All other Elektuur Junior articles 1981

2-54 junior-tekst, show text on led displays

3-54 junior-journaal, practical tips

4-52 junior-software aangevuld

6-53 Junior Computer als Voltmeter (G.Sullivan)

met een bijdrage van U. Seyffert

junior-tekst

Het display van de junior-computer is niet alleen maar goed voor de weergave van numerieke, hexadecimale data. Met een beetje water in de wijn (een zeven-segment-alfabet) kan er ook een tekst op worden weergegeven. De tekst kan statisch, onveranderlijk zijn (maximaal zes letters) of dynamisch. In het laatste geval is er sprake van een soort lichtkrant.

Een interessante vraag is hoe dat allemaal in zijn werk gaat. In het normale geval wordt de monitor-routine SCANDS gebruikt voor het displayen van data. Het gaat daarbij echter uitsluitend om de mogelijkheden Ø . . . F per display. Dit is uitgebreid besproken in hoofdstuk 7 van het boek "Junior Computer 2".
Voor de weergave van tekst kan dus geen gebruik worden gemaakt van monitor-routines. We moeten de subroutine SHOW opnieuw maken, maar dan met een aangepaste opzoektabel (die voor elke letter een passend zeven-segmentpatroon bevat).

In tabel 1 staat een overzicht van letters en cijfers met daarbij de data die op poort A moet worden gezet voor de weergave ervan. De tabel is voor een deel gebaseerd op suggesties van de heer Seyffert. Toegegeven: voor letters met schuine strepen (M, N, V, W, X) is het maar behelpen, omdat de segmenten horizontaal of vertikaal zijn opgesteld, maar de ervaring leert dat het went.
Eerst maar eens een programma, waarmee een woord van maximaal zes letters permanent in beeld verschijnt. Bijvoorbeeld het woord "Junior", dat men op het display aantreft op de voorplaat van het maartnummer van Elektuur en op de voorplaat van de junior-computer-boeken. Het programma (JUNIOR) staat in tabel 2.
De aangepaste subroutine SHOW heet hier SHOWDS en de opzoektabel, waarin per display is vastgelegd wat er moet worden weergegeven, heet TXT (tekst-geheugen). Het Y-indexregister doet onder meer dienst als display-teller/tekst-index. Achtereenvolgens en periodiek (display-multiplexing) doorloopt Y de waarden ØØ . . . Ø5; zodra Y na INY

Tabel 1

Ø	40	a	20	M	48
1	79	b	Ø3	n	2B
2	24	c	27	o	23
3	30	d	21	P	ØC
4	19	e	Ø4	q	18
5	12	E	Ø6	r	2F
6	Ø2	F	ØE	s	52
7	78	G	42	s(5)	12
8	ØØ	g(9)	10	t	Ø7
9	10	h	ØB	u	63
A	Ø8	H	Ø9	V	41
B(b)	Ø3	i	7A	W	Ø1
C	46	i	6F	X	36
D(d)	21	J	72	ij	11
E	Ø6	K	ØA	Z	64
F	ØE	L	47	sp	7F

Tabel 2

```
JUNIOR  0200  A9 7F     LDA #7F
        0202  8D 81 1A  STA-PADD     PBØ . . . PB6 uitgang
DISMPX  0205  A2 Ø8     LDX #Ø8      begin met Di1
        0207  AØ ØØ     LDY #ØØ      dus displayteller Y = ØØ
ONEDIS  0209  84 Ø4     STY-TEMPY    bewaar display-tellerstand
        020B  20 17 Ø2  JSR-SHOWDS   laat het display zien
        020E  A4 Ø4     LDY-TEMPY    haal tellerstand op
        0210  C8        INY          verhoog display-teller
        0211  CØ Ø6     CPY #Ø6      alle zes een keer gehad?
        0213  FØ FØ     BEQ DISMPX   zoja: begin opnieuw
        0215  DØ F2     BNE ONEDIS   zonee: volgende display
SHOWDS  0217  B9 3Ø Ø2  LDA-TXT,Y   haal segmentpatroon op
        021A  8D 8Ø 1A  STA-PAD      zet dat patroon op poort A
        021D  8E 82 1A  STX-PBD      schakel display in
        0220  AØ 7F     LDY #7F
DELAY   0222  88        DEY          } wacht een tijdje
        0223  1Ø FD     BPL DELAY    }
        0225  8C 8Ø 1A  STY-PAD      Y = FF (blanking) naar poort A
        0228  AØ Ø6     LDY #Ø6
        022A  8C 82 1A  STY-PBD      schakel display uit
        022D  E8        INX
        022E  E8        INX          bereid volgende display voor
        022F  6Ø        RTS
TXT     0230  61        "J"          opzoektabel
        0231  63        "u"          Y = tekstindex
        0232  2B        "n"          (Y = ØØ . . . Ø5)
        0233  6F        "i"
        0234  23        "o"
        0235  2F        "r"
```

Tabel 3

```
JUNTXT  0200  A9 7F      LDA #7F
        0202  8D 81 1A   STA-PADD     PB0 . . . PB6 uitgang
        0205  A5 00      LDA-NUM      inh. NUM (0000) naar accu
        0207  38         SEC          C = 1
        0208  E9 05      SBC #05
        020A  85 02      STA-NUMCOR   NUMCOR ←NUM minus 05
BEGIN   020C  A9 00      LDA #00
        020E  85 01      STA-NUMVAR   eerste display-tekst
DSTIME  0210  A9 6F      LDA #6F
        0212  85 03      STA-DISCNT   leg teksttijd vast
DISMPX  0214  A2 08      LDX #08      begin met Di1
        0216  A0 00      LDY #00      dus display-teller Y = 00
ONEDIS  0218  84 04      STY-TEMPY    bewaar display-tellerstand
        021A  98         TYA          Y naar accu
        021B  18         CLC          C = 0
        021C  65 01      ADC-NUMVAR   Y ←Y + inh. NUMVAR (0001
        021E  A8         TAY          accu naar Y
        021F  20 39 02   JSR-SHOWDS   laat een display zien
        0222  A4 04      LDY-TEMPY    haal tellerstand op
        0224  C8         INY          verhoog display-teller
        0225  C0 06      CPY #06      alle zes een keer gehad?
        0227  F0 02      BEQ TMECHK   zoja: door naar tijdskontrole
        0229  D0 ED      BNE ONEDIS   zonee: volgende display
TMECHK  022B  C6 03      DEC-DISCNT   teksttijd om?
        022D  D0 E5      BNE DISMPX   zonee: nog een rondje van 6
        022F  E6 01      INC-NUMVAR   zoja: volgende tekst
        0231  A5 02      LDA-NUMCOR
        0233  C5 01      CMP-NUMVAR   alle tekst gehad?
        0235  B0 D9      BCS DSTIME   zonee: laat nieuwe tekst zien
        0237  90 D3      BCC BEGIN    zoja: begin opnieuw
SHOWDS  0239  B9 00 03   LDA-TXT,Y
        023C  8D 80 1A   STA-PAD
        023F  8E 82 1A   STX-PBD
        0242  A0 7F      LDY #7F      zie het programma
DELAY   0244  88         DEY          "JUNIOR"
        0245  10 FD      BPL DELAY    TXT: 0300
        0247  8C 80 1A   STY-PAD      tekstindex:
        024A  A0 06      LDY #06      Y + inh. NUMVAR
        024C  8C 82 1A   STY-PBD
        024F  E8         INX
        0250  E8         INX
        0251  60         RTS
```

Tabel 4

```
     0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
030 7F 7F 7F 7F 7F 7F 21 06 7F 63 6F 07 0A 23 48 12
031 07 7F 41 20 2B 7F 0B 06 07 7F 23 0C 07 06 47 0C
032 2F 23 10 2F 20 48 48 20 7F 23 0C 7F 0C 20 10 6F
033 2B 20 7F 02 30 7F 41 20 2B 7F 03 23 06 0A 7F 79
034 7F 6F 12 7F 79 03 7F 06 2B 7F 2B 6F 06 07 7F 24
035 79 3F 21 6F 07 7F 6F 12 7F 10 06 0A 23 2F 2F 6F
036 10 06 06 2F 21 7F 6F 2B 7F 21 06 7F 07 01 06 06
037 21 06 7F 21 2F 63 0A 7F 41 20 2B 7F 03 23 06 0A
038 7F 79 XX XX XX XX XX XX XX XX XX XX XX XX XX XX

0000 (NUM): 81
```

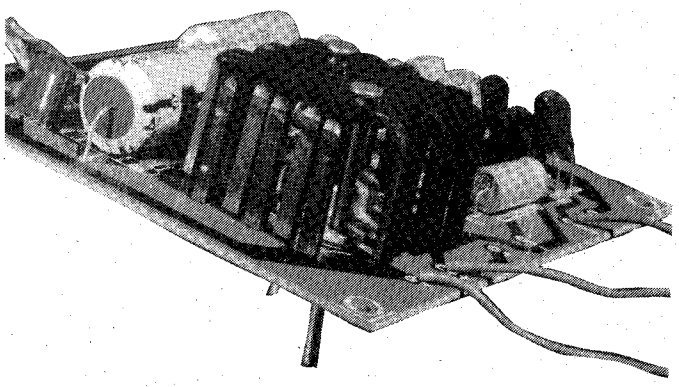
06 wordt, krijgt deze de beginwaarde 00 (sprong naar DISMPX voor een nieuw rondje van zes). Tijdens SHOWDS doet Y dienst als afteller (display-oplichttijd). Vandaar de noodzaak om de waarde van Y als display-teller/tekst-index te bewaren in TEMPY (0004) vóór de sprong naar SHOWDS.
De funktie van X is gelijk aan die tijdens SHOW: display-schakelaar via poort B. Achtereenvolgens en periodiek doorloopt X de waarden 08, 0A, 0C, 0E, 10 en 12.

