY INL            Haal een byte op
840: 0246 20 6C 02  JSR  OCHKT
850: 0249 E6 F8      INC  INL            Verhoog start adres met 1
860: 024B D0 02      BNE  WDTAA
870: 024D E6 F9      INC  INH
880:
890: 024F A5 F8      WDTAA LDA  INL      Indien startadres = eindadres + 1
900: 0251 CD 77 1A  CMP  EAL            dan berekende checksum op tape
910: 0254 D0 EC      BNE  WDATA          zetten en terug naar reset.
920: 0256 A5 F9      LDA  INH
930: 0258 CD 78 1A  CMP  EAH
940: 025B D0 E5      BNE  WDATA
950: 025D AD 73 1A  LDA  CHKL
960: 0260 20 6F 02  JSR  OBYTE
970: 0263 AD 74 1A  LDA  CHKH
980: 0266 20 6F 02  JSR  OBYTE
990: 0269 4C 1D 1C  JMP  RESET
000:

```


6502 SYSTEEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 05

```

2010: 026C 20 9C 02 OCHKT JSR CHKT Wegg schrijven met checksum
2020:
2030: 026F 18 OBYTE CLC Wegg schrijven zonder checksum
2040: 0270 48 PHA Stop te schrijven byte op de stack
2050: 0271 20 89 02 JSR OUTBI en geef start bit
2060: 0274 68 PLA Haal te schrijven byte op
2070: 0275 A0 08 LDYIM #08 Aantal bits per byte in Y
2080:
2090: 0277 0A OBTEA ASLA Schuif bit in carry flag
2100: 0278 48 PHA Restant op stack
2110: 0279 20 89 02 JSR OUTBI Schrijf een bit weg
2120: 027C 68 PLA Haal resterende bits weer op
2130: 027D 88 DEY Herhaal dit tot dat
2140: 027E D0 F7 BNE OBTEA we 8 bits hebben gehad.
2150: 0280 38 SEC Schrijf 2 stop bits
2160: 0281 20 89 02 JSR OUTBI
2170: 0284 38 SEC
2180: 0285 20 89 02 JSR OUTBI
2190: 0288 60 RTS Ga terug naar hoofd routine
2200:
2210: 0289 A2 00 OUTBI LDXIM #00 Schrijf een bit
2220: 028B 90 01 BCC OUTA naar PBS
2230: 028D CA DEX
2240: 028E 8E 82 1A OUTA STX PBD
2250: 0291 A9 0A LDAIM HEEL Zet hele bit tijd
2260: 0293 8D F7 1A STA TIMER voor timer en
2270: 0296 2C F7 1A OUTB BIT TIMER laat timer aflopen
2280: 0299 10 FB BPL OUTB
2290: 029B 60 RTS Ga terug naar hoofd routine
2300:
2310: 029C A8 CHKT TAY Bereken de check sum
2320: 029D 18 CLC
2330: 029E 6D 73 1A ADC CHKL
2340: 02A1 8D 73 1A STA CHKL
2350: 02A4 AD 74 1A LDA CHKH
2360: 02A7 69 00 ADCIM #00
2370: 02A9 8D 74 1A STA CHKH
2380: 02AC 98 TYA
2390: 02AD 60 RTS
2400:
2410: L E E S R O U T I N E :
2420: -----
2430:
2440: 02AE A9 00 READ LDAIM #00 Zet data direction register voor input
2450: 02B0 8D 83 1A STA PBDD
2460: 02B3 8D 73 1A STA CHKL Maak checksum schoon
2470: 02B6 8D 74 1A STA CHKH
2480: 02B9 20 81 03 WAIT JSR RDPBD Wacht tot er wat komt
2490: 02BC D0 FB BNE WAIT
2500: 02BE 20 87 03 SYNCA JSR RBYTE Lees een byte en

```

6502 SYSTEEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 06

2510:	02C1	C9	16		CMPIM	\$16	kijk of het een SYNC is.
2520:	02C3	D0	F9		BNE	SYNCA	
2530:	02C5	A2	0A		LDXIM	\$0A	Lees 10 SYNC tekens
2540:	02C7	20	87	03	SYNCB	JSR	RBYTE
2550:	02CA	C9	16		CMPIM	\$16	
2560:	02CC	D0	F0		BNE	SYNCA	
2570:	02CE	CA			DEX		
2580:	02CF	D0	F6		BNE	SYNCB	
2590:	02D1	20	87	03	DBGIN	JSR	RBYTE
2600:	02D4	C9	2A		CMPIM	\$2A	Wacht op data begin teken
2610:	02D6	F0	06		BEQ	LDID	
2620:	02D8	C9	16		CMPIM	\$16	
2630:	02DA	D0	E2		BNE	SYNCA	
2640:	02DC	F0	F3		BEQ	DBGIN	
2650:	02DE	20	87	03	LDID	JSR	RBYTE
2660:	02E1	CD	79	1A	CMP	ID	Lees identifikatie nummer
2670:	02E4	F0	25		BEQ	LDSA	Indien gelijk dan naar laadroutine
2680:	02E6	AD	79	1A	LDA	ID	
2690:	02E9	C9	00		CMPIM	\$00	Indien 00 ook naar laadroutine
2700:	02EB	F0	1E		BEQ	LDSA	
2710:	02ED	C9	FF		CMPIM	\$FF	Indien FF dan slaan we behalve de ID
2720:	02EF	F0	02		BEQ	IGNRE	ook het start- en eind adres over
2730:	02F1	D0	BB		BNE	READ	Zoek naar volgende file.
2740:	02F3	A2	04		LDXIM	\$04	
2750:	02F5	20	87	03	IGNRA	JSR	RBYTE
2760:	02F8	20	9C	02		JSR	CHKT
2770:	02FB	CA			DEX		
2780:	02FC	D0	F7		BNE	IGNRA	
2790:	02FE	AD	75	1A	LDA	SAL	Haal nieuw start- en eindadres op
2800:	0301	85	F8		STA	INL	en zet die op de juiste plaats
2810:	0303	AD	76	1A	LDA	SAH	
2820:	0306	85	F9		STA	INH	
2830:	0308	4C	33	03		JMP	RDATA
2840:	030B	20	87	03	LDSA	JSR	RBYTE
2850:	030E	20	9C	02		JSR	CHKT
2860:	0311	85	F8		STA	INL	Ga data lezen.
2870:	0313	8D	75	1A	STA	SAL	Laad start adres van tape met checksum
2880:	0316	20	87	03		JSR	RBYTE
2890:	0319	20	9C	02		JSR	CHKT
2900:	031C	85	F9		STA	INH	berekening
2910:	031E	8D	76	1A	STA	SAH	
2920:	0321	20	87	03	LDEA	JSR	RBYTE
2930:	0324	20	9C	02		JSR	CHKT
2940:	0327	8D	77	1A	STA	EAL	Laad eind adres van tape met checksum
2950:	032A	20	87	03		JSR	RBYTE
2960:	032D	20	9C	02		JSR	CHKT
2970:	0330	8D	78	1A	STA	EAH	berekening
2980:	0333	20	87	03	RDATA	JSR	RBYTE
2990:	0336	20	9C	02		JSR	CHKT
3000:	0339	A0	00		LDYIM	\$00	Lees data van tape met checksum-

6502 SYSTEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 07

3010:	033B	91 F8	STAIY	INL	Store character op geïndexeerde adres
3020:	033D	E6 F8	INC	INL	Verhoog adres met 1
3030:	033F	D0 02	BNE	RDTAA	
3040:	0341	E6 F9	INC	INH	
3050:	0343	A5 F8	RDTAA	LDA	INL
3060:	0345	CD 77 1A		CMP	EAL
3070:	0348	D0 E9		BNE	RDATA
3080:	034A	A5 F9		LDA	INH
3090:	034C	CD 78 1A		CMP	EAH
3100:	034F	D0 E2		BNE	RDATA
3110:	0351	20 87 03		JSR	RBYTE
3120:	0354	CD 73 1A		CMP	CHKL
3130:	0357	D0 08		BNE	ERROR
3140:	0359	20 87 03		JSR	RBYTE
3150:	035C	CD 74 1A		CMP	CHKH
3160:	035F	F0 0B		BEQ	END
3170:	0361	A9 AE	ERROR	LDAIM	READ
3180:	0363	85 FA		STA	POINTL
3190:	0365	A9 02		LDAIM	READ
3200:	0367	85 FB		STA	POINTH
3210:	0369	4C 1D 1C		JMP	RESET
3220:	036C	AD 75 1A	END	LDA	SAL
3230:	036F	85 FA		STA	POINTL
3240:	0371	AD 76 1A		LDA	SAH
3250:	0374	85 FB		STA	POINTH
3260:	0376	4C 1D 1C		JMP	RESET
3270:	0379	8D F7 1A	DELAY	STA	TIMER
3280:	037C	2C F7 1A		BIT	TIMER
3290:	037F	10 F8		BPL	DELAY
3300:	0381	AD 82 1A	RDPBD	LDA	PBD
3310:	0384	29 40		ANDIM	\$40
3320:	0386	60		RTS	
3330:	0387	20 81 03	RBYTE	JSR	RDPBD
3340:	038A	D0 FB		BNE	RBYTE
3350:	038C	20 81 03		JSR	RDPBD
3360:	038F	D0 F6		BNE	RBYTE
3370:	0391	A9 05		LDAIM	HALF
3380:	0393	20 79 03		JSR	DELAY
3390:	0396	A0 08		LDYIM	\$08
3400:	0398	48		PHA	
3410:	0399	A9 0A	RBTEA	LDAIM	HEEL
3420:	039B	20 79 03		JSR	DELAY
3430:	039E	18		CLC	
3440:	039F	F0 01		BEQ	RBTEB
3450:	03A1	38		SEC	
3460:	03A2	68	RBTEB	PLA	
3470:	03A3	2A		ROLA	
3480:	03A4	48		PHA	
3490:	03A5	88		DEY	
3500:	03A6	D0 F1		BNE	RBTEA

6502 SYSTEEM SOFTWARE

KIM SOFTWARE LIBRARY

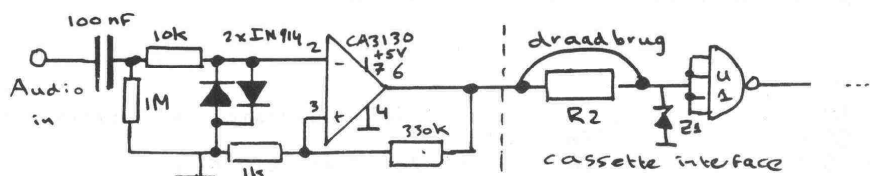
PAGE 08

3510:	03A8	68		PLA		Red accumulator in Temporary storage.
3520:	03A9	85	E0	STA	TEMP	
3530:	03AB	A9	18	LDAIM	TWEEH	Zet 2 1/2 bit tijd
3540:	03AD	20	C2 03	JSR	VUTAP	
3550:	03B0	A5	E0	LDA	TEMP	Haal data character
3560:	03B2	60		RTS		Terug naar hoofd routine.
3570:	03B3	20	87 03	BEGIN JSR	RBYTE	Lees een byte en
3580:	03B6	C9	16	CMPIM	\$16	kijk of het een SYNC is.
3590:	03B8	F0	07	BEQ	UIT	Zo ja terug,
3600:	03BA	A9	0A	LDAIM	HEEL	Zo nee, wacht een hele bit tijd en
3610:	03BC	20	79 03	JSR	DELAY	doe het nog eens
3620:	03BF	10	F2	BPL	BEGIN	
3630:	03C1	60		UIT RTS		
3640:	03C2	8D	F7 1A	VUTAP STA	TIMER	Zet 2 1/2 bit tijd uit
3650:	03C5	86	FD	STX	TEMPX	Save X-register
3660:	03C7	A9	BF	LDAIM	\$BF	PB6 insang, rest poort B uitsang
3670:	03C9	8D	83 1A	STA	PBDD	
3680:	03CC	A9	7F	LDAIM	\$7F	Poort A uitsang
3690:	03CE	8D	81 1A	STA	PADD	
3700:	03D1	A0	04	LDYIM	\$04	4 X Display
3710:	03D3	A2	10	DISPL LDXIM	\$10	Digit 5 aan de beurt
3720:	03D5	A5	E0	LDA	TEMP	Neem data byte en display hem
3730:	03D7	20	CC 1D	JSR	SHOW	
3740:	03DA	88		DEY		
3750:	03DB	D0	F6	BNE	DISPL	
3760:	03DD	A6	FD	LDX	TEMPX	Haal X op
3770:	03DF	2C	F7 1A	WACHT BIT	TIMER	Wacht tot 2 1/2 bit tijd om is.
3780:	03E2	10	FB	BPL	WACHT	
3790:	03E4	60		RTS		

Bronnen:

- 1) Lees- en schrijfroutine van Sebo Woldringh in KIM-Kenner 14.
- 2) Programma SUPERTAPE van Y.L. Bicknese in KIM-Kenner 12
- 3) Junior computer boek, deel 1 en 2 van Elektuur
- 4) Microcomputer systems van Camp, Smay en Triska
- 5) KIM-1 Monitor programma (Mos Technology)
- 6) Programma VUTAPE uit First Book of KIM.

6502 SYSTEM SOFTWARE



Bovenstaande schakeling is een voorversterker voor de cassette interface. R2 (100 ohm) uit de cassette interface moet zoals in de tekening aangegeven worden overbrugd met een draadbrug. Testen van de cassette interface is mogelijk door een opname van 2400 HZ te maken en door R4 te variëren (potmeter van maken) de LED constant te laten branden.

De volgende twee routines zijn in samenwerking met de lees- en schrijf routines te gebruiken om de interface te testen. De eerste produceert alleen nullen op de band , de tweede afwisselend een nul en en één.