Nu wat beweeglijker

Prachtig, zo'n statische tekst, maar het wordt op den duur wel wat eentonig. Waarom niet om de zoveel tijd een andere tekst op het display zetten? Dus geen woorden maar hele zinnen? Dat kan. Met het programma JUNTXT van tabel 3, waarmee een lichtkrant is te realiseren. Het gaat hier om een uitbreiding van JUNIOR (tabel 1). Pagina 03 is als tekstgeheugen beschikbaar. Dat zijn 256 letters en dat is een gemiddelde alinea tekst.
Er wordt ook hier gebruik gemaakt van de subroutine SHOWDIS. Nu is echter TXT gelijk aan 0300 en is er een verschil tussen de waarde van Y als display-teller en de waarde van Y als tekst-index. Er geldt: tekst-index = tellerstand plus inhoud NUMVAR (adres 0001). Gedurende het displayen van een bepaalde tekst (tijdsduur regelbaar door wijziging van de inhoud van 0211) is de waarde van NUMVAR konstant. Zodra de tijd om is wordt NUMVAR met één opgehoogd: de tekst schuift een plaats op naar links, het rechtse display toont een nieuwe letter en zo krijg je een lichtkrant. Zodra NUMVAR na ophoging één hoger is dan de inhoud van NUMCOR, beginnen we weer bij het begin omdat alle tekst de revu is gepasseerd. Dat zit zo: de inhoud van NUMCOR is 05 lager dan die van NUM, en in NUM (0000) heeft de gebruiker het rechter byte ADL gestopt van het adres in pagina 03, waarop de laatste letter van de tekst staat.
In tabel 4 staat een voorbeeld van een tekst voor weergave met het programma JUNTXT van tabel 3. De tekst is een mededeling voor de bezitters van het boek *"Junior Computer 1"*. Begin een tekst altijd met zes spaties (7F), zodat er een duidelijke scheiding is tussen einde en begin van de tekst.
Binnenkort, als de interface-kaart beschikbaar is, hoeven de programma's en, in het geval van JUNTXT, de tekst op pagina 03, slechts één keer te worden ingetoetst. Daarna kunnen ze op de band worden vastgelegd.

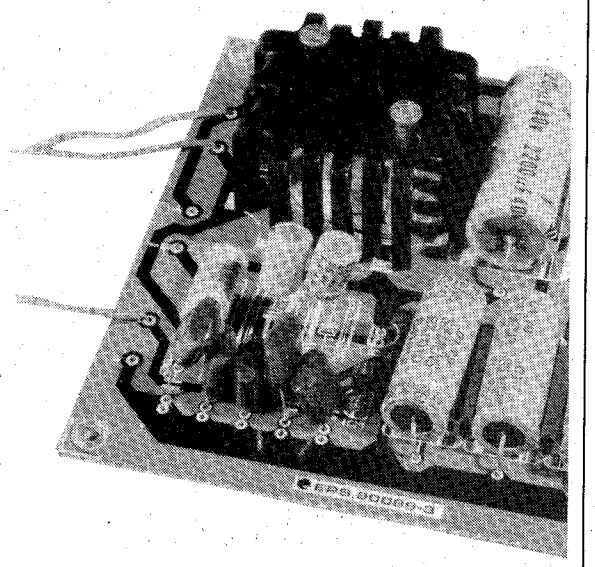
Literatuur:
Hoofdstuk 7 van het boek "Junior Computer 2".

7



Figuur 7. Luchtfoto van een door een lezer opgebouwde voedingsprint. Gezien vanuit zuidwestelijke richting.

8



Figuur 8. Als figuur 7, maar nu uit oostelijke richting.

Tabel 1

```
START    0210 A9 00      LDA #00
         0212 85 00      STAZ-COUNTL
         0214 85 01      STAZ-COUNTH
BEG1     0216 A5 01      LDAZ-COUNTH
         0218 85 FB      STAZ-POINTH
         021A A5 00      LDAZ-COUNTL
         021C 85 FA      STAZ-POINTL
         021E A9 FF      LDA #FF
         0220 85 02      STAZ-DSCNT
DISPLAY  0222 20 8E 1D   JSR-SCANDS
         0225 C6 02      DECZ-DSCNT
         0227 D0 F9      BNE DISPLAY
         0229 E6 00      INCZ-COUNTL
         022B D0 02      BNE BEG2
         022D E6 01      INCZ-COUNTH
BEG2     022F A5 01      LDAZ-COUNTH
         0231 C9 20      CMP #20
         0233 D0 E1      BNE BEG1
         0235 00         BRK
         1A7E 00         IRQ-sprongvektor
         1A7F 1C         (i.v.m. BRK)
```

Tabel 1. Het programma, nodig om de software-teller (boek 1, pagina 78/79) zichtbaar te laten tellen. Door op adres 0222 JSR-SCANDS (20 8E 1D) te vervangen door JSR-SCAND (20 88 1D) krijgt men achtereenvolgens de inhoud van de adressen 0000 . . . 1FFF te zien. Hiermee is de software-teller niet alleen leuk of leerzaam, maar ook nog nuttig geworden: een snelle visuele check van geheugen-inhouden. Zie ook figuur 5.

gebleven draadeindjes neemt men stuk voor stuk letterlijk in de tang; onder gelijktijdige plaatselijke verhitting van het soldeerpunt trekt men elk draadeinde uit, als ware het een rotte kies.
Met name in verband met de uitbreiding in de vorm van de interface-kaart is het noodzakelijk dat de pullup-weerstand R5 wordt verlaagd van 4k7 naar 470 Ω; u wordt dan beloond met buitengewoon schone flanken van het signaal RAM-R/W. Het is raadzaam om hetzelfde te doen met de weerstanden R14, R15 en R16, zodat ook de chip-select van EPROM, PIA en RAM gegarandeerd vlekkeloos verloopt.

Pas op bij het gebruik van bevestigingsringen en -moeren van S24 en S25, en van metalen afstandsbussen voor de montage van de hoofdprint. Er kunnen makkelijk plaatselijk kortsluitingen ontstaan tussen printsporen. Neem bij voorkeur nylon-afstandsbussen met aan beide zijden een gat met 3 mm schroefdraad. Vervang eventueel bij S24 en S25 de kontramoer door een isolatiering. Neem het zekere voor het onzekere, dus deze voorzorgsmaatregelen.

Punt 13: Meer over "meer junior-geheugen"
Hoe je (voorlopig) meer K op de JC kunt aansluiten is verteld in een artikeltje in Elektuur, oktober 1980. Eerst even iets rechtzetten: in tegenstelling tot wat er in dat artikel is gezegd kan op de standaard-junior-computer slechts één RAM/EPROM-kaart *direkt* worden aangesloten. Dat komt omdat de adresbus en de databus op de hoofdprint niet zijn gebufferd (bij een grotere belasting gaan de adreslijnen A0 . . . A9 moeilijk doen). De interne adresbus en databus van een RAM/EPROM-kaart zijn dat overigens wel. Dus er kunnen niet meerdere geheugenkaarten worden aangesloten? Jazeker wel! Op de interface-kaart volgt namelijk direkt op de aansluiting op de uitbreidingskonnektor van de hoofdprint buffering van adresbus en databus. Die interface-kaart heeft óók weer een uitbreidingskonnektor, waarop een bestaande SC/MP-busprint kan worden aangesloten die plaats biedt aan maximaal *vijf* RAM/EPROM-kaarten.
Wil men meer geheugen, dan is het niet altijd nodig om (een) geheugenkaart(en) te gebruiken. De interface-kaart biedt plaats aan RAM en EPROM. Een deel van dit stuk geheugen is nodig voor het funktioneren van de interface-kaart (kassette-software, printer-monitor), een ander deel staat de junior-programmeur vrij ter beschikking.
In het artikel "meer junior-geheugen" is aangegeven hoe in afwachting van de interface-kaart de junior-computer geschikt kan worden gemaakt voor aansluiting op een RAM/EPROM-kaart. Voor toekomstige interface-kaartbezitters is dit een tijdelijke oplossing, voor diegenen die zich beperken tot de kombinatie standaard-junior-computer + RAM/EPROM-kaart moet een dergelijke hardware-aanpassing permanent aanwezig zijn. Vandaar dat we nu toch een elegant alternatief van de heer De Mülder bespreken.

dag in dag uit het weten waard

junior-journaal

Vragen, opmerkingen, suggesties, wensen, hartekreten en verzuchtingen (wanneer . . .?). Van partikulieren en uit de onderwijswereld. Per telefoon, telex of brief. En allemaal over de junior-computer. Voldoende aanleiding voor dit artikel, waarin een aantal zaken puntsgewijze zal worden besproken.

met bijdragen van H.P. Diehl en H.D. de Mülder

Na uw reakties volgt nu onze aktie. Aktie in de vorm van dit artikel, met tips, aanvullende opmerkingen, alternatieven en andere nuttige informatie voor de bouwers en gebruikers van de junior-computer. Punt voor punt behandeld: punten op de "i" van "junior". Daar gaan we:

Punt 1: Decimaal rekenen

Zoals bekend uit boek i kan men zowel binair als decimaal optellen en aftrekken. Gaat men decimaal te werk, dus komt de instruktie SED (F8) in het programma voor, dan krijgen we problemen als het probleem op te lossen, met name als u uw junior-computer gaat uitbreiden.

Een voorbeeld van decimaal rekenen is wel handig. We nemen het aangepaste optelprogramma van pagina 63 van boek 1:

```
Ø1ØØ  18       CLC
Ø1Ø1  A9 13    LDA #13
Ø1Ø3  F8       SED          D=1 decimaal
                            rekenen
Ø1Ø4  69 Ø8    ADC #Ø8
Ø1Ø6  D8       CLD          D = Ø binair
                            rekenen
Ø1Ø7  ØØ       BRK
1A7E  ØØ       IRQ-vektor wijst naar monitor
1A7F  1C
```

na afloop van het programma via een BRK, of na de afwerking van één instruktie bij stap-voor-stap naar de monitor (1CØØ) wordt gesprongen. Wat is namelijk het geval? Als tijdens de sprong naar de monitor de D-vlag 1 is gebeuren er gekke dingen: de toetsen F, +, AD, DA, PC en GO zijn ineffektief geworden; de toets*funkties* A . . . F verloren gegaan. De toetsfunktie AD wordt overgenomen door toets A, funktie DA door B, funktie + door C, funktie GO door D en funktie PC door toets E. Met andere woorden: de numerieke toetsfunkties A . . . F zijn niet beschikbaar. Adressen die A . . . F bevatten kunnen niet direkt worden opgegeven, wèl via een omweg: toets het dichstbij gelegen lagere adres met uitsluitend Ø . . . 9 in en verhoog het adres zo vaak als nodig is via de plustoetsfunktie (toets C!) . . . Hoogst omslachtig dus.

Hoe komt dat? In de monitor-subroutine GETKEY wordt de toetswaarde bepaald via optelling van nul, één of twee keer Ø7 bij een basiswaarde. Alles gaat prima zolang dat binair gebeurt.

En is daar nou werkelijk niets aan te doen? Jazeker wel! Zorg er bijvoorbeeld voor dat de SAVE-routine van de monitor, net zoals de RESET-routine, de instruktie CLD met opcode D8 bevat. Voor het programma kan dat geen kwaad omdat de situatie D=1 is vastgelegd in het geredde P-register (ØØF1) en na terugkeer uit de monitor (programmadeel GOEXEC) wordt hersteld. Een en ander houdt de volgende wijzigingen in van de EPROM:

```
1C1A  4C 32 1C  JMP-START
1C31  78        SEI
1C32  D8        CLD D=Ø START
                (tijdelijk) binair rekenen
```

Hiermee is bereikt dat het centrale deel START van de monitor begint met CLD. Ook na SAVE is er nu sprake van binair rekenen en kunnen alle toetsen normaal worden gebruikt. Het is overigens niet de bedoeling om de aldus gewijzigde EPROM in de ESS te brengen. Er zijn namelijk nog andere mogelijkheden om

Na het opgeven van het startadres en het indrukken van GO wikkelt het programma zich af; er volgt een sprong naar de monitor. De toetsen zijn nu normaal bruikbaar omdat na afloop van de optelling weer op binair rekenen is overgeschakeld. Het intoetsen van adres ØØF3 geeft de data 21, zijnde het resultaat van de optelling (bezitters van de eerste druk van boek 1: nu klopt het wèl!).

Het had ook anders gekund:

```
Ø1ØØ  18       CLC
Ø1Ø1  A9 13    LDA #13
Ø1Ø3  F8       SED          D=1 decimaal
                            rekenen
Ø1Ø4  69 Ø8    ADC #Ø8
Ø1Ø6  ØØ       BRK
1A7E  ØØ       IRQ-vektor wijst naar
               1AØØ
1A7F  1A
1AØØ  D8       CLD          D=Ø binair
                            rekenen
1AØ1  4C ØØ 1C JMP-SAVE spring naar
               monitor
```

Aan het eind van het programma leidt de BRK ons via de IRQ-sprongvektor naar 1AØØ; na de overschakeling op binair rekenen volgt de sprong naar de monitor. Voor het ineens doorlopen van een programma is de zo juist besproken metode niet de handigste, voor het stap-voor-stap doorlopen van het programma vanaf Ø1Ø3 is het de enige metode.

Er moet echter nog meer gebeuren in het stap-voor-stap-geval. Een hardware-aanpassing namelijk. Het stap-voor-stap doorlopen van de monitor is verboden. Dit omdat de taak van de monitor (display-multiplexing, wachten op een ingedrukte toets) met zich meebrengt dat een flink aantal instrukties kontinu en periodiek moet worden doorlopen. Vandaar de aanwezigheid van poort N3 in het principeschema op pagina 16/17 van boek 1. Zolang het signaal K7 hoog is (EPROM niet geadresseerd) ontstaat uit elke SYNC-puls (opgewekt tijdens de opcode-fase van een instruktie) een NMI, die na afwerking van de lopende instruktie leidt tot een sprong naar de monitor (mits de NMI-spronvektor gericht staat op 1CØØ). Is daarentegen

K7 laag (monitor geadresseerd), dan ontstaat geen NMI.

Welnu: Tot de komende uitbreidingen hoort een printer-monitor, die het adresbereik 1ØØØ . . . 13FF omvat dat wordt geselekteerd door K4. Ook de printer-monitor mag niet stap-voor-stap worden doorlopen. Dat betekent dus uitbreiding van de schakeling rond N3. De details staan in figuur 1b. Er zijn nu twee mogelijkheden om een NMI via SYNC te blokkeren. Voor het ene signaal neemt men K4 of K7, voor het andere K6. (Over figuur 1b is het laatste woord nog niet gezegd; de praktijk komt later aan de orde). Waar het hier en nu om gaat is dat door de aansluiting van K6 ook een programma in pagina 1A niet stap-voor-stap kan worden doorlopen. Het decimale optelprogramma van daarnet kan nu ook stap-voor-stap worden doorlopen.

Punt 2: Maak zelf uw min-toets

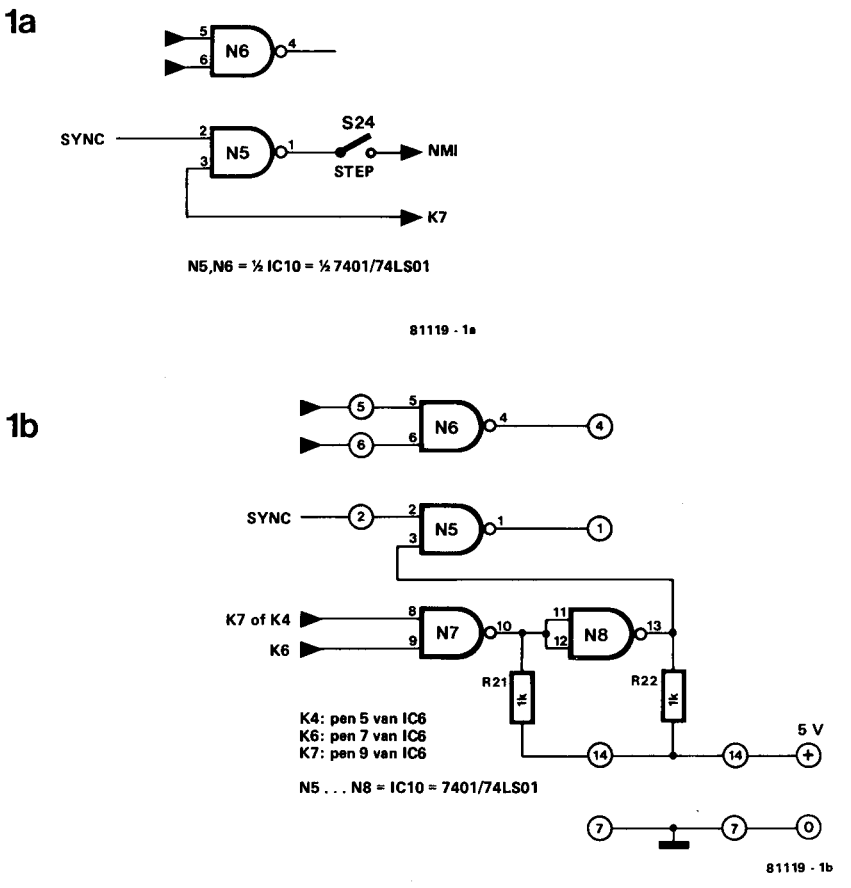
Zoals bekend kan men tijdens het verblijf in de monitor het te displayen adres met één verhogen door gebruik te maken van de plustoets. Zo vaak als men wil. Bij het checken van een (ingetoetst) programma wil het nog wel eens voorkomen dat men na het indrukken van de plustoets wil weten wat er ook weer op het vorige adres stond. Een *mintoetsfunktie* is dan reuze handig. Dat kan heel eenvoudig via de STOP/NMI-toets. De NMI-sprongvektor (1A7A en 1A7B) richten we op het adres 1AØØ, waar zich het volgende programma afspeelt:

```
MIN  1AØØ 48       PHA          red accu op de stack
     1AØ1 A5  FA   LDAZ-POINTL  haal rechter adresbyte
                                op
     1AØ3 DØ  Ø2   BNE ADL      door naar ADL als
                                A≠ØØØØØØØØ
     1AØ5 C6  FB   DECZ-POINTH  verlaag POINTH met 1
ADL  1AØ7 C6  FA   DECZ-POINTL  verlaag POINTL met 1
     1AØ9 68       PLA          herstel oorspronkelijke
                                A
     1AØA 4Ø       RTI          terug naar monitor
```

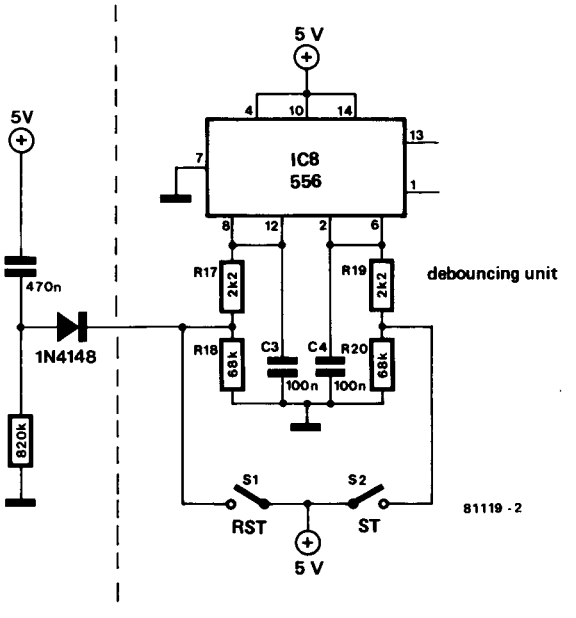
Waarmee is bereikt dat het indrukken van de STOP/NMI-toets tijdens het verblijf in de monitor verlaging van het gedisplayde adres met één tot gevolg heeft. Men kan nu programma's in een willekeurige volgorde doorlopen. Het is nu ook mogelijk om data via toets DA op te geven in de volgorde met afnemend adres, omdat de werking van de mintoets net zoals die van de plustoets onafhankelijk is van de vraag of er sprake is van de adres-mode danwel van de data-mode. N.B. Ook zeer handig voor het bekijken van de stapel (adres Ø1FF en *lager*).