```

0100 A9 20    BEGIN LDA $20    maak PB5 uitgang
0102 8D 83 1A    STA PBDD    voor audio uit
0105 18          CLC          nul
0106 20 89 02    JSR OUTBI    schrijf nul
0109 4C 00 01    JMP BEGIN    herhaal
    
```

```

0100 A9 20    BEGIN LDA $20    maak PB5 uitgang
0102 8D 83 1A    STA PBDD    voor audio uit
0105 18          CLC          nul
0106 20 89 02    JSR OUTBI    schrijf nul
0109 38          SEC          één
010A 20 89 02    JSR OUTBI    schrijf één
010D 4C 00 01    JMP BEGIN    herhaal
    
```




APPLICATIE

KIM SOFTWARE LIBRARY

PAGE 02

0510:

0520:

0530:

0540:

0550:

0560:

0570:

0580:

0590:

0600:

0610:

0620:

0630:

0640:

0650:

0660:

0670:

0680: 0200

0690: 0200

0700: 0200

0710: 0200

0720: 0200

0730: 0200

0740: 0200

0750: 0200

0760: 0200

0770: 0200

0780:

0790:

0800:

0810:

0820: 0200

0830: 0200

0840:

0850: 0200 20 01 03

0860: 0203 86 49

0870: 0205 20 37 03

0880: 0208 A6 49

0890: 020A C9 12

0900: 020C D0 06

0910: 020E 20 AB 02

0920: 0211 4C 03 02

0930: 0214 C9 11

0940: 0216 D0 06

0950: 0218 20 C0 02

0960: 021B 4C 03 02

0970: 021E C9 10

0980: 0220 D0 06

0990: 0222 20 89 02

1000: 0225 4C 03 02

Toetsfuncties:

+ : Omhoog in tabel

DA : Omlaag in tabel

AD : Clear

PC : Spring naar het begin van de tabel

F : Spring naar het begin van de tabel voor
een nieuwe serie

C : -, trekt 0,25 van het cijfer af

D : +, telt 0,25 bij het cijfer op

A : Wordt gebruikt voor het cijfer 10.

Page zero locaties:

ZWAAR * \$0000 Zwaartes

KEY * \$0049

TEMPA * KEY +01

TEMPB * TEMPA +01

TEMPC * TEMPB +01

TEMPD * TEMPC +01

TABEL * \$0050

INH * \$00F9

POINTL * INH +01

POINTH * POINTL +01

Junior monitor routines:

SCANDS * \$1D8E

GETKEY * \$1DF9

MAIN JSR INIT

MAINA STX KEY

JSR KEYDIS

LDX KEY

CMPIM \$12 '+'

BNE MAINB

JSR PLUS

JMP MAINA

MAINB CMPIM \$11 'DA'

BNE MAINC

JSR MIN

JMP MAINA

MAINC CMPIM \$10 'AD'

BNE MAIND

JSR CLEAR

JMP MAINA

6502

APPLICATIE

KIM SOFTWARE LIBRARY

PAGE 03

```
1010: 0228 C9 14      MAIND  CMPIM $14      'PC'
1020: 022A D0 06      BNE  MAINE
1030: 022C 20 27 03    JSR  LYST
1040: 022F 4C 03 02    JMP  MAINA
1050: 0232 C9 0F      MAINE  CMPIM $0F      'F'
1060: 0234 D0 03      BNE  MAINF
1070: 0236 4C 00 02    JMP  MAIN
1080: 0239 C9 0C      MAINF  CMPIM $0C      'C'
1090: 023B D0 06      BNE  MAING
1100: 023D 20 EB 02    JSR  GMIN
1110: 0240 4C 03 02    JMP  MAINA
1120: 0243 C9 0D      MAING  CMPIM $0D      'D'
1130: 0245 D0 06      BNE  MAINH
1140: 0247 20 D5 02    JSR  GPLUS
1150: 024A 4C 03 02    JMP  MAINA
1160: 024D C9 0A      MAINH  CMPIM $0A      'A'
1170: 024F D0 06      BNE  MAINI
1180: 0251 20 22 03    JSR  TIEN
1190: 0254 4C 03 02    JMP  MAINA
1200: 0257 C9 13      MAINI  CMPIM $13      'GO'
1210: 0259 D0 06      BNE  MAINJ
1220: 025B 20 4A 03    JSR  REKEN
1230: 025E 4C 03 02    JMP  MAINA
1240: 0261 20 6B 02    MAINJ  JSR  SHIFT
1250: 0264 05 FA      ORA  POINTL
1260: 0266 85 FA      STA  POINTL
1270: 0268 4C 03 02    JMP  MAINA
1280:
1290: 026B A0 04      SHIFT  LDYIM $04
1300: 026D 06 4B      SHIFTA  ASL  TEMPB
1310: 026F 26 FA      ROL  POINTL
1320: 0271 26 FB      ROL  POINTH
1330: 0273 88      DEY
1340: 0274 D0 F7      BNE  SHIFTA
1350: 0276 60      RTS
1360:
1370: 0277 A5 FA      STORE  LDA  POINTL
1380: 0279 95 50      STAAX TABEL
1390: 027B A5 FB      LDA  POINTH
1400: 027D 95 51      STAAX TABEL +01
1410: 027F 60      RTS
1420:
1430: 0280 85 50      SHOW  LDAAX TABEL
1440: 0282 85 FA      STA  POINTL
1450: 0284 85 51      LDAAX TABEL +01
1460: 0286 85 FB      STA  POINTH
1470: 0288 60      RTS
1480:
1490: 0289 A9 00      CLEAR  LDAIM $00
1500: 028B 85 FA      STA  POINTL
```


6502

APPLICATIE

KIM SOFTWARE LIBRARY

PAGE 04

```

1510: 028D 85 FB      STA  POINTH
1520: 028F 85 4B      STA  TEMPB
1530: 0291 60          RTS
1540:
1550: 0292 A5 FA      SETCYF LDA  POINTL Het 1e cijfer komt op
1560: 0294 F0 14      BEQ   SETCYC de 2e plaats van links; behalve
1570: 0296 C9 10      CMPIM $10 bij de 10.
1580: 0298 F0 0A      BEQ   SETCYB
1590: 029A A5 FB      SETCYA LDA  POINTH
1600: 029C D0 0C      BNE   SETCYC
1610: 029E 20 6B 02   JSR   SHIFT
1620: 02A1 4C 9A 02   JMP   SETCYA
1630: 02A4 20 6B 02   SETCYB JSR   SHIFT
1640: 02A7 20 6B 02   JSR   SHIFT
1650: 02AA 60          SETCYC RTS
1660:
1670: 02AB 20 92 02   PLUS   JSR   SETCYF
1680: 02AE 20 77 02   JSR   STORE
1690: 02B1 E8          INX
1700: 02B2 E8          INX
1710: 02B3 A5 F9      LDA   INH
1720: 02B5 18          CLC
1730: 02B6 F8          SED
1740: 02B7 69 01      ADCIM $01
1750: 02B9 D8          CLD
1760: 02BA 85 F9      STA   INH
1770: 02BC 20 80 02   JSR   SHOW
1780: 02BF 60          RTS
1790:
1800: 02C0 20 92 02   MIN    JSR   SETCYF
1810: 02C3 20 77 02   JSR   STORE
1820: 02C6 CA          DEX
1830: 02C7 CA          DEX
1840: 02C8 A5 F9      LDA   INH
1850: 02CA 38          SEC
1860: 02CB F8          SED
1870: 02CC E9 01      SBCIM $01
1880: 02CE D8          CLD
1890: 02CF 85 F9      STA   INH
1900: 02D1 20 80 02   JSR   SHOW
1910: 02D4 60          RTS
1920:
1930: 02D5 20 92 02   GPLUS   JSR   SETCYF
1940: 02D8 18          CLC
1950: 02D9 F8          SED
1960: 02DA A5 FA      LDA   POINTL
1970: 02DC 69 25      ADCIM $25
1980: 02DE 85 FA      STA   POINTL
1990: 02E0 A5 FB      LDA   POINTH
2000: 02E2 69 00      ADCIM $00

```



APPLICATIE

KIM SOFTWARE LIBRARY

PAGE 05

```
2010: 02E4 85 FB          STA  POINTH
2020: 02E6 D8            CLD
2030: 02E7 20 77 02      JSR  STORE
2040: 02EA 60            RTS
2050:
2060: 02EB 20 92 02  GMIN JSR  SETCYF
2070: 02EE 38            SEC
2080: 02EF F8            SED
2090: 02F0 A5 FA          LDA  POINTL
2100: 02F2 E9 25          SBCIM $25
2110: 02F4 85 FA          STA  POINTL
2120: 02F6 A5 FB          LDA  POINTH
2130: 02F8 E9 00          SBCIM $00
2140: 02FA 85 FB          STA  POINTH
2150: 02FC D8            CLD
2160: 02FD 20 77 02      JSR  STORE
2170: 0300 60            RTS
2180:
2190: 0301 20 89 02  INIT JSR  CLEAR
2200: 0304 AA            TAX
2210: 0305 85 49          STA  KEY
2220: 0307 95 50          INITA STAAX TABEL
2230: 0309 E8            INX
2240: 030A E0 50          CPXIM $50
2250: 030C D0 F9          BNE  INITA
2260: 030E A9 01          LDAIM $01
2270: 0310 85 F9          STA  INH
2280: 0312 A2 00          LDXIM $00
2290: 0314 60            RTS
2300:
2310: 0315 A5 FB          LINKS LDA  POINTH
2320: 0317 29 0F          ANDIM $0F
2330: 0319 D0 06          BNE  LINKSA
2340: 031B 20 6B 02      JSR  SHIFT
2350: 031E 4C 15 03      JMP  LINKS
2360: 0321 60          LINKSA RTS
2370:
2380: 0322 A9 10          TIEN  LDAIM $10
2390: 0324 85 FA          STA  POINTL
2400: 0326 60            RTS
2410:
2420: 0327 20 92 02  LYST JSR  SETCYF
2430: 032A 20 77 02      JSR  STORE
2440: 032D A2 00          LDXIM $00
2450: 032F A9 01          LDAIM $01
2460: 0331 85 F9          STA  INH
2470: 0333 20 80 02      JSR  SHOW
2480: 0336 60            RTS
2490:
2500:
```



APPLICATIE

KIM SOFTWARE LIBRARY

PAGE 06

```
2510:
2520: 0337 20 8E 1D KEYDIS JSR SCANDS
2530: 033A D0 FB BNE KEYDIS
2540: 033C 20 8E 1D KEYDIA JSR SCANDS
2550: 033F F0 FB BEQ KEYDIA
2560: 0341 20 8E 1D JSR SCANDS
2570: 0344 F0 F6 BEQ KEYDIA
2580: 0346 20 F9 1D JSR GETKEY
2590: 0349 60 RTS
2600:
2610: 034A 20 92 02 REKEN JSR SETCYF De hierna volgende instructies
2620: 034D 20 77 02 JSR STORE zorgen voor de optelling.
2630: 0350 20 89 02 JSR CLEAR
2640: 0353 85 F9 STA INH
2650: 0355 AA TAX
2660: 0356 A8 TAY
2670:
2680: 0357 B9 00 00 REKENA LDAAY ZWAAR De zwaarte
2690: 035A 85 4A STA TEMPB
2700: 035C F0 32 BEQ REKENE Als zwaarte 0 is, zijn we klaar.
2710: 035E B5 50 LDAAX TABEL Als het getal 0 is, moeten we
2720: 0360 D0 07 BNE REKENB naar het volgende getal, zonder
2730: 0362 B5 51 LDAAX TABEL +01 dat de zwaartes worden
2740: 0364 D0 03 BNE REKENB opgeteld.
2750: 0366 4C 8A 03 JMP REKEND
2760:
2770: 0369 A5 4A REKENB LDA TEMPB De zwaartes worden opgeteld
2780: 036B 18 CLC antwoord in INH
2790: 036C F8 SED
2800: 036D 65 F9 ADC INH
2810: 036F D8 CLD
2820: 0370 85 F9 STA INH
2830:
2840: 0372 F8 REKENC SED Optellen van de cijfers, antwoord
2850: 0373 A5 4B LDA TEMPB in POINTH, POINTL, TEMPB
2860: 0375 75 50 ADCAX TABEL
2870: 0377 85 4B STA TEMPB
2880: 0379 A5 FA LDA POINTL
2890: 037B 75 51 ADCAX TABEL +01
2900: 037D 85 FA STA POINTL
2910: 037F A5 FB LDA POINTH
2920: 0381 69 00 ADCIM $00
2930: 0383 85 FB STA POINTH
2940: 0385 C6 4A DEC TEMPB
2950: 0387 D0 E9 BNE REKENC
2960: 0389 D8 CLD
2970:
2980: 038A E8 REKEND INX
2990: 038B E8 INX
3000: 038C C8 INY
```

6502

APPLICATIE

KIM SOFTWARE LIBRARY

PAGE 07

```
3010: 038D 4C 57 03      JMP    REKENA
3020:
3030: 0390 20 15 03      REKENE JSR    LINKS  Het antwoord wordt zover
3040:                                     mogelijk naar links
3050:                                     geschoven.
3060:
3070: 0393 A9 00      LDAIM $00  De hierna volgende instructies
3080: 0395 85 4A      STA    TEMPA  zorgen voor de deling.
3090: 0397 85 4C      STA    TEMPC
3100: 0399 85 4D      STA    TEMPD
3110: 039B 8D 4D 00      STA
3120: 039E F8      SED
3130:
3140: 039F A5 4A      REKENF LDA    TEMPA  Er wordt 1 bij het antwoord
3150: 03A1 18      CLC      opgeteld. Antwoord in
3160: 03A2 69 01      ADCIM $01  TEMPD, TEMPC, TEMPA
3170: 03A4 85 4A      STA    TEMPA
3180: 03A6 A5 4C      LDA    TEMPC
3190: 03A8 69 00      ADCIM $00
3200: 03AA 85 4C      STA    TEMPC
3210: 03AC A5 4D      LDA    TEMPD
3220: 03AE 69 00      ADCIM $00
3230: 03B0 85 4D      STA    TEMPD
3240: 03B2 38      SEC      Er wordt 1 keer de deler (INH)
3250: 03B3 A5 4B      LDA    TEMPB  afgetrokken van
3260: 03B5 E5 F9      SBC    INH    POINTH, POINTL, TEMPB
3270: 03B7 85 4B      STA    TEMPB
3280: 03B9 A5 FA      LDA    POINTL
3290: 03BB E9 00      SBCIM $00
3300: 03BD 85 FA      STA    POINTL
3310: 03BF A5 FB      LDA    POINTH
3320: 03C1 E9 00      SBCIM $00
3330: 03C3 85 FB      STA    POINTH
3340: 03C5 30 03      BMI    REKENG  Bij een getal >8 slaat hij meteen af
3350: 03C7 4C 9F 03      JMP    REKENF  daarvan 'links' niet helemaal.
3360:
3370: 03CA 38      REKENG SEC      Er is 1 keer teveel afgetrokken
3380: 03CB A5 4A      LDA    TEMPA  dat wordt gecorrigeerd; bovendien
3390: 03CD E9 01      SBCIM $01  wordt het antwoord nu op
3400: 03CF 85 4B      STA    TEMPB  POINTH, POINTL, TEMPB gezet
3410: 03D1 A5 4C      LDA    TEMPC
3420: 03D3 E9 00      SBCIM $00
3430: 03D5 85 FA      STA    POINTL
3440: 03D7 A5 4D      LDA    TEMPD
3450: 03D9 E9 00      SBCIM $00
3460: 03DB 85 FB      STA    POINTH
3470: 03DD 08      CLD
3480: 03DE 20 15 03      JSR    LINKS
3490: 03E1 20 6B 02      JSR    SHIFT
3500: 03E4 60      RTS
```

6502 SYSTEEM SOFTWARE

DISPLAY OP OSCILLOSCOOP

C. Totté
Graaf Albrechtweg 4
3331 HP Zwijndrecht
Tel. 078 - 127324 na 17.00 u

Voor het eigenlijke ontwerp te beschrijven wil ik eerst beschrijven hoe mijn KIM is uitgebreid. In het bekende gat van 0400-13FF zit 8K RAM en vanaf 2000 8K RAM. Die 8K RAM bevindt zich voor de helft op de RAM/EPROM kaart uit Elektuur. Eventuele bouwers van die kaart wil ik de volgende tip geven. Velen zullen de 2716 gebruiken. Elektuur schrijft dat de EPROM's in de volgorde IC25, IC27, IC26 en IC28 worden geadresseerd, zie Elektuur 203 blz 51 en 52. Dit kan worden opgeheven door niet de draadbruggen R-P en T-S te leggen maar T-P en R-S, van geïsoleerd montage draad. Nu is de volgorde IC25, IC26, IC27, IC28, als je naar de tekening op blz 49 zie je waarom dit zo is. Verder hangt aan de inputlijnen PA1-PA7 (7 bits) een ASCII toetsenbord, de strobe daarvan zit met een transistor aan de IRQ. (zie afb.1).

Voor het hier beschreven display zijn nodig natuurlijk een oscilloscoop en twee Digitaal naar Analooq Converters (DAC's) opgebouwd uit 11 weerstanden en 9 buffers, de ene DAC is een 6 bit en de andere een 3 bit DAC. De 6 bit DAC hangt aan de X ingang van de scoop en de 3 bit DAC aan de Y ingang. (afb.2) Let op ! De elektronen bundel van mijn scoop wordt naar links afgebogen bij een positieve ingangsspanning. Voor scoop's met een gewone afbuiging kan ipv de 4050 een inverterende 4049 buffer worden gebruikt.

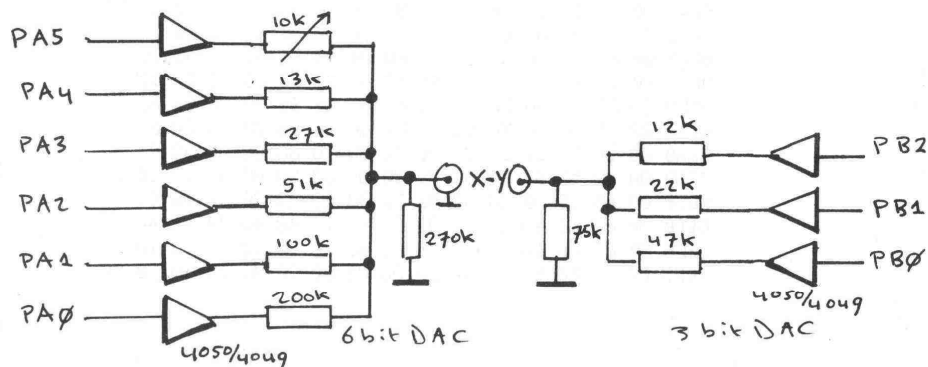
Het is nu de bedoeling door zeer snel wegsturen van verschillende spanningen op de scoop een puntjes display te veroorzaken. Er zijn dan $64 \times 8 = 512$ puntjes die onafhankelijk kunnen worden aangestuurd. Het is duidelijk dat hiervoor 64 bytes moeten worden gereserveerd. Ik heb gekozen voor 0140-017F, ja, midden in de stack. Voor het zichtbaar maken van de puntjes is een zo snel mogelijk programma gemaakt dat net als SCANDS dat vereist, zo vaak mogelijk moet worden aangeroepen.

6502 SYSTEM SOFTWARE

Voor het display heb ik drie routines geschreven :

1. DISP . Deze routine laat de in het geheugen opgeslagen puntjes zien (0140 - 017F)
2. SHIFT. Deze routine gebruikt de ASCII code in A om het adres te berekenen in de tabel waar de punten bij dit ASCII karakter staan, deze worden dan in 0180 - 0187 gezet. Daarna wordt de hele regel (10½ karakter) een punt naar links geschoven, ook de letter in 0180 - 0187 wordt een punt naar links geschoven. Daarna wordt de puntjes buffer gedurende 0,041 s getoond, waarna weer een puntje wordt ingeschoven etc. Een karakter neemt 6 punten in de breedte in, waarvan het zesde uit is. Door de LDAIM \$07 in lokatie \$0F79 te veranderen in LDAIM \$07 worden er tussen de karakters twee punten uit gelaten. Dan gaan er nog maar 9 karakters op een regel helaas. Een karakter is 8 punten hoog.
3. SPACE, INPUT en OUT zorgen dat de gewenste letter op het display komt, een Return (\$0D) levert twee spaties op, simpel te veranderen in een of geen spaties.

De INPUT is specifiek voor mijn IRQ routine, die namelijk de ASCII code in een buffertje zet. De INPUT routine roept DISP aan en laat de de display buffer zien. In BUFFER (\$1780) staat hoeveel karakters er in de buffer zitten. Als er geen karakters in de buffer staan wordt gewacht op invoer, anders het eerste karakter uit de buffer gehaald, via OUT wordt het karakter getoond en uiteindelijk aan de vragende routine terug gegeven. Essentieel is deze invoer routine niet.



6502 SYSTEM SOFTWARE

Geheugen indeling :

\$0140 - \$017F te displayen punten
64x8

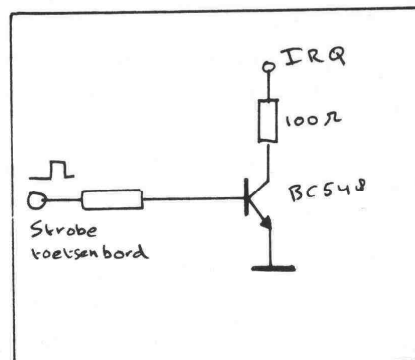
\$0180 - \$0187 nieuwe in te schuiven
karakter

\$0D00 - \$0EFF karakter punten tabel

\$0F00 - \$0FDE routines DISP, SHIFT
en SPACE, INPUT en OUT

Tabel met punten, 1 letter is 8 bytes, bevat de ASCII codes

\$21 tm \$5F , adres karakter = \$0D00 + ((ASCII-code -\$21) AND \$3F)



```

0000 00 00 00 00 00 00 00 00 20 00 20 20 20 20
0010 00 00 00 00 00 00 50 50 00 00 50 F8 50 F8 50
0020 00 20 F0 20 70 A0 70 20 00 90 90 40 20 10 C0 C0
0030 00 60 90 A0 40 A0 90 60 00 00 00 00 00 30 10
0040 00 10 20 20 20 20 20 10 00 40 20 20 20 20 40
0050 00 00 80 50 F0 50 80 00 00 00 20 20 F0 20 00
0060 30 10 00 00 00 00 00 00 00 00 00 F0 00 00 00
0070 00 20 00 00 00 00 00 00 00 80 40 40 20 10 00
0080 00 70 80 C0 A0 90 80 70 00 70 20 20 20 20 60 20
0090 00 F0 80 40 30 80 80 70 00 70 80 80 70 00 80 70
00A0 00 00 00 00 F0 80 80 80 00 70 80 00 F0 00 00 F0
00B0 00 70 80 80 F0 80 80 70 00 80 80 40 20 10 00 F0
00C0 00 70 80 80 70 80 80 70 00 70 80 80 70 80 80 70
00D0 00 20 00 00 20 00 00 00 60 20 00 00 20 00 00 00
00E0 00 00 10 20 40 20 10 00 00 F0 00 00 F0 00 00 00
00F0 00 80 40 20 10 20 40 80 00 20 00 20 20 10 40 30
0E00 00 70 80 B0 A0 80 80 70 00 80 80 80 F0 80 80 70
0E10 00 F0 40 40 70 40 40 F0 00 70 80 80 80 80 80 70
0E20 00 F0 40 40 40 40 40 F0 00 F0 80 80 F0 80 80 F0
0E30 00 80 80 80 F0 80 80 F0 00 70 80 90 80 80 80 70
0E40 00 80 80 80 F0 80 80 80 00 70 20 20 20 20 20 70
0E50 30 40 80 80 80 80 80 70 00 80 90 A0 C0 A0 90 80
0E60 00 F0 80 80 80 80 80 80 00 80 80 80 A0 A0 D0 80
0E70 00 80 80 90 A0 C0 80 80 00 70 80 80 80 80 80 70
0E80 80 80 80 80 F0 80 80 F0 00 60 90 A0 80 80 80 70
0E90 00 80 90 A0 F0 80 80 F0 00 70 80 80 70 80 80 70
0EA0 00 20 20 20 20 20 20 F0 00 70 80 80 80 80 80 80
0EB0 00 20 50 50 50 80 80 80 00 80 D0 A0 A0 80 80 80
0EC0 00 80 80 50 20 50 80 80 00 20 20 70 80 80 80
0ED0 00 F0 80 40 20 10 80 F0 00 70 40 40 40 40 40 70
0EE0 00 00 80 10 20 40 80 80 00 70 10 10 10 10 10 70
0EF0 00 80 80 50 50 50 20 00 00 00 20 40 B0 40 20 00

```

6502 SYSTEM SOFTWARE

```

0010: *****
0020: *
0030: * DISPLAY VAN 64*8 PUNTEN *
0040: * OP EEN OSCILLOSCOOP *
0050: * *
0060: * C. TOTTE 1981 *
0070: *
0080: *****
0090: ;
0100: ; DEFENITIES :
0110: ;
0120: 0F00 DISP ORG $0F00
0130: ;
0140: 0F00 DDPB * $1703 ; Y-SPANNING
0150: 0F00 PB * $1702
0160: 0F00 DDPB * $1741
0170: 0F00 PA * $1740
0180: 0F00 UX * $00E8
0190: 0F00 DISBYT * $00EA
0200: 0F00 COUNT * $00EB
0210: ;
0240: 0F00 A9 08 INIT LDARL $08 ; 8 PUNTEN
0250: 0F02 85 EB STAZ COUNT
0260: 0F04 A9 3F LDARL $3F
0270: 0F06 00 41 17 ORA DDPB ; INIT I/O POORTEN
0280: 0F09 8D 41 17 STA DDPB
0290: 0F0C A9 07 LDARL $07
0300: 0F0E 00 03 17 ORA DDPB
0310: 0F11 8D 03 17 STA DDPB
0320: 0F14 A9 3F LDARL $3F
0330: 0F16 85 E8 STAZ UX
0340: 0F18 A9 00 LDARL $00
0350: 0F1A A8 TRY
0360: 0F1B AA TAX
0370: 0F1C B9 40 01 START LDARL $0140 ; DISPLAY BUFFER BEGIN
0380: 0F1F 85 EA STAZ DISBYT BEWAAR DAT BYTE
0390: 0F21 26 EA ROTATE ROLZ DISBYT ; SCHUIF BYTE ROND IN CARRY
0400: 0F23 90 00 BCC TEL
0410: 0F25 A5 E8 LDARL UX
0420: 0F27 8D 40 17 STA PA ; OP DAC ZETTEN
0430: 0F2A 8E 02 17 STX PB
0440: 0F2D C6 E8 TEL DECZ UX ; 64 TOT 0 AFTELLEN
0450: 0F2F 30 06 BMI INCUY
0460: 0F31 C6 EB DECZ COUNT
0470: 0F33 F0 07 BEQ INCV
0480: 0F35 D0 EA BNE ROTATE
0490: 0F37 E8 INCUY INX
0500: 0F38 A9 3F LDARL $3F ; 64*8 DISPLAY
0510: 0F3A 85 E8 STAZ UX
0520: 0F3C C8 INCV INV
0530: 0F3D A9 08 LDARL $08
0540: 0F3F 85 EB STAZ COUNT
0550: 0F41 C0 40 CPYIM $40
0560: 0F43 D0 D7 BNE START
0570: 0F45 60 RTS ; ZO NIET DAN AFGELOPEN

```


6502 SYSTEM SOFTWARE