Punt 3: Automatisch opstarten

Direkt na het inschakelen van de junior-computer moet de toets RST worden ingedrukt. De monitor staat dan klaar voor gebruik. Met de schakeling van figuur 2, van de heer H.P. Diehl kan het indrukken van RST achterwege blijven omdat automatisch wordt gereset. Daartoe zijn drie extra onderdelen nodig. Met een beetje wikken en wegen (let op



Figuur 1. De hardware, nodig om de stap-voor-stap-procedure te blokkeren tijden shet verblijf in de monitor (1a) of tijdens het verblijf in één van de twee geheugengebieden die door de twee chip-selektsignalen K4 of K7 respektievelijk K6 worden geaktiveerd (1b).



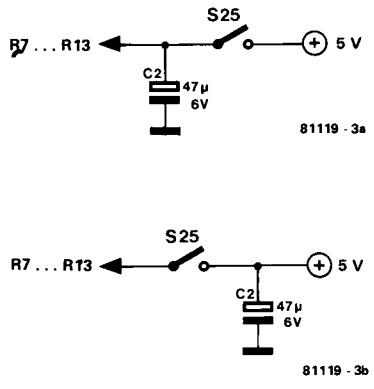
Figuur 2. Sterke koffie bij het opstaan: Drie extra komponenten links van de streep zorgen voor het automatisch opstarten (RESET) na het inschakelen van de junior-computer.

kortsluitingen!) zijn deze extra onderdelen onderop de hoofdprint ter plaatse van de RST-toets aan te brengen.

Punt 4:: Adresmode en datamode
Zodra de monitor is bereikt na het indrukken van RST of via 1CØØ (na een BRK of bij stap-voor-stap-programmeren) is er *automatisch* sprake van de adres-mode. Het indrukken van toets AD — zoals aangegeven in de toetsprogramma's van de boeken 1 en 2 — is dan overbodig. Toets AD hoeft alleen te worden ingedrukt als wordt "teruggeschakeld" van de data-mode. Het verschil tussen AD- en DA-mode: in de adres-mode leidt het indrukken van een numerieke toets tot wijziging van minstens vier displays: in ieder geval de vier adres-displays, waarschijnlijk ook de twee data-displays. In de data-mode zien alleen de twee data-displays er anders uit.

Punt 5: Display-schakelaar
Met de display-schakelaar (S25 in het principeschema op pagina 16/17 van boek 1) is het mogelijk om het display uit te schakelen. Wanneer men een computer-sessie tot in de kleine uurtjes heeft volbracht en de volgende dag weer verder wil zonder allerlei data opnieuw in te hoeven toetsen laat men de juniorcomputer ingeschakeld, maar schakelt het display uit. Zo wordt een onnodige negatieve invloed op de levensduur van de displays voorkomen. Bij het begin van een volgende sessie schakelt men het display weer in. (Uiteraard is dit allemaal niet meer nodig als men data op band kan zetten, dus als de kassetteinterface voorhanden is).
In een heel enkel geval kan er iets mis gaan bij het weer inschakelen van het display. Kondensator C2 (figuur 3a) moet weer tot de volle voedingsspanning worden opgeladen. Direkt nadat de verbinding via S25 weer tot stand komt kan er in sommige gevallen een flinke stoorpuls ontstaan op de +5 V-voedingslijnen die tot gevolg heeft dat de zo zorgvuldig bewaarde RAM-geheugeninhoud geheel of gedeeltelijk verloren gaat. Het hangt dan van *uw* geheugen af hoe snel de schade kan worden hersteld. De remedie: sluit de pluskant van C2 aan op de andere kant van S25 en u heeft geen centje pijn. Op de print zit C2 direkt naast de aansluitingen van S25.

3



Figuur 3. Door het verplaatsen van de pulsaansluiting van C2 heeft het uit- en weer inschakelen van de display-schakelaar S25 gegarandeerd niet tot gevolg dat geheugeninhouden worden uitgeschakeld.

Punt 6: Nogmaals: junior-tekst
Naar aanleiding van het artikel "juniortekst" (Elektuur, februari 1981) nog enige aanvullende informatie in de vorm van de tabel van figuur 4. Hij is ingezonden door H.D. de Mülder en geeft alle 128 mogelijke verschillende kombinaties van segmentpatronen op. Patronen op dezelfde rij hebben een onderling identiek linker nibble H en patronen op dezelfde kolom een onderling identiek rechter nibble L. Het komt erop neer dat voor een oplichtend segment het overeenkomstige bit *nul* moet zijn.

Punt 7: Data ingeven
In de data-mode — een kwestie van DA indrukken — kan men data ingeven op de geheugenplaats met het gedisplayde adres. Maar niet altijd! Neem nou het volgende geval:

4

H \ L	Ø	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Ø																
1																
2																
3																
4																
5																
6																
7																

81119 - 4

Figuur 4. Een handige tabel van alle 128 mogelijke segmentpatronen.

toetsen:				display	
RST	(	AD	)	XXXX	XX
1	F	7	C	1F7C	88
DA				1F7C	88
A	B			1F7C	88

De inhoud van 1F7C blijft 88 en wordt niet AB. Adres 1F7C maakt deel uit van de EPROM, die nu eenmaal niet reageert op de in de datamode gebruikelijke schrijfoperatie. Een ander voorbeeld:

RST	(	AD	)	XXXX	XX
Ø	7	B	E	Ø7BE	YY
DA				Ø7BE	YY
2	F			Ø7BE	YY

(YY≠2F verondersteld)

Ook nu gebeurt er niets. Op adres Ø7BE is in de standaard junior-computer geen geheugen aangesloten. Schrijven van data is in dit geval "schrijven in de lucht".

Ook het "data dikteren" op het PIA-adres 1AD5 (RDFLAG) is onmogelijk.

Punt 8: Data lezen

De monitor zorgt ervoor dat de inhoud van het gedisplayde adres te zien is op de rechter twee displays. In sommige gevallen is de data niet stabiel. Klopt dat? Ja, dat klopt. Niet stabiele, of juister niet konstante data treft de monitor namelijk aan op PIA-adressen die er zijn voor het lezen van het timer-dataregister. Dat zijn RDTDIS (1AD4; interrupt geblokkeerd) en RDTEN (1ADC; interrupt mogelijk). Tijdens het verblijf in de monitor wordt de inhoud van het gedisplayde adres periodiek gelezen. Aangezien het timer-dataregister voortdurend van inhoud verandert is het niet verwonderlijk dat in dat geval de data-displays een onrustig beeld te zien geven.

Punt 9: Nog meer PIA-adressen

In hoofdstuk 6 van boek 2 (eerste druk) is er sprake van 17, onderling essentieel verschillende PIA-adressen. Dat zijn er dus twee te weinig. Het lezen van de timer met geblokkeerde interrupt gaat namelijk niet via het lezen van één der adressen CNTA . . . CNTD, maar van 1AD4 (RDTDIS). Het lezen van de timer met toegestane interrupt geschiedt niet via het lezen van één der adressen CNTE . . . CNTH, maar van 1ADC (RDTEN). Het programmeermodel volgens figuur 12 op pagina 60 van boek 1 is dus niet volledig; de tabel op pagina 89 is wat dit betreft wèl in orde. In de tweede druk van boek 2 zal e.e.a. worden rechtgezet.

De negentien verschillende PIA-geheugenplaatsen zijn elk echter via meer dan één adres bereikbaar voor lezen en/of schrijven. Tabel 1 op pagina 89 maakt duidelijk waarom. Als we de zeven kruisjes voor AØ . . . A7 in het geval PIA-RAM even buiten beschouwing laten, en als we de kruisjes onder A12, A11 en A1Ø vergeten (voor de selektie van de PIA moet K6 nul zijn, dus A12=1, A11=1 en A1Ø=Ø), dan betekent elk ander kruisje in de tabel een verdubbeling van het aantal adressen waarop een bepaalde PIA-geheugenplaats bereikbaar is.

Adreslijn A8 bijvoorbeeld speelt geen enkele rol bij het PIA-gebeuren; dus A8=X=Ø of 1=dontcare=mag van alles zijn, als het maar Ø of 1 is. Dit betekent dat alle geheugenplaatsen in de 6532-PIA, óók de 128 byte PIA-RAM, zowel op een bepaalde adres in pagina 1A bereikbaar zijn (bijvoorbeeld de RAM-plaats op 1A2F) als op het overeenkomstige adres in pagina 1B (voorbeeld: 1B2F).

Voor poort A zijn A3, A4, A5 en A6 niet van belang leert ons de tabel. Poort A (PAD) is dus bereikbaar op de adressen 1AØØ, 1A88, 1A9Ø, 1A98, 1AAØ, 1AA8, 1ABØ, 1AB8, 1ACØ, 1AC8, 1ADØ, 1AD8, 1AEØ, 1AE8, 1AFØ en 1AF8.

En dan nog eens de overeenkomstige adressen in pagina 1B. Dat zijn dan bij elkaar 32 in plaats van één adres voor één en dezelfde PIA-geheugenplaats. Voor alle andere PIA-plaatsen van tabel 1 gelden, afhankelijk van het aantal X-en in een rij, meerdere adresseermogelijkheden. Het kan allemaal geen kwaad, maar het is wel goed om te weten. De situatie is te vergelijken met een bepaald postaal bezorgadres, dat meerdere postkodes bezit, waarvan de postbode op de hoogte is . . .

Punt 10: Achtvoudige adressering

U heeft het zich misschien nog nooit gerealiseerd, maar in de standaard-JC is elke geheugenplaats bereikbaar op acht adressen. Dat komt omdat slechts 8K van het 16K-adresbereik is gedekodeerd. De adreslijnen A13, A14, en A15 spelen geen enkele rol. De tabel op pagina 15 van boek 1 laat dat zien; deze adreslijnen zijn op geen enkel geheugen- of adresdekoder-IC aangesloten. Met als gevolg:

Alle pagina's ØX, met X=Ø . . . F komen overeen met 2X, 4X, 6X, 8X, AX, CX, EX (pagina's, beginnend met een even nibble);

Alle pagina's 1X, met X=Ø . . . F komen overeen met 3X, 5X, 7X, 9X, BX, DX, FX (pagina's, beginnend met een oneven nibble).

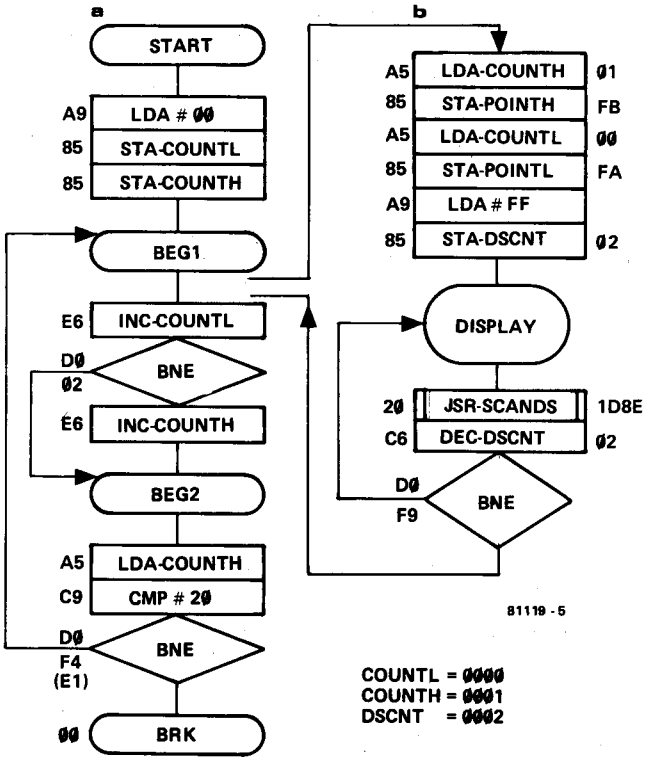
Een voorbeeld. Poort A (PAD) is op 32 "verschillende" adressen bereikbaar. Zie punt 9. In de standaard-versie kunnen we e.e.a. met acht vermenigvuldigen: 8 x 32 = 256!

In het artikel "Meer junior geheugen" (Elektuur, oktober 1980) is al aangegeven wat er moet gebeuren om deze achtvoudige adressering op te heffen – wat nodig is als men het geheugenbereik vanaf pagina 2Ø wil benutten. De meervoudige adressering betreft zowel het gebruik van de monitor als een gebruikersprogramma:

4C DF Ø2 (JMP-Ø2DF) is hetzelfde als 4C DF E2 (JMP-E2DF)

8D 3C Ø3 (STA-Ø33C) is hetzelfde als 4C 3C 93 (STA-933C)

5



Figuur 5. Het aangepaste programma van de software-teller. Aan figuur 9 op pagina 78 van boek 1 moet het gedeelte onder b worden toegevoegd om de teller zichtbaar te laten tellen. Het programma staat in tabel 1. De offset van de aan BRK voorafgaande BNE wordt in dit geval E1.

AD Ø Ø F 3 (inhoud accu): de data vindt men ook op het adres 20F3. Dus: AD 2 Ø F 3

Punt 11: Herdruk

Van boek 1 is sinds eind januari de derde druk beschikbaar. Er is van de gelegenheid gebruik gemaakt om een paar voorbeeldprogramma's aan te vullen. De software-teller van pagina 78/79 is er zo een. Velen verwachten de teller te *zien* tellen. In die fase zijn subroutines en SCANDS nog niet besproken (de software-teller is bedoeld als illustratie bij de voorwaardelijke spronginstrukties). Op pagina 91 is dit wel het geval; hier is in de derde druk het zichtbaar tellen uit de doeken gedaan. Voor de junioren van het eerste uur is het aangepaste programma te vinden in figuur 5 (gedetailleerd stroomdiagram) en in tabel 1 (het programma). Overigens: de presentatie van programma's doen we niet meer via een toetsprogramma, maar op een kompaktere manier: de eerste kolom voor de labels, dan een kolom met de opcode-adressen, dan de data, de mnemonics en eventueel een kolom kommentaar ("comments"). O ja, nog wat: de toevoeging Z aan de mnemonics van pagina-nulinstrukties is niet verplicht. Vandaar dat u die Z niet altijd tegenkomt.

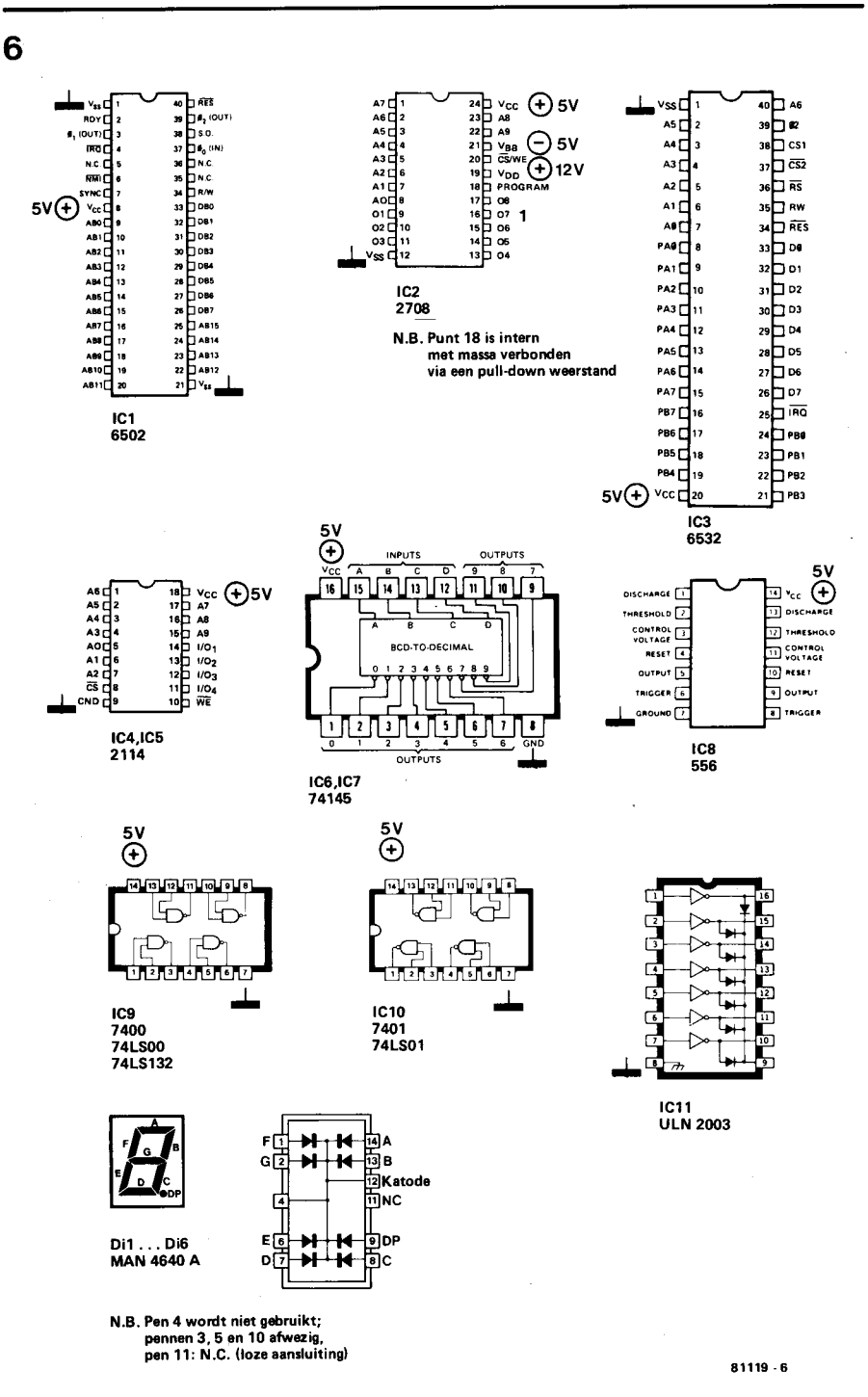
Punt 12: Nog wat bouwtips

Om te beginnen: de plattegrond van de IC's van de hoofdprint (bovenaanzicht) en van de zeven-segmentdisplays. Figuur 6 laat alles zien. De pinning van de uitbreidingskonnektor en die van de poortkonnektor zijn als aanhangsel 5 opgenomen in de tweede en derde druk van boek 1. Allemaal reuze handige visuele hulpmiddelen bij het onverhoopt servicen of foutzoeken.