```

0010: *****
0020: *
0030: * SCHUIF EEN LETTER IN HET DISPLAY *
0040: * *
0050: * C. TOTTE 1981 *
0060: * *
0070: *****
0080: ;
0090: 0F50 SHIFT ORG $0F50 ;
0100: ;
0110: ; DEFINES
0120: ;
0130: 0F50 SAVE * $00ED
0140: 0F50 ACCU * $00E9
0150: 0F50 TIME * $0028
0160: 0F50 POINTL * $00EA
0170: 0F50 TIMER * $1747
0180: 0F50 POINTH * $00EB
0190: 0F50 DISP * $0F00
0200: 0F50 COUNT * $00EC
0210: ;
0220: 0F50 85 E9 START STAZ ACCU SAVE ACCU
0230: 0F52 C9 61 CMPIN $61 MARK KLEINE LETTERS=GROTE LETTERS
0240: 0F54 90 02 BCC CONT
0250: 0F56 E9 20 SBCIM $20
0260: 0F58 38 CONT SEC
0270: 0F59 E9 20 SBCIM $20
0280: 0F5B 30 56 BMI END ALS HET EEN CTRL CHARACTER WAS,DAN END
0290: 0F5D 29 3F ANDIM $3F
0300: 0F5F 0A ASLA
0310: 0F60 0A ASLA
0320: 0F61 A2 03 LDXIM $03
0330: 0F63 86 EB STXZ POINTH
0340: 0F65 18 CLC
0350: 0F66 69 80 ADCIM $80
0360: 0F68 26 EB ROLZ POINTH
0370: 0F6A 0A ASLA
0380: 0F6B 26 EB ROLZ POINTH
0390: 0F6D 85 EA STAZ POINTL
0400: 0F6F A0 07 LDYIM $07
0410: 0F71 B1 EA LOOP LDRIY POINTL
0420: 0F73 99 00 01 STARY $0100
0430: 0F76 88 DEY
0440: 0F77 10 F8 BPL LOOP HET NIEUWE TEKEN IS NU GEPLAATST
0450: 0F79 A9 06 LDIM $06
0460: 0F7B 85 EC STAZ COUNT 6* NAAR LINKS SCHUIVEN
0470: 0F7D A2 07 LOOPD LDXIM $07 8 RIJEN VAN 64 BIT SCHUIVEN
0480: 0F7F 86 ED STXZ SAVE
0490: 0F81 A9 01 LDIM $01
0500: 0F83 85 EB STAZ POINTH
0510: 0F85 A5 ED LOOPB LDZ SAVE WELKE REGEL ?
0520: 0F87 0A ASLA
0530: 0F88 0A ASLA
0540: 0F89 0A ASLA ACCU * 8
0550: 0F8A 18 CLC
0560: 0F8B 69 40 ADCIM $40

```

6502 SYSTEEM SOFTWARE

```

0570: 0F8D 85 EA      STAZ POINTL WIJST EEN BYTE UIT DE LINKSE KOLOM AAN
0580: 0F8F A6 ED      LDXZ SAVE
0590: 0F91 3E 80 01    ROLAX $0180
0600: 0F94 A0 07      LDVIN $07      SHIFT 8 BYTES
0610: 0F96 B1 EA      LOOPA LDAIV POINTL DOE HET
0620: 0F98 2A        ROLA
0630: 0F99 91 EA      STAIRV POINTL
0640: 0F9B 88        DEV
0650: 0F9C 10 F8      BPL LOOPA
0660: 0F9E C6 ED      DECZ SAVE
0670: 0FA0 10 E3      BPL LOOPB
0680: 0FA2 A9 28      LDRAIM TIME
0690: 0FA4 8D 47 17   STA TIMER
0700: 0FA7 20 00 0F   LOOPC JSR DISP
0710: 0FAA 2C 47 17   BIT TIMER
0720: 0FAD 10 F8      BPL LOOPC
0730: 0FAF C6 EC      DEC COUNT
0740: 0FB1 D0 CA      BNE LOOPD
0750: 0FB3 A5 E9      END LDA ACCU
0760: 0FB5 60        RTS

```

```

0010: *****
0020: *
0030: * I/O MET TOETSENBOORD EN SCOP
0040: *
0050: * C. TOTTE 1981
0060: *****
0070: ;
0080: ; DEFINES
0090: ;
0100: 0FB6 IO ORG $0FB6 ;
0110: 0FB6 BUFFER * $1780
0120: 0FB6 DISP * $0F00
0130: 0FB6 SHIFT * $0F50
0140: 0FB6 A9 20 SPACE LDRAIM $20
0150: ;
0160: 0FB8 4C C9 0F JMP OUT
0170: 0FBB 20 00 0F INPUT JSR DISP
0180: 0FBE AE 80 17 LDX BUFFER IS ER EEN TOETS INGEDRUKT ?
0190: 0FC1 F0 F8 BEQ INPUT
0200: 0FC3 CE 80 17 DEC BUFFER
0210: 0FC6 B0 80 17 LDRAW BUFFER HAAL DIE TOETS
0220: 0FC9 48 OUT PHA SAVE DE ACCU
0230: 0FCA C9 0D CMPIM $0D IS HET CR ?
0240: 0FCC D0 08 BNE VERDA
0250: 0FCE 20 B6 0F JSR SPACE
0260: 0FD1 20 B6 0F JSR SPACE
0270: 0FD4 68 WEG PLA
0280: 0FD5 60 RTS
0290: ;
0300: 0FD6 C9 7F VERDA CMPIM $7F IS HET DEL ?
0310: 0FD8 F0 FA BEQ WEG
0320: 0FDA 20 50 0F JSR SHIFT
0330: 0FDD 68 PLA
0340: 0FDE 60 RTS

```

6502 SYSTEEM SOFTWARE

Aanpassingen aan de MACRO ASSEMBLER
van Moser

Van Nieuwenhove Koen
Consciencestraat 60 bus 4
B-3000 Leuven
Belgie

Deze aanpassingen zijn bedoeld voor een Junior-computer, uitgebreid met een interfacekaart, met minstens 8-K ram, liever 16K. Verder is ook nog de elekterminal of een andere gelijkaardige terminal vereist. Aan mijn systeem hangt ook nog een parallelprinter (80-kolom). Deze heb ik er aan gekoppeld via de vrije 6522 op de interfacekaart.

Ik kocht mijn versie aan bij U.Schroder. Deze levert bij de cassette en het handboek nog een aantal bladen bij die uitleg geven over een mogelijke aanpassing. Het is de bedoeling dat dit samen met mijn uitleg gebruikt wordt. Ik zal mijn best doen zo volledig mogelijk te zijn.

Wat betreft echter de Junior-gebruikers die nu nog deze assembler bij U.Schroder zouden willen aankopen moet ik wel opmerken dat ik aan U.Schroder een copy gestuurd heb van een bij mij WERKENDE versie, zodat hij nu misschien wel al een Junior-versie ter beschikking heeft. Wie weet!

```
0030 ;
0040 ;DE INPUT-OUTPUT NAAR DE TERMINAL
0050 ;
0060 JUN/OUT .DE $1334
0070 DELAY .DE $3F87
0080 ;
0090 .BA $3FD6
3FD6- A9 1F 0100 CRT/CR.FND LDA ##1F
3FD8- 20 87 3F 0110 JSR DELAY
3FDB- A9 0A 0120 LDA ##0A
3FDD- 20 E8 3F 0130 JSR CRT/OUTPUT
3FE0- A9 1E 0140 LDA ##1E
3FE2- 20 87 3F 0150 JSR DELAY
3FE5- EA 0160 NOP
3FE6- EA 0170 NOP
3FE7- 60 0180 RTS
0190 ;IK HEB DIT STUKJE ER ALLEEN BIJGEVOEGD OMDAT
0200 ;DE RTS OP DE MIJ GELEVERDE CASSETTE ER NIET
0210 ;STOND.
3FE8- 48 0220 CRT/OUTPUT PHA
3FE9- EA 0230 NOP
3FEA- EA 0240 NOP
3FEB- EA 0250 NOP
3FEC- EA 0260 NOP
3FED- EA 0270 NOP
3FEE- 20 34 13 0280 JSR JUN/OUT
3FF1- 68 0290 PLA
3FF2- C9 0D 0300 CMP ##0D
3FF4- F0 E0 0310 BEQ CRT/CR.FND
3FF6- 60 0320 RTS
```

6502 SYSTEEM SOFTWARE

```

0330 ;KEYBOARD INPUT
0340 ;
0350 JUN/IN      .DE $12AE
0360 ;
0370            .BA $3C89
3C89- 20 AE 12  0380      JSR JUN/IN
0390 ;
0400 ;
0410            .BA $3C9F
3C9F- 20 F1 3B  0420      JSR $3BF1
0430 ;
0440 ;HET BOVENSTAANDE, "JSR $3BF1", MAG U
0450 ;PLAATSEN IDIEN U ECHO VAN HET INPUT-
0460 ;KARAKTER KAN VERMIJDEN. BIJ GEZAMEN-
0470 ;LIJK GEBRUIK VAN JUNIOR EN ELEKTERMINAL
0480 ;IS DIT ZEER EENVOUDIG TE VERWEZENLIJKEN.
0490 ;MEN LAAT DAN DE VERBINDING TUSSEN P EN
0500 ;Q OP DE JUNIOR EN TUSSEN U EN V OP DE
0510 ;ELEKTERMINAL OPEN.
0520 ;
0530 ;
0540 ;BREAKTEST-ROUTINE
0550 ;
0560            .BA $3BD2
3BD2- 20 D5 3B  0570      JSR BREAKTEST
0580 ;
0590 ;
3BD5- 2C 80 1A  0600 BREAKTEST BIT $1A80
0610            CLC
3BD8- 18        0620            BMI NO/BREAK
3BD9- 30 09      0630 BRK/WAIT BIT $1A80
3BDB- 2C 80 1A  0640            BPL BRK/WAIT
3BDE- 10 FB      0650      JSR JUN/IN
3BE0- 20 AE 12  0660            SEC
3BE3- 38        0670 NO/BREAK RTS
3BE4- 60        0680 ;
0690 ;
0700 ;BREAK-VECTOR INITIALISATIE
0710 ;
0720            .BA $202B
202B- 20 A6 3F  0730      JSR BRK/VECT/I
0740 ;
0750 ;
0760            .BA $3FA6
3FA6- A9 CF      0770 BRK/VECT/I LDA #$CF
3FA8- 8D 7E 1A  0780            STA $1A7E
3FAB- A9 14      0790            LDA #$14
3FAD- 8D 7F 1A  0800            STA $1A7F
0810 ;
0820 ;DE BREAK-VECTOR WIJST HIER NAAR DE
0830 ;PRINTER MONITOR "$14CF"
0840 ;
0850 ;

```

6502 SYSTEM SOFTWARE

```

0860 ;DE INITIALISATIE VAN DE POORTEN VOOR
0870 ;DE CASSETTEINTERFACE DIE ZICH BEVIND
0880 ;OP $3FB0 TOT $3FB8 MOET WEGGELATEN
0890 ;WORDEN.
0900 ;WE ZULLEN DIT VERVANGEN DOOR HET
0910 ;VOLGENDE:
0920 ;
0930 ; .BA $3FB0
0940 LDA #$00
0950 NOP
0960 NOP
0970 NOP
0980 NOP
0990 NOP
1000 NOP
1010 RTS
1020 ;
1030 ;
1040 ;DE DOOR MIJ GEBRUIKTE PRINTER IS DE
1050 ;OKI MICROLINE-80
1060 ;DEZE IS AANGESLOTEN OP DE VRIJE 6522.
1070 ;IK MAAK GEBRUIK VAN DE "PRINTER WITH
1080 ;AUTO-LINEFEED" ROUTINE VAN SCHRODER.
1090 ;POORT A WORDT GEBRUIKT VOOR PARALLEL
1100 ;DATA-TRANSPORT. POORT B VOOR DE CON-
1110 ;TROLE-SIGNALLEN. PB0=STROBE
1120 ; PB2=BUSY
1130 ; PB4=SELECT
1140 ;
1150 ;
1160 ; .BA $38D4
1170 JSR HARDCOPY/A
1180 ;
1190 ; .BA $4000
1200 HARDCOPY/A PHA
1210 JSR PRINT
1220 PLA
1230 CMP #$0D
1240 BNE TTTT
1250 LDA #$0A
1260 TTTT RTS
1270 ;
1280 PRINT PHA
1290 LDA #$FF
1300 STA $1803
1310 LDA #$11
1320 STA $1802
1330 LDA $1800
1340 ORA #$01
1350 STA $1800
1360 PLA
1370 STA $1801
1380 LDA $1800
1390 ORA #$10
1400 STA $1800

```

6502 SYSTEM SOFTWARE

```

402B- AD 00 18 1410 WAIT      LDA $1800
402E- 29 04      1420      AND #$04
4030- D0 F9      1430      BNE WAIT
4032- AD 00 18 1440      LDA $1800
4035- 29 FE      1450      AND #$FE
4037- 8D 00 18 1460      STA $1800
403A- 09 01      1470      ORA #$01
403C- 8D 00 18 1480      STA $1800
403F- 2D EF 00 1490      AND $EF
4042- 8D 00 18 1500      STA $1800
4045- 60      1510      RTS
      1520 ;
      1530 ;
      1540 ;WAT BETREFT DE CASSETTE INPUT-
      1550 ;OUTPUT HEB IK ZOVEEL MOGELIJK
      1560 ;GEBRUIK GEMAAKT VAN DE IN DE
      1570 ;JUNIOR AANWEZIGE SUBROUTINES.
      1580 ;
      1590      .BA $3FA3
3FA3- 20 48 40 1600      JSR LOAD
      1610 ;
      1620      .BA $3FD3
3FD3- 20 6D 40 1630      JSR RECORD
      1640 ;
      1650 ;
      1660      .BA $4048
4048- AD 10 01 1670 LOAD      LDA $110
404B- 8D 79 1A 1680      STA $1A79
404E- AD 24 01 1690      LDA $124
4051- 8D 70 1A 1700      STA $1A70
4054- AD 25 01 1710      LDA $125
4057- 8D 71 1A 1720      STA $1A71
405A- AD 26 01 1730      LDA $126
405D- 8D 72 1A 1740      STA $1A72
4060- AD 27 01 1750      LDA $127
4063- 8D 73 1A 1760      STA $1A73
4066- 20 DF 09 1770      JSR $09DF
4069- 20 BC 14 1780      JSR $14BC
406C- 60      1790      RTS
      1800 ;
      1810 ;
      1820 ;ZOWEL VOOR LEZEN VAN, ALS SCHRIJVEN
      1830 ;NAAR CASSETTE GEBEURT, VISUEEL GEZIEN,
      1840 ;ALLES ZOALS BIJ DE JUNIOR. WEL WIL IK
      1850 ;OPMERKEN DAT PER WEGGESCHREVEN FILE MEN
      1860 ;TWEEMAAL IETS ZIET LEZEN OF SCHRIJVEN.
      1870 ;ER WORDT NAMELIJK EERST EEN FILE-
      1880 ;HEADER WEGGESCHREVEN.
      1890 ;
      1900 ;
      1910 RECORD      LDA #$32
406D- A9 32      1920      STA $1A82
406F- 8D 82 1A 1930      STA $1A78
4072- 8D 78 1A 1940      LDA #$7E
4075- A9 7E      1950      STA $1A83
4077- 8D 83 1A

```

6502 SYSTEM SOFTWARE

407A-	A9 7F	1960		LDA #\$7F
407C-	8D 81 1A	1970		STA \$1A81
407F-	A9 00	1980		LDA #\$00
4081-	8D 6E 1A	1990		STA \$1A6E
4084-	8D 6F 1A	2000		STA \$1A6F
4087-	A9 FF	2010	CCCC	LDA #\$FF
4089-	8D 6B 1A	2020		STA \$1A6B
408C-	20 C2 0B	2030	BBBB	JSR \$BC2
408F-	6E 6B 1A	2040		ROR \$1A6B
4092-	AD 6B 1A	2050		LDA \$1A6B
4095-	20 E8 0B	2060		JSR \$BE8
4098-	C9 16	2070		CMP #\$16
409A-	D0 F0	2080		BNE BBBB
409C-	A0 0A	2090		LDY #\$0A
409E-	8C 69 1A	2100		STY \$1A69
40A1-	20 36 0C	2110	DDDD	JSR \$C36
40A4-	20 5D 0C	2120		JSR \$C5D
40A7-	C9 16	2130		CMP #\$16
40A9-	D0 DC	2140		BNE CCCC
40AB-	CE 69 1A	2150		DEC \$1A69
40AE-	D0 F1	2160		BNE DDDD
40B0-	20 36 0C	2170	GGGG	JSR \$C36
40B3-	20 5D 0C	2180		JSR \$C5D
40B6-	C9 2A	2190		CMP #\$2A
40B8-	F0 06	2200		BEQ EEEE
40BA-	C9 16	2210		CMP #\$16
40BC-	F0 F2	2220		BEQ GGGG
40BE-	D0 AD	2230		BNE RECORD
40C0-	20 5D 0C	2240	EEEE	JSR \$C5D
40C3-	20 F3 0B	2250		JSR \$BF3
40C6-	20 F3 0B	2260	RRRR	JSR \$BF3
40C9-	20 4B 0C	2270		JSR \$C4B
40CC-	AD 24 01	2280		LDA \$124
40CF-	8D FA 00	2290		STA \$FA
40D2-	20 F3 0B	2300		JSR \$BF3
40D5-	20 4B 0C	2310		JSR \$C4B
40D8-	AD 25 01	2320		LDA \$125
40DB-	8D FB 00	2330		STA \$FB
40DE-	20 F3 0B	2340	KKKK	JSR \$BF3
40E1-	30 8A	2350		BMI RECORD
40E3-	F0 1D	2360		BEQ HHHH
40E5-	20 4B 0C	2370		JSR \$C4B
40E8-	A0 00	2380		LDY #\$00
40EA-	AA	2390		TAX
40EB-	AD 23 01	2400		LDA \$123
40EE-	F0 03	2410		BEQ IIII
40F0-	8A	2420		TXA
40F1-	91 FA	2430		STA (\$FA),Y
40F3-	8A	2440	IIII	TXA
40F4-	EE FA 00	2450		INC \$FA
40F7-	D0 03	2460		BNE JJJJ
40F9-	EE FB 00	2470		INC \$FB
40FC-	20 64 0C	2480	JJJJ	JSR \$CE4
40FF-	4C DE 40	2490		JMP KKKK
4102-	20 F3 0B	2500	HHHH	JSR \$BF3

6502 SYSTEEM SOFTWARE

```

4105- CD 6E 1A 2510      CMP $1A6E
4108- D0 0C 2520      BNE LLLL
410A- 20 F3 0B 2530      JSR $BF3
410D- CD 6F 1A 2540      CMP $1A6F
4110- D0 04 2550      BNE LLLL
4112- 20 BC 14 2560      JSR $14BC
4115- 60 2570      RTS
411E- 4C 6D 40 2580 LLLL  JMP RECORD
      2590 ;
      2600 ;
      2610 ;CONTROLE VAN DE CASSETTE-MOTOR DOOR
      2620 ;CONTRDL/T, ON EN OFF BEVEL.
      2630 ;
      2640 ;
      2650      .BA $332F
      2660 ;
332F- AD 82 1A 2670      LDA $1A82
3332- 09 20 2680      ORA #$00100000
3334- 8D 82 1A 2690      STA $1A82
3337- 60 2700      RTS
      2710 ;
3338- AD 82 1A 2720      LDA $1A82
333B- 09 40 2730      ORA #$01000000
333D- 8D 82 1A 2740      STA $1A82
3340- 60 2750      RTS
      2760 ;
3341- AD 82 1A 2770      LDA $1A82
3344- 29 DF 2780      AND #$11011111
3346- 8D 82 1A 2790      STA $1A82
3349- 60 2800      RTS
      2810 ;
334A- AD 82 1A 2820      LDA $1A82
334D- 29 BF 2830      AND #$10111111
334F- 8D 82 1A 2840      STA $1A82
3352- 60 2850      RTS
      2860 ;
      2870 ;
      2880      .BA $3C73
      2890 ;
3C73- AD 82 1A 2900      LDA $1A82
3C76- 49 40 2910      EOR #$01000000
3C78- 8D 82 1A 2920      STA $1A82
      2930 ;
3C7B- AD 82 1A 2940      LDA $1A82
3C7E- 49 20 2950      EOR #$00100000
3C80- 8D 82 1A 2960      STA $1A82
      2970 ;
      2980 ;
      2990      .EN

```


6502 PROGRAMMEERTALEN

PATCHES OP BASIC

H.J.C. Otten

In het mei 1981 nummer van Byte verscheen een artikel met de naam : ' Faster BASIC for the Ohio Scientific' .

Daarin wordt beschreven hoe bij gebrek aan de ROR instructie bij de eerste 6502 CPU in de Microsoft Basic voor de Challengers een veel tijd vragende vervangings instructie reeks daarvoor was toegepast. Ook de KIM en TIM versies van Microsoft Basic waren voorzien van deze zogenaamde ROR macro's.

In Byte van september 1981 vond ik een ingezonden brief waar in werd vermeld wat precies de patches op de KIM versie van Microsoft Basic zijn om de veel snellere ROR instructie toe te passen. Hieronder staan deze patches uit Byte herhaald :

Op de volgende lokaties moet komen te staan :

\$37C2 B0 18	BCS \$ 37DC	\$38C3 66 73	ROR \$73
		\$38C5 66 74	ROR \$74
\$37D1 76 02	ROR \$02,X	\$38C7 66 75	ROR \$75
\$37D3 76 03	ROR \$03,X	\$38C9 66 76	ROR \$76
\$37D5 76 04	ROR \$04,X	\$38CB 66 BD	ROR \$BD
\$37D7 68	PLA	\$38CD 98	TYA
\$37D8 6A	ROR A	\$38CE 4A	LSR
\$37D9 C8	INY	\$38CF D0 D6	BNE \$38A7
\$37DA D0 E8	BNE \$37C4	\$38D1 60	RTS
\$37DC 18	CLC		
\$37DD 60	RTS		

(\$37DE tot \$3801 en \$38D2 tot \$3903 zijn nu ongebruikt)

Andere Basic's van Microsoft zoals PET/CBM en Apple hebben deze patch niet nodig maar voor Challenger bezitters is het nuttig als ze een Disc Basic gebruiken het meinummer van Byte te raadplegen. Basic in Rom bezitters kunnen niets doen zonder nieuwe ROM's te kopen.....

Zelf heb ik in de KIM versie de bovenstaande patches ingevoerd en heb een snelheidstoename van ruim 10 % kunnen waarnemen. Vooral veel berekeningen hebben bij deze patch baat omdat bij de floating point berekeningen deze routines veelvuldig worden aangeroepen.

6502 SYSTEEM SOFTWARE

Programma lichtshow voor de JUNIOR en de KIM

K.Kikke
Jonkerlaantje 22
1135 TL Edam

Het programma lichtshow is gebaseerd op een lopend display. Er zijn al inmiddels verschillende versies hiervan gepubliceerd en een ieder weet dus wel hoe het werkt.

Het programma kent twee modes. Door een reeks tekens al dan niet met \$00 te beginnen geeft men aan welke mode gewenst is. De 00-mode moet worden begonnen en afgesloten met \$00. De reeks tekens tussen de nullen worden dan achtereenvolgens gedisplayed als een gewone lopend display. De niet-00-mode geeft aan hoe vaak een teken moet worden gedisplayed. 00 7F betekent bijvoorbeeld 7 maal het streepje (- = 77) laten zien, waarmee 5 geheugenplaatsen worden bespaard. De niet-00-mode bevat achtereenvolgens het aantal en het teken.

De gehele reeks wordt afgesloten met \$FF. Het display loopt dan leeg en via een sprong naar het begin van het programma begint de lichtshow weer van voren af aan. Door de sprong opdracht te vervangen door een RTS kan het programma als subroutine worden gebruikt. Voordat de lichtshow wordt aangeroepen moet het begin adres van het te displayen geheugen worden vermeld in \$000F en \$0010 (MSB)

Als de lichtshow zelfstandig wordt gebruikt moet met de hand \$000F en \$0010 worden gevuld en het programma op \$0200 worden gestart.

Een voorbeeld van een tekenreeks is het volgende, waarbij het start adres op 0300 moet worden gezet :

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0300	06	77	00	6F	5F	00	06	7E	00	7D	7B	00	04	77	00	3F
0310	7E	3F	77	77	3F	7E	00	0A	3F	07	77	03	37	07	36	02
0320	37	04	77	00	6F	3F	7D	6F	3F	76	6E	3F	7D	6F	3F	7D
0330	5D	59	00	03	49	04	6B	06	2B	05	67	00	77	77	3F	39
0340	79	79	7D	7D	1C	54	1C	54	1C	54	1C	54	1C	54	1C	
0350	54	00	08	77	FF											

Noot van de redactie voor KIM gebruikers :

Let U op de KIM/JUN defenitie, de JUNIOR invertteert niet in de hardware , vandaar dat de software voor de KIM moet inverteren om het zelfde effect als dit oorspronkelijk voor de JUNIOR geschreven programma te geven op de KIM displays. EOR KIM/JUN geeft met 00 geen inversie (JUNIOR) en met FF wel (KIM). De lichtkrant display code in het geheugen kan hiermee hetzelfde blijven.

6502 SYSTEEM SOFTWARE

```

0020: ; LICHTSHOW OP LED DISPLAYS
0030: ;
0040: ; AUTEUR : W.KIKKE
0050: ; JONKERLAANTJE 22
0060: ; 1135 TL EDAM
0070: ;
0080: 0200 LICHT ORG $0200 ;
0090: ;
0100: ; ZERO PAGE VARIABELEN
0110: ;
0120: 0200 ZP * $0000 ;
0130: 0200 BDISPL * ZP ; DISPLAY GEBIED VOOR DIGITS
0140: 0200 BD * ZP +01 ;
0150: 0200 LCOUNT * ZP +07 ; DISPLAY SNELHEID
0160: 0200 YSAVE * ZP +08 ; Y REGISTER OPSLAG
0170: 0200 INDEX * ZP +09 ; INDEX IN DISPLAY GEHEUGEN
0180: 0200 NCOUNT * ZP +0C ; AANTAL MAAL NIEUWE DATA
0190: 0200 NDATA * ZP +0E ; NIEUWE DATA
0200: 0200 ADDR1 * ZP +0F ; ADRES VAN TE DISPLAYEN GEHEUGEN
0210: 0200 ADDRH * ZP +10 ;
0220: ;
0230: ; PIA ADRESSEN :
0240: ;
0250: 0200 PIA * $1A80 ; VOOR KIM $1740
0260: 0200 PAD * PIA +00 ; PIA A DATA
0270: 0200 PADD * PIA +01 ; PIA DATA DIRECTION
0280: 0200 PBD * PIA +02 ; PIA B DATA
0290: 0200 PBDD * PIA +03 ; PIA B DATA DIRECTION
0300: ;
0310: ; CONSTATES
0320: ;
0330: 0200 KIMJUN * $00 ; VOOR KIM $FF ;
0340: ;
0350: ; PROGRAMMA LICHTSHOW
0360: ;
0370: 0200 A9 FF START LDAB $FF ; INDEX OP NUL ZETTEN
0380: 0202 85 09 STA INDEX ;
0390: 0204 A2 06 LDAB $06 ; 6 DIGITS UITZETTEN
0400: 0206 95 00 CLEAN STAB BDISPL ; DOOR DISPLAY GEBIED TE VULLEN
0410: 0208 CA DEX ;
0420: 0209 10 FB BPL CLEAN ;
0430: ;
0440: 020B 20 32 02 NNMODE JSR NAD ; HAAL DATA
0450: 020E F0 0F BEQ NMODE ; ALS OO MODE DAN NAAR NMODE
0460: 0210 C9 FF CMPIM $FF ; ALS EINDE DATA DAN
0470: 0212 F0 2D BEQ LDISP ; DISPLAY LATEN LEEGLOPEN
0480: 0214 85 0C STA NCOUNT ; AANTAL MALEN NIEUWE DATA
0490: 0216 20 32 02 JSR NAD ; HAAL DISPLAY DATA
0500: 0219 20 4D 02 JSR DISPL ; VOEG TOE OP DISPLAY
0510: 021C 4C 0B 02 JMP NNMODE ;
0520: ;
0530: 021F 20 32 02 NMODE JSR NAD ; HAAL DATA
0540: 0222 C9 00 CMPIM $00 ; ALS EINDE OO MODE
0550: 0224 F0 E5 BEQ NNMODE ; DAN NAAR NNMODE
0560: 0226 48 PHA ;
0570: 0227 A9 01 LDAB $01 ; EENMAAL NIEUWE DATA
0580: 0229 85 0C STA NCOUNT ;
0590: 022B 68 PLA ;

```

6502 SYSTEM SOFTWARE