Er zijn inmiddels een paar K junior-computers opgebouwd. Tot nog toe zijn ons buitengewoon weinig bouwproblemen bekend. Dat wil niet zeggen dat ons helemaal niets bekend is. Neem nou de JC-voeding waarvan we een paar foto's hebben gemaakt (figuren 7 en 8). Ons was verteld dat deze JC "het niet deed". Hetgeen niet verwonderlijk is. De +5 V-stabilisator wordt warm. Vandaar ook de noodzaak van een koelplaat. Het was deze bouwer bekend dat warme lucht opstijgt. Om nu te voorkomen dat het voor de koelplaat een keer "hemelvaartsdag" wordt is de koelplaat stevig "geaard" met behulp van twee spijkers! Het geheel is energiebewust aangepakt: een prima warmte-isolatie. Verder waren IC's en elko's verkeerd om gemonteerd of verwisseld.

Okay, het is allemaal weer op zijn pootjes terecht gekomen (ook die koelplaat), maar kennelijk is de bouwhandleiding van hoofdstuk 1 niet tegen dergelijke eventualiteiten bestand.

Lastiger daarentegen is het probleem dat ontstaat als de klokgenerator niet goed of helemaal niet funktioneert. Eén van de punten om te kijken (op de skoop) of de klok werkt is het knooppunt van



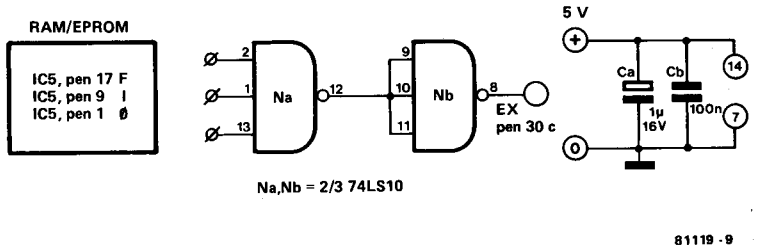
Figuur 6. De plattegronden van alle IC's van de hoofdprint van de junior-computer, en van de displays.

de anode van D1 en (ondermeer) pen 37 van IC1. De heer De Mülder wijst erop dat het met de hand aanraken van dit knooppunt bij ingeschakelde JC tot gevolg kan hebben dat de klokoscillator afslaat (invloed parasitaire kapaciteit). Het uit- en vervolgens weer inschakelen van het apparaat is de enige metode om de zaak weer aan de gang te krijgen.

Een goed kloksignaal levert de signalen Φ1 en Φ2, die elkaar niet mogen overlappen, d.w.z. ze mogen op geen enkel moment beide logisch 1 zijn. Afhankelijk van het toegepaste type 1 MHz-kristal en 6502-fabrikaat kan het soms gebeuren dat Φ1 en Φ2 elkaar wel overlappen, waardoor de JC het niet doet en soms IC1 warmer dan nodig wordt. Remedie: neem voor C1 een kondensator van 6p8.

Nu we het toch over het verwisselen van komponenten hebben: de hoofdprint is dubbelzijdig en doorgemetalliseerd en bevat vrij veel dunne sporen. Een verkeerd gehanteerde soldeerbout kan een ware ravage aanrichten. Gebruik liever geen *de*soldeerbout of -litze. Beter is het om van alle te verwijderen komponenten alles tot op een kort draadeindje af te knippen (óók van de IC's — de dure IC's zitten in een voetje, zie hoofdstuk 1 van boek 1). De over-

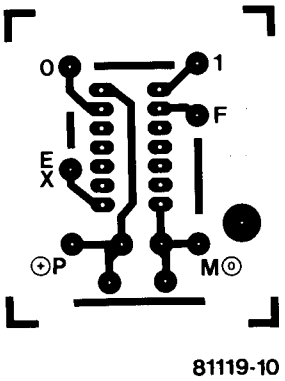
9



Figuur 9. Een alternatieve schakeling, nodig voor de kombinatie standaard-junior-computer plus één RAM/EPROM-kaart. Het punt "F" wordt niet aangesloten indien men de vektoren RES, NMI en IRQ in EPROM lokaliseert die op de RAM/EPROM-kaart zit.

Het alternatief van de heer De Mülder komt erop neer dat twee of drie uitgangssignalen van de hoofd-adresdekoder IC5 van de RAM/EPROM-kaart worden benut voor de sturing van EX. De schakeling van figuur 9 zorgt ervoor dat EX laag is als of "Ø" laag is (4K-adresbereik ØØØØ...ØFFF), of "1" laag is (4K-adresbereik 1ØØØ...1FFF), of "F" laag (4K-adresbereik FØØØ . . . FFFF). De draadbrug op de hoofdprint moet ook nu tussen de punten D en EX liggen. Een printje voor figuur 9 en van de hand van de heer De Mülder is te zien in figuur 10. Dit kan men in de buurt (direkt boven) van IC5 op de RAM/EPROM-kaart plaatsen. De aanpalende desbetreffende konnektorpennen kunnen als aanknopingspunt dienen voor de verbindingen EX, + en 0 . Indien men de vektoren RES, NMI en IRQ wil lokaliseren op EPROM die op de RAM/EPROM-kaart zit moet de doorverbinding "F" niet worden gemaakt.

10



Figuur 10. Een print (koperzijde) voor de schakeling van figuur 9.

Punt 14: Stap voor stap

Tijdens het stapsgewijze, dus instruktie voor instruktie doorlopen van een programma is het niet alleen mogelijk om na de sprong naar de monitor allerlei geheugenplaatsen te bekijken (lezen). Het is ook mogelijk om via één of meerdere schrijfoperaties invloed uit te oefenen op het betreffende programma. Ter illustratie hiervan gaan we eens wat spelen met het optelprogramma op pagina 63 van boek 1:

```
Ø1ØØ 18          CLC
Ø1Ø1 A9 13       LDA #13
Ø1Ø3 69 Ø8       ADC #Ø8
Ø1Ø5 ØØ          BRK
1A7A ØØ          NMI-sprongvektor
1A7B 1C
1A7E ØØ          IRQ-sprongvektor
1A7F IC
(STEP : ON)
AD Ø 1 Ø Ø GO Ø1Ø1 A9
GO            Ø1Ø3 69
AD Ø Ø F 1    ØØF1 24   (dus: C=Ø)
DA 2 5        ØØF1 25   (dus: C=1)
PC            Ø1Ø3 69
GO            Ø1Ø5 ØØ
GO            Ø1Ø7 XX
AD Ø Ø F 3    ØØF3 1C
```

Vlak voor de uitvoering van de instruktie ADC is de carry-vlag 1 gemaakt (het carry-bit is het meest rechtse bit van het P-register, dat in ØØF1 wordt gered). Hierdoor wordt het resultaat van de optelling één hoger dan zonder de schrijfoperatie het geval zou zijn; 1C in plaats van 1B.

Punt 15: BRK-remweg

In programma's die worden afgesloten of onderbroken met een BRK toont het display na afloop een adres dat meestal 2 hoger is dan het adres waarop de BRK staat. Mits natuurlijk de IRQ-sprongvektor naar de monitor wijst! Deze BRK-remweg is verklaard op pagina 199 van boek 2. Er kan handig gebruik van worden gemaakt. Bijvoorbeeld bij het oriëntatieprogramma op pagina 58 van boek 1 (derde druk):

```
Ø2ØØ 18          CLC
Ø2Ø1 A9 Ø3       LDA #Ø3
Ø2Ø3 69 Ø7       ADC #Ø7
Ø2Ø5 8D ØA Ø2    STA Ø2ØA
Ø2Ø8 ØØ          BRK
1A7E ØØ          IRQ-sprongvektor
1A7F 1C
```

Het resultaat van de optelling wordt op het adres gezet dat na afloop van het programma te zien is: twee plaatsen vanaf Ø2Ø8 ligt Ø2ØA.