```

0600: 0220 20 40 02      JSR  DISPL ; VOEG DTOE AAN DISPLAY
0610: 022F 4C 1F 02      JMP  NMODE ;
0620:                      ;
0630:                      ; ROUTINE OM VOLGENDE ADRES EN DATA TE HALEN
0640:                      ;
0650: 0232 E6 09      NAD  INC  INDEX ; VOLGENDE DATABYTE
0660: 0234 A4 09      LDY  INDEX ;
0670: 0236 B1 0F      LDRIY ADDR1 ; HAAL DATABYTE
0680: 0238 48          PHA          ;
0690: 0239 C0 FF      CPYIM $FF ; VOLGENDE PAGINA ?
0700: 023B D0 02      BNE  NP      ;
0710: 023D E6 10      INC  ADDR1 ;
0720: 023F 68      NP  PLA          ;
0730: 0240 60          RTS          ;
0740:                      ;
0750:                      ; DISPLAY LEEG LATEN LOPEN
0760:                      ;
0770: 0241 A9 06      LDISP LDRIIM $06 ; LAAT DISPLAY LEEG
0780: 0243 85 0C      STA  NCOUNT ; LOPEN DOOR
0790: 0245 A9 FF      LDRIIM $FF ; ZES MAAL UIT
0800: 0247 20 40 02      JSR  DISPL ; TOE TE VOEGEN
0810: 024A 4C 00 02      JMP  START ;
0820:                      ;
0830:                      ; ROUTINE DISPLAY
0840:                      ;
0850: 024D 85 0E      DISPL STA  NDATA ; VOEG NIEUWE DATA TOE
0860: 024F A5 0E      DIS  LDA  NDATA ; AAN DISPLAY GEBIED
0870: 0251 85 00      STA  BDISPL ;
0880: 0253 A9 10      LDRIIM $10 ; DISPLAY SNELHEID
0890: 0255 85 07      STA  LCOUNT ;
0900: 0257 A9 7F      LOOP0 LDRIIM $7F ;
0910: 0259 8D 81 1A    STA  PADD ;
0920: 025C A2 08      LDYIM $08 ;
0930: 025E A0 06      LDYIM $06 ; ZES DISPLAYS
0940: 0260 84 08      LOOP1 STY  YSAVE ;
0950: 0262 B9 00 00    LDRIY BDISPL ;
0960: 0265 0E 02 1A    STX  PBD ;
0970: 0268 49 00      EORIM KIMJUN ; KIM/JUNIOR INVERSIE
0980: 026A 8D 80 1A    STA  PAD ; ZET OP DISPLAY
0990: 026D A0 FF      LDYIM $FF ; WACHT EVEN MET DIT DIGIT
1000: 026F 88      WAIT  DEY          ;
1010: 0270 D0 FD      BNE  WAIT      ;
1020: 0272 E8      INX          ; VOLGENDE DIGIT
1030: 0273 E8      INX          ;
1040: 0274 A4 08      LDY  YSAVE ;
1050: 0276 88      DEY          ; VOLGENDE GEHEUGEN
1060: 0277 10 E7      BPL  LOOP1 ;
1070: 0279 8C 81 1A    STY  PADD ; ZET DIGIT AF
1080: 027C C6 07      DEC  LCOUNT ; VOLGENDE LOOP
1090: 027E D0 D7      BNE  LOOP0 ;
1100:                      ;
1110: 0280 A2 05      LDYIM $05 ;
1120: 0282 B5 00      MOVE  LDRIYX BDISPL ; MOVE DISPLAY GEBIED
1130: 0284 95 01      STARIY BD ;
1140: 0286 CA      DEX          ;
1150: 0287 10 F9      BPL  MOVE ; EEN DIGIT VOORUIT
1160: 0289 C6 0C      DEC  NCOUNT ; AANTAL MALEN NIEUWE DATA ?
1170: 028B D0 C2      BNE  DIS      ;
1180: 028D A0 6A      RTS          ;

```

6502 SYSTEEM SOFTWARE

4 KOLOM printer programma print 4 kolommen met 16 regels op het display van de Elekterminal. Het beginadres van het te displayen stuk moet op \$0000 (LSB) en \$0001 (MSB) worden gezet.

Hetprogramma print dan adres , opcode en eventuele operanden.

Andere terminals zijn ook bruikbaar mits voorzien van cursor besturing.

```

0020:          ; PROGRAMMA 4 KOLOM PRINTER
0030:          ;
0040:          ; AUTEUR : A.S.HANKEL
0050:          ; W.KLOOSSSTRAAT 32.
0060:          ; 7606 BB ALMELO
0070:          ;
0080: 0200      VKPR   ORG   $0200
0090:          ;
0100:          ; ZERO PAGE LOKATIES
0110:          ;
0120: 0200      SAL   *    $0000 ; STARTADRES      TE PRINTEN STUK
0130: 0200      SAH   *    $0001 ;
0140: 0200      UTEL  *    $0002 ; VERTIKALE POSITIE TELLER
0150: 0200      HTEL  *    $0003 ; HORIZONTALE POSITIE TELLER
0160: 0200      BYTES *    $00F6 ; UITKOMST INSTRUCTIE LENGTE NETING LENACC
0170: 0200      POINTL *   $00FA ; POINTER
0180: 0200      POINTH *   $00FB ;
0190:          ;
0200:          ; JUNIOR MONITOR ROUTINES
0210:          ;
0220: 0200      MONITO *   $1010 ;
0230: 0200      CRLF  *   $11E8 ;
0240: 0200      PRSP  *   $11F3 ; PRINT (SPATIE)
0250: 0200      PRBYTE *   $128F ; PRINT BYTE ALS TWEE ASCII KARAKTERS
0260: 0200      PROHA *   $1334 ; PRINT ASCII KARAKTER IN A
0270: 0200      LENACC *   $1E60 ; MEET INSTRUCTIE BYTE LENGTE, NR IN $00F6
0280:          ; PIA LOKATIES
0290:          ;
0300: 0200      PBD   *   $1A83 ; PIA B DATA REGISTER
0310: 0200      TIMER *   $1AF7 ; TIMER 1024/
0320:          ;
0330:          ; CURSOR BESTURINGS KARAKTER ELEKTUUR TERMINAL
0340:          ;
0350: 0200      CR    *    $0D ;
0360: 0200      RIGHTC *   $09 ;
0370: 0200      UP    *    $0B ;
0380: 0200      FF    *    $0C ; FORM FEED
0390:          ;
0400:          ; PROGRAMMA :
0410:          ;
0420: 0200 A5 00      START LDA   SAL   ; ADRES := SARTADRES
0430: 0202 85 FA      STA   POINTL ;
0440: 0204 A5 01      LDA   SAH   ;
0450: 0206 85 FB      STA   POINTH ;
0460: 0208 A9 00      LDAIM $00 ; HOR POS := 0
0470: 020A 85 03      STA   HTEL  ;
0480: 020C A9 01      LDAIM $01 ;
0490: 020E 8D 03 1A  STA   PBD   ; INITIALIZE I/O

```

6502 SYSTEM SOFTWARE

```

0500: 0211 A9 0C      LDAIM FF      ; CLEAR SCREEN
0510: 0213 20 34 13   JSR  PRCHA ;
0520: 0216 A9 80      LDAIM $80    ; DELAY 132 MS
0530: 0218 80 F7 1A   STA  TIMER ;
0540: 021B 2C F7 1A   CHECK BIT  TIMER ;
0550: 021E 10 FB      BPL  CHECK ;
0560: 0220 A9 00      PRINTA LDAIM $00 ; WHILE HTEL <> 64 DO (* MAIN LOOP *)
0570: 0222 85 03      STA  HTEL ; UTEL := 0
0580: 0224 A6 03      LDX  HTEL ; IF HOR POS <> 0 THEN
0590: 0226 F0 08      BEQ  HIBYTE ; WHILE CURSOR POS <> HTEL DO
0600: 0228 A9 09      RIGHT LDAIM $09 ; MOVE CURSOR RIGHT
0610: 022A 20 34 13   JSR  PRCHA ; CURSOR POS := CURSOR POS + 1
0620: 022D CA        DEX ; ENDWHILE
0630: 022E D0 F8      BNE  RIGHT ;
0640: 0230 A5 FB      HIBYTE LDA  POINTH ; PRINT (POINTER)
0650: 0232 20 8F 12   JSR  PRBYTE ;
0660: 0235 A5 FA      LDA  POINTL ;
0670: 0237 20 8F 12   JSR  PRBYTE ;
0680: 023A 20 F3 11   JSR  PRSP ; PRINT (SPATIE )
0690: 023D A0 00      LDYIM $00 ; INSTR := LENGTH(INSTRUCTION)
0700: 023F B1 FA      LDAIV POINTL ;
0710: 0241 20 60 1E   JSR  LENACC ;
0720: 0244 A6 F6      LDX  BYTES ;
0730: 0246 A0 00      LDYIM $00 ;
0740: 0248 B1 FA      PRDATA LDAIV POINTL ; WHILE INSTR > 0 DO
0750: 024A 20 8F 12   JSR  PRBYTE ; PRINT(POINTERT.DATA)
0760: 024D 20 F3 11   JSR  PRSP ; PRINT (SPATIE)
0770: 0250 E6 FA      INC  POINTL ; POINTER := POINTER + 1
0780: 0252 D0 02      BNE  DOWNX ;
0790: 0254 E6 FB      INC  POINTH ;
0800: 0256 CA        DOWNX DEX ; INSTR := INSTR - 1
0810: 0257 D0 EF      BNE  PRDATA ;
0820: 0259 E6 02      INC  UTEL ; VERT POS := VERT POS + 1
0830: 025B A5 02      LDA  UTEL ; IF VERT POS < 16 THEN
0840: 025D C9 10      CMPIM $10 ;
0850: 025F F0 05      BEQ  PLUSHO ; PRINT CARR RETURN + LINEFEED
0860: 0261 20 E8 11   JSR  CRLF ;
0870: 0264 F0 C2      BEQ  RIGHT ; MOVE TO COLUMN
0880: 0266 18        PLUSHO CLC ;
0890: 0267 A5 03      LDA  HTEL ; HOR POS := HOR POS + 1
0900: 0269 69 0F      ADCIM $0F ;
0910: 026B 85 03      STA  HTEL ;
0920: 026D C9 3C      CMPIM $3C ; IF HOR POS = 64 THEN EXIT
0930: 026F D0 03      BNE  BACK ;
0940: 0271 4C 10 1C   JMP  MONITO ;
0950: 0274 A9 0D      BACK LDAIM CR ; ( SEND CURSOR HOME )
0960: 0276 20 34 13   JSR  PRCHA ; PRINT CARR RETURN
0970: 0279 A2 0F      LDXIM $0F ; COUNT := 16
0980: 027B A9 0B      UPB LDAIM UP ; WHILE COUNT > 0 DO
0990: 027D 20 34 13   JSR  PRCHA ; PRINT (CURSOR UP)
1000: 0280 CA        DEX ; COUNT := COUNT - 1
1010: 0281 D0 F8      BNE  UPB ; END WHILE
1020: 0283 F0 9B      BEQ  PRINTA ; ENDWHILE ( MAINLOOP )

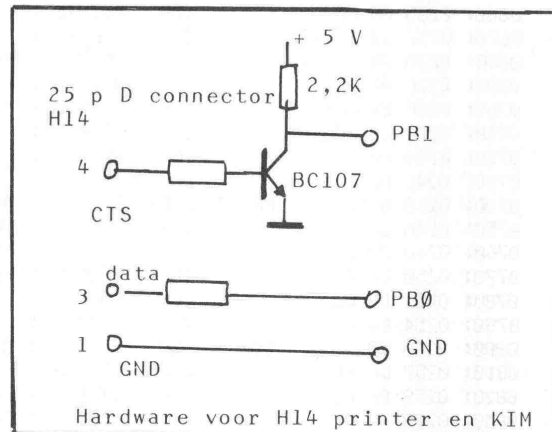
```

6502 SYSTEEM SOFTWARE

```

0010: *****
0020: *
0030: PRINTER OUTPUT H14 MET HANDSHAKE
0040: *
0050: HJC OTTEN EN PGJ DE BEER
0060: *
0070: UITGEBREIDE VERSIE VAN DE AL EERDER VERSCHENEN RS232
0080: INTERFACE ROUTINE
0090: MET HANDSHAKE , OPTIMALISATIE NAAR 4800 HZ
0100: EN REGISTER BEVEILIGING
0110: OOK OPGENOMEN EEN PARALLEL KEYBOARD DRIVER PIA A
0120: STROBE IN BIT 7
0130: *
0140: VERBINDEN VAN DEZE DEVICE DRIVERS MET MICRO ADE
0150: *****
0160: ;
0170: 0300 PRINT ORG $0300
0180: ;
0190: ZEROPAGE
0200: ;
0210: 0300 CHAR * $00FE
0220: ;
0230: PIA ADRES
0240: ;
0250: 0300 PIAAD * $1700
0260: 0300 PIAADD * $1701
0270: 0300 PIABD * $1702
0280: 0300 PIABDD * $1703
0290: ;
0300: MONITOR ADRESSEN
0310: ;
0320: 0300 OUTCH * $1EA0
0330: 0300 GETCH * $1E5A
0340: ;
0350: MICRO-ADE ADRESSEN
0360: ;
0370: 0300 MAWS * $2031
0380: ;
0390: PRINTER ROUTINE
0400: ;
0410: 0300 00 UITPR PHP BEWAAR REGISTERS
0420: 0301 40 PHA SAVE REGISTERS OP STACK
0430: 0302 90 TVR
0440: 0303 40 PHA
0450: 0304 8A TXR
0460: 0305 40 PHA
0470: 0306 A5 FE LDA CHAR
0480: 0308 40 PHA
0490: 0309 BA TSX
0500: 030A BD 04 01 LDARX $0104
0510: 030D 85 FE STA CHAR
0520: 030F AD 02 17 BUFFV LDA PIABD PRINTER RTS
0530: 0312 29 02 ANDIM $02 READY BIT 2
0540: 0314 F0 F9 BEQ BUFFV WACHT TOT PRINTER KLAAR
0550: 0316 AD 02 17 LDA PIABD
0560: 0319 09 01 ORAIM $01 STARTBIT

```



6502 SYSTEM SOFTWARE