Literatuur:

1. *Junior-computer; Elektuur, maart 1980, p. 3-26ff. Het kennismakingsartikel.*
2. *Junior-monitor; Elektuur, april 1980, p. 4-72.*
3. *Nogmaals: junior-computer; Elektuur, mei 1980, p. 5-66. Diverse zaken.*
4. *Elektuur µP-systemen; Elektuur, juni 1980, p. 6-56ff. Een overzicht van de mogelijkheden voor zelfbouw.*
5. *RAM/EPROM-kaart; Elektuur, september 1980, p. 9-48ff. Universele geheugenuitbreiding.*
6. *Meer junior-geheugen; Elektuur, oktober 1980, p. 10-24ff. De aansluiting van de RAM/EPROM-kaart (lit. 5) op de junior-computer.*
7. *Junior Computer 1 — een volwassen computer voor beginners; Uitgeverij Elektuur B.V. ISBN 90 70160 15 3. Derde druk, december 1980.*
8. *Junior Computer 2 — een volwassen computer voor beginners; Uitgeverij Elektuur B.V. ISBN 90 70160 16 1. Eerste druk, augustus 1980.*
9. *Junior Computer 3 — een volwassen computer voor beginners; Uitgeverij Elektuur B.V. Even geduld a.u.b.!*
10. *Junior groeit!; Elektuur, november 1980, p. 11-77ff. De toekomst op korte en langere termijn.*
11. *Junior-tekst; Elektuur, februari 1981, p. 2-54ff. Weer eens iets anders dan al die hexadecimale tekens.*

het lek van elektuur

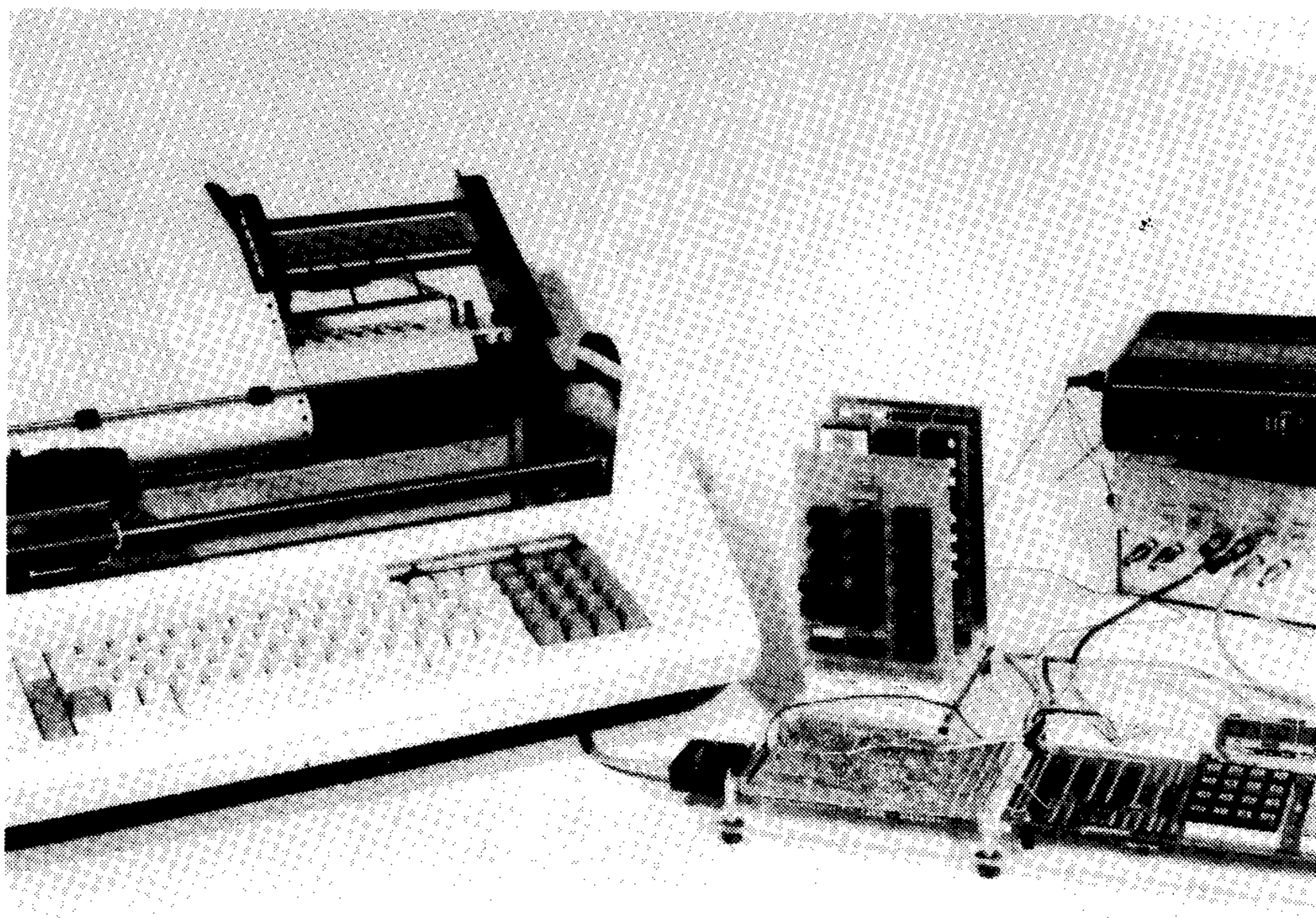
Highcompander

Na overleg met de firma Telefunken zijn er nog enkele wijzigingen gekomen in de schema's van de highcompander (februari 1981, blz. 2-60). Oplettende lezers zullen ook wel gezien hebben dat de komponenten-opdruk van de basisprint niet juist was. Helaas was het niet meer mogelijk dit alles nog op tijd te veranderen.

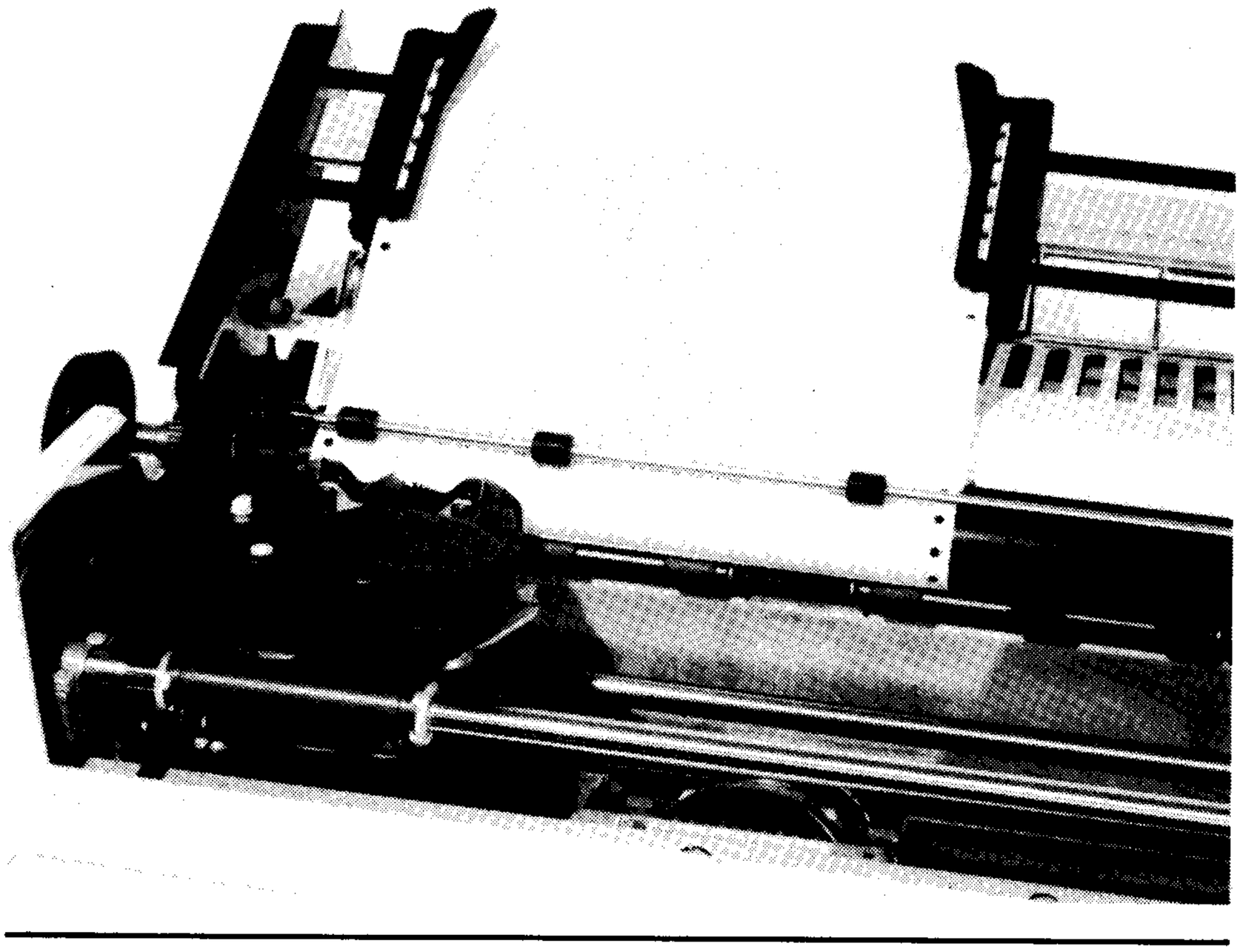
De basisprint die met de twee opsteekmodules wordt aangeboden in de ESS is aangepast aan de nieuwe schema's. De wijzigingen worden meegeleverd. Bij dit pakket wordt ook nog een meet-cassette geleverd met onder andere een testtoon voor het kontroleren van de afregeling.