```

0570: 031B 8D 02 17 STA PIABD
0580: 031E 20 56 03 JSR TDELAY
0590: 0321 A2 08 LDXIM #08 8 BITS VERSTUREN
0600: 0323 10 00 BPL VOLGB DUMMY
0610: 0325 AD 02 17 VOLGB LDA PIABD
0620: 0328 46 FE LSR CHAR LSB IN CARRY
0630: 032A B0 05 BCS EEN
0640: 032C 09 01 ORAIM #01 BIT0IS 0
0650: 032E 90 03 BCC BUIT
0660: 0330 EA NOP TIMING
0670: 0331 29 FE EEN ANDIM #FE BIT IS 1
0680: 0333 8D 02 17 BUIT STA PIABD
0690: 0336 20 56 03 JSR TDELAY
0700: 0339 CA DEX VOLGENDE BIT
0710: 033A D0 E9 BNE VOLGB
0720: 033C AD 02 17 LDA PIABD
0730: 033F 29 FE ANDIM #FE STOPBIT
0740: 0341 EA NOP
0750: 0342 EA NOP
0760: 0343 EA NOP TIMING
0770: 0344 10 00 BPL DUMM
0780: 0346 8D 02 17 DUMM STA PIABD
0790: 0349 20 56 03 JSR TDELAY
0800: 034C 68 PLA
0810: 034D 85 FE STA CHAR
0820: 034F 68 PLA
0830: 0350 AA TAX
0840: 0351 68 PLA
0850: 0352 A8 TAY
0860: 0353 68 PLA
0870: 0354 28 PLP
0880: 0355 60 RTS
0890: ;
0900: DELAY ROUTINE 4800 BAUD
0910: ;
0920: 0356 A0 22 TDELAY LDYIM #22
0930: 0358 88 DLOOP DEY
0940: 0359 D0 FD BNE DLOOP
0950: 035B EA NOP
0960: 035C 60 RTS
0970: ;
0980: *****
0990: ;
1000: PARALLEL INPUT KEYBOARD
1010: ;
1020: *****
1030: ;
1040: 035D 2C 00 17 INKEYB BIT PIABD
1050: 0360 10 FB BPL INKEYB BIT 7 IS STROBE
1060: 0362 2C 00 17 STROBE BIT PIABD WACHT OP EINDE STARTBIT
1070: 0365 30 FB BMI STROBE
1080: 0367 AD 00 17 LDA PIABD
1090: 036A 48 PHA
1100: 036B 20 A0 1E JSR OUTCH ECHO CHARACTER
1110: 036E 68 PLA
1120: 036F 60 RTS

```


6502 SYSTEM SOFTWARE

```

1130:      ;
1140:      INPUT AAN MICRO ADE HANGEN
1150:      ;
1160: 0370 48      PKB   PHA
1170: 0371 A9 00      LDAIM $00
1180: 0373 8D 01 17    STA   PIAADD
1190: 0376 A9 5D      LDAIM INKEYB
1200: 0378 8D 9E 2E    STA   $2E9E MICRO ADE INPUT VECTOR
1210: 037B A9 03      LDAIM INKEYB /
1220: 037D 8D 9F 2E    STA   $2E9F
1230: 0380 68      PLA
1240: 0381 4C 31 20    JMP   MAWS
1250:      ;
1260:      SERIE KEYBOARD VIA KIM TTY AAN MICRO ADE HANGEN
1270:      ;
1280: 0384 A9 5A      SKB   LDAIM GETCH
1290: 0386 8D 9E 2E    STA   $2E9E
1300: 0389 A9 1E      LDAIM GETCH /
1310: 038B 8D 9F 2E    STA   $2E9F
1320: 038E 4C 31 20    JMP   MAWS
1330:      ;
1340:      PRINTER PARALLEL VIDEO AAN MICRO ADE HANGEN
1350:      ;
1360: 0391 A9 03      PRAAN LDAIM PIAADD
1370: 0393 09 01      ORAIM $01
1380: 0395 29 7D      ANDIM $7D
1390: 0397 8D 03 17    STA   PIAADD BIT0 UITGANG, BIT1 INGANG
1400: 039A A9 8B      LDAIM PRENVI
1410: 039C 8D A1 2E    STA   $2EA1
1420: 039F A9 03      LDAIM PRENVI /
1430: 03A1 8D A2 2E    STA   $2EA2 VERVANG MICRO ADE OUTPUT VECTOR
1440: 03A4 A9 00      LDAIM $00
1450: 03A6 20 00 03    JSR   UITPR
1460: 03A9 A9 1B      LDAIM $1B ESCAPE NAAR PRINTER
1470: 03AB 20 00 03    JSR   UITPR
1480: 03AE A9 75      LDAIM $75
1490: 03B0 20 00 03    JSR   UITPR
1500: 03B3 A9 10      LDAIM $10
1510: 03B5 20 00 03    JSR   UITPR
1520: 03B8 4C 31 20    JMP   MAWS
1530:      ;
1540:      PRINTER PARALLEL VIDEO OUTPUT
1550:      ;
1560: 03BB 48      PRENVI PHA
1570: 03BC 20 A0 1E    JSR   OUTCH
1580: 03BF 68      PLA
1590: 03C0 20 00 03    JSR   UITPR
1600: 03C3 60      RTS
1610:      ;
1620:      ALLEEN VIDEO OUTPUT M A
1630:      ;
1640: 03C4 A9 A0      PRUIT LDAIM OUTCH
1650: 03C6 8D A1 2E    STA   $2EA1 MICRO ADE OUTPUT VECTOR NAAR KIM TTY
1660: 03C9 A9 1E      LDAIM OUTCH /
1670: 03CB 8D A2 2E    STA   $2EA2
1680: 03CE 4C 31 20    JMP   $2031

```

6502 SYSTEEM SOFTWARE

Programma BREAK voor Junior en KIM

Frans Mepschen
Eikenlaan 4
9636 CV Zuidbroek (GR.)
tel . 05985 - 2124

Doel van het programma is het inlassen van enkele break opdrachten in een programma; op die brekpoints zorgt BREAK voor een automatische register uitlezing en het vervangen van de break code door een code uit de tabel.

Break start automatisch na het lezen van een break code via de IRQ vector die van te voren moet worden geladen met 00 en 01 (LSB en MSB)
Voor de KIM is de IRQ vector op \$17FE en \$17FF te vinden , bij de JUNIOR op \$1A7E en \$1A7F .

Na het aanroepen van BREAK verschijnen op de displays de inhoud van de registers A, X en Y van links naar rechts respectievelijk.
Indrukken van de + toets laat de registers S en P zien op de twee linker en twee midden displays , de rechtse displays zijn gedoofd (JUNIOR) en 00 (KIM).
Weer op de + toets drukken laat weer A, X en Y zien.
Door de DATA toets in te drukken wordt BREAK verlaten, de break code wordt vervangen door de waarde uit een tabel , op de displays verschijnt het adres van de lokatie waar de BREAK code stond en onder de PC toets staat het begin adres van het programma.

Voordat BREAK kan worden gebruikt moeten de volgende adressen worden gevuld :

IRQ vector (zie boven)

\$00D0 start adres te testen programma
\$00D1 " " MSB
\$00D2 Counter, laden met 00

Tabel met vervangende codes voor break opdrachten :
\$00D3 Eerste vervangende code voor eerste break opdracht

⋮
\$00DD Laatste " "

VOORBEELD :

Hoofdprogramma :

0200	A9 55	LDA \$55
0202	A2 00	LDX \$00
0204	A0 FF	LDY \$00
0206	0A	ASL
0207	CA	DEX
0208	C8	INY
0209	4C 1D 1C	RESET

De code op de adressen 0206
vervangen door BRK (00)
00D0 = 00 , 00D1 = 02 , 00D2 = 00
00D3 = 0A

Start programma op 0200 G, geeft display
55 00 FF (resp A, X en Y.
+ toets geeft display B4 FD -- (S en SP)
DATA toets geeft display 0206 0A.
PC toets geeft display 0200 A9

6502 SYSTEEM SOFTWARE

```

0030:          ; PROGRAMMA BREAK
0040:          ;
0050: 0200      BREAK ORG    $0200
0060:          ;
0070:          ; ZERO PAGE GEBRUIK :
0080:          ;
0090: 0200      STHL *      $00D0 ; STARTADRES
0100: 0200      STHH *      $00D1 ;
0110: 0200      COUNT *     $00D2 ; TABEL WIJZER
0120: 0200      TABEL *     $00D3 ; TABEL MET CODES VAN BREAKPOINTS
0130: 0200      MEML *      $00DE ; BREAK ADDRESS
0140: 0200      MEMH *      $00DF ;
0150: 0200      SAVER *      $00E0 ; REGISTER SAVE AREA
0160: 0200      PCL *        $00EF ;
0170: 0200      PCH *        $00F0 ;
0180: 0200      SAVEP *      $00F1 ;
0190: 0200      SAVES *      $00F2 ;
0200: 0200      LAST *       $00F3 ;
0210: 0200      SAVEY *      $00F4 ;
0220: 0200      SAVEX *      $00F5 ;
0230: 0200      NRDIG *      $00F6 ;
0240: 0200      INH *       $00F9 ; DISPLAY AREA
0250: 0200      POINTL *     $00FA ;
0260: 0200      POINTH *     $00FB ;
0270:          ;
0280:          ; JUNIOR MONITOR ROUTINES :
0290:          ;
0300: 0200      SCANDS *      $1D8E ; TOON DISPLAY AREA
0310: 0200      GETKEY *     $1DF9 ; HAAL TOETS
0320: 0200      RESET *     $1C10 ; RESET VAN MONITOR
0330:          ;
0340:          ; VOOR DE KIM VERSIE WORDEN DEZE LOKATIES ;
0350:          ;
0360: 0200      SCANDS *      $1F1F ;
0370: 0200      GETKEY *     $1F6A ;
0380: 0200      RESET *     $1C22 ;
0390:          ;
0400:          ; CONSTANTES :
0410:          ;
0420: 0200      PLUS *       $12   ; PLUS KEY CODE
0430: 0200      DATA *      $11   ; DATA KEY CODE
0440:          ;
0450:          ; BREAK ROUTINE
0460:          ;
0470: 0200 85 E0   BREAKR STA  SAVER ; SAVE REGISTERS
0480: 0202 84 F4       STV  SAVEY ;
0490: 0204 86 F5       STX  SAVEX ;
0500: 0206 68         PLA      ;
0510: 0207 85 F1       STA  SAVEP ;
0520: 0209 BA        TSX      ;
0530: 020A 86 F2       STX  SAVES ;
0540:          ;
0550:          ; DISPLAY MAIN LOOP
0560:          ;
0570: 020C A5 E0   DISP  LDA  SAVER ; SHOW A,X,Y ON DISPLAY
0580: 020E 85 FB       STA  POINTH ;
0590: 0210 A5 F5      LDA  SAVEX ; PLACE A,X,Y IN DISPLAY AREA

```

6502 SYSTEM SOFTWARE

```

0600: 0212 85 FA      STA POINTL ;
0610: 0214 A5 F4      LDA SAVEV ;
0620: 0216 85 F9      STA INH ;
0630: 0218 A9 03      LDAIM #03 ; ZINLOOS VOOR KIM MAAR ONSCHULDIG
0640: 021A 85 F6      STA NRDIG ; ( 3 DISPLAYS )
0650: 021C 20 8E 1D DISPA JSR SCANDS ; SHOW DISPLAY AREA
0660: 021F 20 F9 1D DISPA JSR GETKEY ; GET KEY
0670: 0222 C5 F3      CMP LAST ; WHILE OLDKEY <> NEWKEY DO
0680: 0224 F0 F6      BEQ DISPA ; GETKEY
0690: 0226 85 F3      STA LAST ;
0700: 0228 C9 12      CMPIM PLUS ; IF KEY = PLUS THEN SHOW S AND P
0710: 022A F0 07      BEQ DISPB ;
0720: 022C C9 11      CMPIM DATA ; IF KEY = DATA THEN RETURN FROM BREAK
0730: 022E F0 26      BEQ RETURN ;
0740: 0230 4C 1C 02 DISPB JMP DISPA ; IF OTHER KEY KEEP WAITING
0750: 0233 A5 F1      LDA SAVEP ; SHOW S AND P
0760: 0235 85 FB      STA POINTH ;
0770: 0237 A5 F2      LDA SAVES ; PLACE S AND P IN DISPLAY AREA
0780: 0239 85 FA      STA POINTL ;
0790: 023B A9 02      LDAIM #02 ; ( 2 DISPLAYS )
0800: 023D 85 F6      STA NRDIG ; ZINLOOS VOOR KIM , VERVANG DOOR : 85 F9
0810: 023F 20 8E 1D DISPC JSR SCANDS ; SHOW DISPLAY AREA
0820: 0242 20 F9 1D DISPC JSR GETKEY ; GETKEY
0830: 0245 C5 F3      CMP LAST ; WHILE OLDKEY = NEWKEY DO
0840: 0247 F0 F6      BEQ DISPC ; GETKEY
0850: 0249 85 F3      STA LAST ;
0860: 024B C9 12      CMPIM PLUS ; IF KEY = PLUS THEN SHOW A,X,Y
0870: 024D F0 0D      BEQ DISP ;
0880: 024F C9 11      CMPIM DATA ; IF KEY = DATA THEN RETURN FROM BREAK
0890: 0251 F0 03      BEQ RETURN ;
0900: 0253 4C 3F 02 DISPC JMP DISPC ; IF OTHER KEY KEEP WAITING
0910: ;
0920: ; RETURN FROM BREAK ROUTINE
0930: ; PC KEY SET TO BEGIN ADDRESS
0940: ; REPLACE BREAK CODE WITH REAL CODE FROM TABEL
0950: ;
0960: 0256 68          RETURN PLA ; CALCULATE ADDRESS OF BREAK
0970: 0257 38          SEC ;
0980: 0258 E9 02      SBCIM #02 ;
0990: 025A 85 DE      STA MENL ;
1000: 025C 68          PLA ; SAVE IN MEM
1010: 025D E9 00      SBCIM #00 ;
1020: 025F 85 DF      STA MEMH ;
1030: 0261 A5 D0      LDA STHL ;
1040: 0263 85 EF      STA PCL ; SET PC KEY TO BEGIN ADDRESS
1050: 0265 A5 D1      LDA STHH ;
1060: 0267 85 F0      STA PCH ;
1070: 0269 A6 D2      LDX COUNT ; BREAK COUNT
1080: 026B B5 D3      LDZX TABEL ; CODE
1090: 026D E8          INX ;
1100: 026E 86 D2      STX COUNT ; NEXT BREAKPOINT
1110: 0270 A0 00      LDYIM #00 ;
1120: 0272 91 DE      STAIY MENL ; RESTORE REAL CODE AT BREAKPOINT
1130: 0274 A5 DE      LDA MENL ; SET DISPLAY OF BREAK ADDRESS
1140: 0276 85 FA      STA POINTL ;
1150: 0278 A5 DF      LDA MEMH ;
1160: 027A 85 FB      STA POINTH ;
1170: 027C 4C 10 1C JMP RESET ; BACK TO MONITOR

```

6502 SYSTEEM SOFTWARE

FRANS SMEEHUIJZEN
LIPPEDAL 19
2904 CL CAPELLE AAN DEN IJSSEL

PROGRAMMA: HEX-DUMP T.B.V.
DE KIM-1

AANGEZIEN IK VOOR HET WERKEN OP MIJN KIM-1, AANGESLOTEN OP EEN VDU (80X24), BEHOEFTE HAD AAN EEN MEER OVERZICHTELIJKE HEXADECIMALE DUMP DAN DE KIM-1 LEVERT VIA DE Q-TOETS. HEB IK HET VOLGENDE PROGRAMMA GEMAAKT. ZOALS HET VOORBEELD WEERGEeft WORDT MET EEN KOPREGEL GEWERKT WELKE DE MOGELIJKHEID BIEDT EENVOUDIGER DE JUISTE GEHEUGENLOKATIE TE BEPALEN. VERDER ZIJN DE ;18 EN DE CHECKSUM ER AFGELATEN EN ZIJN ER TUSSEN DE VERSCHILLENDE CODES SPATIES AANGEBRACHT. OM TUSSENTIJD S DE GEDISPLAY-DE DUMP TE KUNNEN BEKIJKEN, STOPT HET PROGRAMMA WANNEER HET TE DUMPEN ADRES DE XXFF OVERSCHRIJDT. HIERNA KAN DAN EEN KEUZE GEMAAKT WORDEN OM OF DOOR TE GAAN OF OM OPNIEUW TE BEGINNEN VANAF EEN ANDER IN TE TOETSEN ADRES. E.E.A. VOLDOET MISSCHIE N AAN DE BEHOEFTE VAN ANDERE KIM-1 (JUNIOR) GEBRUIKERS.

N.B. DE DOOR MIJ GEBRUIKTE ASCII-CODE 1A (CLEAR SCREEN) IS UITERAARD VOOR IEDERE VDU VERSCHILLENDE.

```
29F8 A2 00      LDX #$00      : PRINT "HEX-DUMP V01."
29FA BD 96 2A    LDA 2A96,X
29FD 20 A0 1E    JSR 1EA0
2A00 E8          INX
2A01 E0 10      CPX #$10
2A03 D0 F5      BNE 29FA
2A05 20 70 FA    JSR FA70      : PRINT "STARTADRES: "
2A08 20 9D 1F    JSR 1F9D      : INTOETSEN STARTADRES EN ZETTEN
2A0B 85 01      STA 01        : OP ADRESSEN 0000 EN 0001
2A0D 20 9D 1F    JSR 1F9D
2A10 85 00      STA 00
2A12 20 50 2A    JSR 2A50      : PRINT KOPREGEL
2A15 A5 01      LDA 01        : PRINT STARTADRES
2A17 20 3B 1E    JSR 1E3B
2A1A A5 00      LDA 00
2A1C 20 3B 1E    JSR 1E3B
2A1F 20 9E 1E    JSR 1E9E
2A22 A0 00      LDY #$00      : PRINT DE INHOUD VAN DE OP
2A24 B1 00      LDA (00),Y    : DE ADRESSEN 0000 EN 0001 GEVONDEN
2A26 84 03      STY 03        : ADRESSEN
2A28 20 3B 1E    JSR 1E3B
2A2B 20 9E 1E    JSR 1E9E
2A2E A4 03      LDY 03
```

6502 SYSTEEM SOFTWARE

```

2A30 C8      INY
2A31 C0 10   CPY ##10      : KONTROLEER OF ER AL 16 GEHEUGEN-
2A33 D0 EF   BNE 2A24      : LOKATIES ZIJN AFGEDRUKT
2A35 20 2F 1E JSR 1E2F
2A38 A5 00   LDA 00      : ZO JA, VERHOOG DE INHOUD VAN ADRES
2A3A 69 10   ADC ##10     : 0000 MET ##10
2A3C 85 00   STA 00
2A3E B0 03   BCS 2A43     : INDIEN DE INHOUD VAN ADRES 0000
2A40 4C 15 2A JMP 2A15     : ) FF DAN DE INHOUD VAN ADRES 0001
2A43 E6 01   INC 01      : MET 1 VERHOGEN EN STOPPEN MET PRINTEN
2A45 20 5A 1E JSR 1E5A     : HET PRINTEN DOORSTARTEN INDIEN ER
2A48 C9 0D   CMP ##0D     : EEN RETURN WORDT INGETOETST EN
2A4A F0 C9   BEQ 2A15     : OPNIEUW BEGINNEN INDIEN ER EEN SPACE
2A4C C9 20   CMP ##20     : WORDT INGETOETST
2A4E F0 A8   BEQ 29F8
2A50 A2 00   LDX ##00     : SUBROUTINE VOOR HET PRINTEN VAN DE KOP-
2A52 BD 5E 2A LDA 2A5E, X  : REGEL
2A55 20 A0 1E JSR 1EA0
2A58 E8      INX
2A59 E0 38   CPX ##38
2A5B D0 F5   BNE 2A52
2A5D 60      RTS
2A5E 1A 41 44 52 20 20 30 : TEKST :
      30 20 30 31 20 30 32 : ADR 00 01 02 03 04 05 06 07 08 09 0A 0B 0C
      20 30 33 20 30 34 20 :      0D 0E 0F CR LF LF
      30 35 20 30 36 20 30 :
      37 20 30 38 20 30 39 :
      20 30 41 20 30 42 20 :
      30 43 20 30 44 20 30 :
      45 20 30 46 0D 0A 0A :
2A96 1A 48 45 58 2D 44 55 : TEKST HEX-DUMP V01.
      4D 50 20 56 30 31 2E :
      0D 0A

```

HEX-DUMP V01.
STARTADRES:

AFDRUK-VOORBEELD PROGRAMMA HEX-DUMP

```

ADR 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
29F8 A2 00 BD 96 2A 20 A0 1E E8 E0 10 D0 F5 20 70 FA
2A08 20 9D 1F 85 01 20 9D 1F 85 00 20 50 2A A5 01 20
2A18 3B 1E A5 00 20 3B 1E 20 9E 1E A0 00 B1 00 84 03
2A28 20 3B 1E 20 9E 1E A4 03 C8 C0 10 D0 EF 20 2F 1E
2A38 A5 00 69 10 85 00 B0 03 4C 15 2A E6 01 20 5A 1E
2A48 C9 0D F0 C9 C9 20 F0 A8 A2 00 BD 5E 2A 20 A0 1E
2A58 E8 E0 38 D0 F5 60 1A 41 44 52 20 20 30 30 20 30
2A68 31 20 30 32 20 30 33 20 30 34 20 30 35 20 30 36
2A78 20 30 37 20 30 38 20 30 39 20 30 41 20 30 42 20
2A88 30 43 20 30 44 20 30 45 20 30 46 0D 0A 0A 1A 48
2A98 45 58 2D 44 55 4D 50 20 56 30 31 2E 0D 0A 03 82
2AA8 76 08 FF 02 FE 40 FE 0C 7A 43 B6 8A 51 A4 35 0C
2AB8 DC 80 3F 00 F7 20 FF 00 D6 82 EF 00 EF 02 B7 00
2AC8 95 26 7F 44 B9 22 FF 08 65 28 4B 24 DF 92 F7 08
2AD8 B3 01 FB 07 FE 80 DB BA 3A 08 FF 84 BF F8 BF 52
2AE8 FB 02 FB 1A FF 24 FF 04 7B FA DB 32 76 26 BF 8B
2AF8 9D 00 7D 00 FB 08 8B 20 7F 02 5F 6A 2E A0 FD 8C

```

6502 VAN DE REDACTIE

CASSETTEBIBLIOTHEEK

Zoals in 6502 KENNER 16 al is aangekondigd, gaat de redactie de service uitbreiden met een cassette service. Er zijn op het moment twee cassettes leverbaar, opgenomen in KIM + JUNIOR hypertape formaat. Als de source code (alleen Micro Ade nog) aanwezig is dan wordt op de andere zijde van de cassette dat ook opgenomen. In ieder geval is de object (tweemaal) aanwezig . Wat er op de cassette staat en hoe wordt met een begeleidend schrijven duidelijk gemaakt. Gebruik en beschrijving van de programma's is in de 6502 KENNER of KIM KENNER te vinden. S achter de programma naam betekent ook Micro Ade source

<u>KIM cassette nr 1</u>		<u>JUNIOR CASSETTE nr 1</u>	
1. Microchess versnellen	nr 12 S	1. Aut register uitlezing	nr 16 S
2. Supertape	nr 12 S	2. Moonlander	nr 15 S
3. Reactiesnelheidsmeter	nr 12 S	3. Lotto	nr 15 S
4. Verkeerslichten	nr 16 S	4. Browse	nr 16 S
5. Locate en Replace	nr 18 S	5. Stopwatch	nr 18 S
6. Eprom programmer	nr 5 S	6. Belgische LOTTO	nr 18 S
7. KIM schaakprogramma	nr 11	7. SC/MP casstte afregeling	nr 18 S
8. Schaakopeningen	nr 8	8. Zenuwslag	nr 17 S
9. Talenstudie hulpprog	nr 13	9. Muziekdoos	nr 17 S
10. Mastermind	nr 4	10. Dag naar week omrekening	nr 18 S
11. Target 1 en 6 kolommen	nr 8	11. Dokatimer	nr 17 S

Programma's in de cassette bibliotheek zijn of origineel van onze clubleden of bewerkingen van First Book of KIM programma's. 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 KIM club is en wordt geen handel .

Kosten per cassette f 12,50

Bestellen van de cassettes kan door een girobetaalkaart of bankcheque (groene of EURO) en een briefje met het adres en cassette nummer (KIM =1 , JUNIOR = 2) in een envelop te sturen aan het redactie -secretariaat (Hans Otten, Ottoborrengoed 33, 3871 MJ Hoevelaken) te zenden Levering kan enige weken duren . Andere wijzen van

bestellen bestaan niet .

6502 VRAAG EN AANBOOD

Te koop : ASCII display module van Visser Assembling Electronics.
Teletype compatible 16 regels met 64 karakters, interface
met 2 20 mA current loops en video signaal 1 V top-top.
Prijs f 200,- of BF 3000,- incl verzendkosten en handboek.
Van Nieuwenhove Koen, Consciencestraat 60 bus 4
B3000 Leuven België

Te koop : 1 penrecorder Leeds Narthrep met doc, 1 anal 1 marker f 85,-
2 micro fiche readres Philips per stuk - 40,-
2 ponsbandlezers 40 kar-s (transp mech + kop, geen el)- 35,-
1 magn card reader en 20 magn kaarten geen electr - 15,-
20 alma reed relais per stuk - 1,50
B. Eenhuizen , Jol 133 1186 SH Amstelveen tel. 020 - 435348

Gevraagd: Wie heeft documentatie of ervaring met een TA21 van Adler
Information Systems , een machine voor het vastleggen van
boekhouding op cassette met een bolkop printer en een 8080 ?
C. Totté, Gr. Albrechtweg 4 3331 HP Zwijndrecht tel 078-127324

Te koop : Maandblad voor Hobby Electronica, EL0 en Hobbit
jaargangen 1977, '78 , '79 en '80 voor f 20,- per jaargang
Paul van Niekerk Kievitslaan 8 3233 BE Oostvoorne

6502 AGENDA

KIM club bijeenkomsten in 1982 :

16 januari 20 maart 15 mei ... 18 september ... 20 november

6500 gebruikers groep Delft n E-cafe TH :

d0 17 -12 - 1981 , 21 - 01 - 1982 , 25 - 02 1982

DE **KIM** GEBRUIKERS CLUB:
EEN CLUB VAN **6502** KENNERS

0/5/12/85
0/5/10
0/5/10
0/5/10