Lezers die zelf de basisprint willen maken krijgen de gewijzigde schema's en printlayout op verzoek toegestuurd.

frekwentie van het ingangssignaal en biedt na vergelijking met een vaste spanning de mogelijkheid om logisch onderscheid te maken tussen frekwenties : "1" voor de ene frekwentie, "0" voor de andere, geheel in overeenstemming met het principe van FSK (Frequency Shift Keying; zeg maar: frekwentieverschuivingslogica).
De PLL wordt uit +12 V gevoed via de parallelschakeling van D2 + D3 en C8. Dit wat merkwaardige netwerk zorgt ervoor dat de PLL via pen 10 behoorlijk verschoond blijft van allerlei stoorspanningen op de voeding, die de werking van de PLL zouden kunnen verstoren. De "schone 12 V" (ca. 11 V vanwege de dioden) wordt eveneens gebruikt voor de gelijkspanningsinstelling van de ingangen (pennen 2 en 3 van IC6) via R21 . . . R24. Bij de KIM wordt daarvoor een 5 V-voedingsspanning gebruikt die vergeven is van stoorspanningen die regelrecht de (weliswaar gebalanceerde) ingangen ingaan. En dat terwijl er daar óók zo'n "schone 12 V" voorhanden is! Verder is er hier, anders dan bij de KIM, geen sprake van een ca. tienvoudige verzwakking van het ingangssignaal voordat pen 2 is bereikt.
Alle genoemde verbeteringen hebben tot gevolg dat men van goeden huize moet komen om nu ingangsdata de mist in te laten gaan (bijvoorbeeld door "dropouts" op de band, een vuile of niet goed afgestelde kop, enzovoorts). Het is ons al een paar keer overkomen dat *achteraf* bleek dat er data werd ingelezen, terwijl een massaverbinding met de recorder ontbrak!
Van de ingangen 2 en 3 (een soort verschilversterkeringang) wordt alleen ingang 2 gebruikt (asymmetrische sturing). De koppeling met J1 gaat via C1, die een veel kleinere waarde heeft dan de KIM teneinde veel meer storing



beneden 2400 Hz buiten de deur te houden. Weerstand R37 is nodig in het geval dat gebruik wordt gemaakt van de luidsprekeruitgang (of koptelefoonuitgang) van een cassette-recorder, waarbij de luidspreker automatisch wordt afgekoppeld.
Van invloed op de VCO-frekwentie zonder ingangssignaal zijn C3, R25 en P1. De metode, volgens welke P1 wordt ingesteld bepaalt in grote mate de kwaliteit en betrouwbaarheid waarmee data wordt gelezen van de band. In boek 3 zullen verschillende metoden worden gegeven voor de juiste instelling/afregeling van P1.
Punt 7 van IC6 is de uitgang van de PLL. Deze voert de al eerder behandelde regelspanning die ervoor zorgt dat de VCO-frekwentie in de pas loopt met het ingangssignaal. Dit punt is via C9 verbonden met +12 V. Deze kondensator vormt samen met een weerstand van 3K6 in het IC het zogenaamde loopfilter (lusfilter). Dit filter is van groot belang voor het goed, niet al te "emotioneel" (overshoot) reageren van de PLL op frekwentieveranderingen aan de ingang. De uitgang van de PLL gaat via het ladderfilter R26...R28 & C5...C7 naar de min-ingang van de komparator (= vergelijker) IC7. De plus-ingang hiervan ligt via R29 aan een door IC6 geleverde vaste gelijkspanning (op pen 6).
De dimensionering van dat ladderfilter hangt samen met de snelheid waarmee de 3,6 kHz- en 2,4 kHz-frekwenties elkaar opvolgen. En dat hangt op zijn beurt weer samen met de snelheid waarmee de bits op de band zijn geschreven, en van de band moeten worden gelezen. Dat is de zogenaamde baudrate, het aantal verzonden of ontvangen bits per sekonde: 800 (baud) voor de junior-cassette-hardware *en* -software.
We herinneren ons nog wel dat een verhoging van de VCO-frekwentie tot 3600 Hz een verlaging van de uitgangsspanning op punt 7 tot gevolg had (in een gesloten regellus zijn oorzaak en gevolg moeilijk van elkaar te onderscheiden), en een verlaging naar 2400 Hz een verhoging van die spanning. Er van uitgaand dat de gefilterde uitgangsspanning op de min-ingang van IC7 van lager resp. hoger is dan de gelijkspanning op de plus-ingang van IC7, krijgen we de situatie dat de uitgang van IC7 "hoog" is (+5 V, via R31) als de ingangsfrekwentie 3600 Hz is, en "laag" (praktisch op aardpotentiaal) als de ingangsfrekwentie 2400 Hz is. Dat is precies wat



junior-hardware kompleet

In dit artikel worden de laatste stenen aangedragen waarmee, samen met al eerder door Elektuur gepubliceerde computerbouwstenen, het junior-computerbouwwerk kan worden voltooid. De interface-kaart speelt daarbij een sleutelrol als verbindingsschakel met een video-display plus ASCII-toetsenbord (elekterminal), als drager van extra RAM- en EPROM-geheugen en van extra I/O, als begin van een gebufferde bus voor de aansluiting van (een) geheugenkaart(en) en last but not least als data-doorgever van en naar cassette-band.
Dit artikel is een korte vooruitblik op een paar honderd bladzijden boek 3 (verschijnt binnenkort, in twee delen). Kort, dus het verhaal kan niet anders dan onvolledig zijn. Dat neemt niet weg dat het verhaal gaat over de nu volledige hardware.

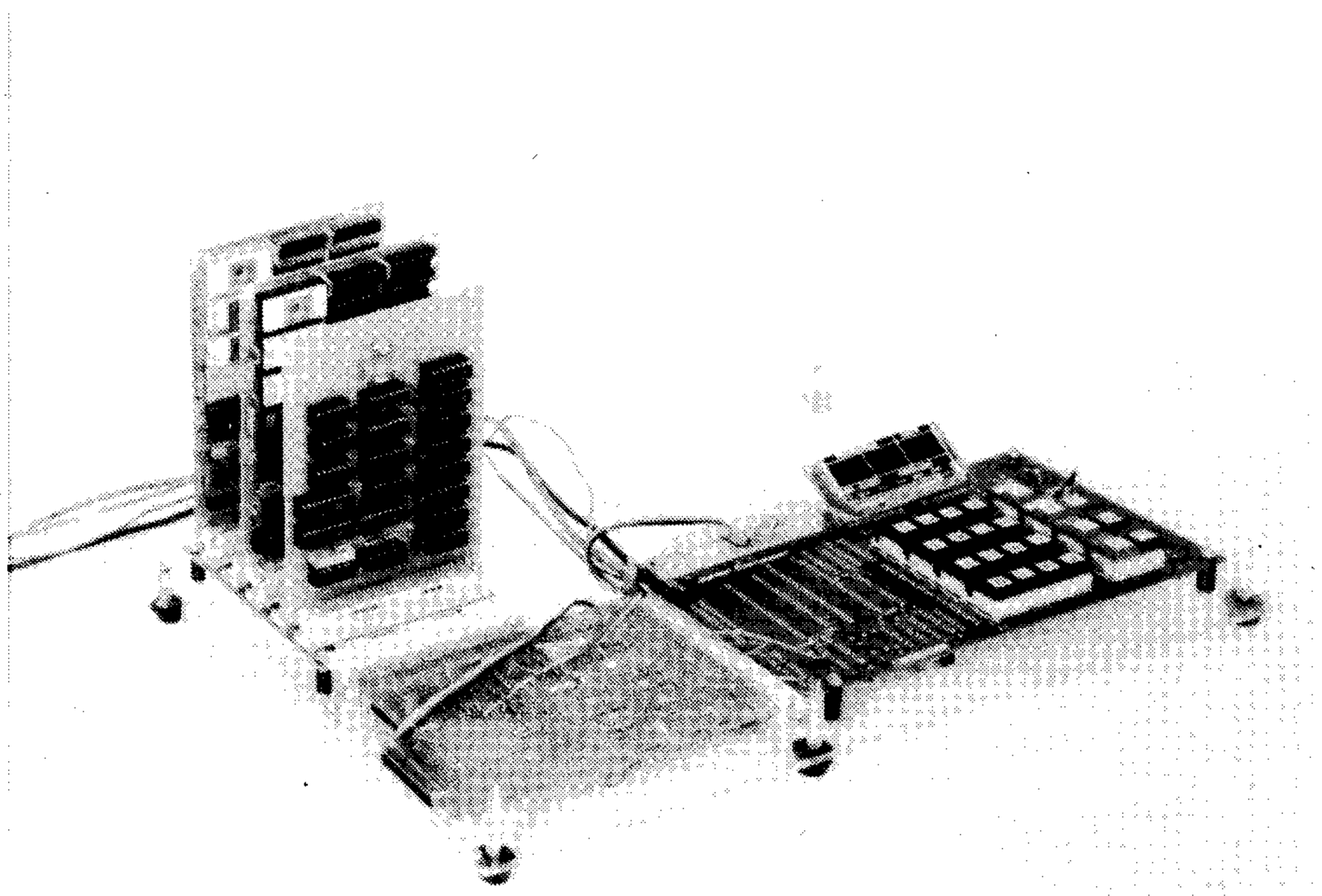
resterende soldeerpunten

Er is ooit een vergelijking gemaakt tussen mens en computer: de hardware is het vlees en bloed, de software de geest. Vanaf een zekere leeftijd is de lichamelijke groei van de mens voltooid; de geestelijke groei gaat gewoon door. De junior-computer is nu volgroeid; alle nodige hardware is beschikbaar. De software kan verder groeien. Vergeten we even gebruikersprogramma's in machinetaal, en ook allerlei systeemsoftware (zoals de twee "software-hulpstukken" die in een ander artikel in dit nummer aan bod komen), dan is natuurlijk de hamvraag: hoe zit het met een hogere programmeertaal? Het staat wel vast dat de voorkeur uitgaat naar BASIC. Het staat ook wel vast dat, als het feest doorgaat (hoe groot is uw belangstelling eigenlijk?), zo'n BASIC (bijvoorbeeld 8K-BASIC op cassette in kombinatie met 16K-RAM) zal zijn gebaseerd op bestaande, niet door Elektuur ontworpen of te ontwerpen software. U gelieve hierover hardop na te denken; individueel of in gebruikersklubverband.
(Een 16K-dynamische-RAM-kaart staat hoe dan ook op stapel. Geen wezenlijke hardware-uitbreiding, maar een goedkoop alternatief voor twee RAM/EPROM-kaarten met elk 8K-RAM).
Maar alles op zijn tijd. Terug naar wat nu aan de orde is. Welke aanvullende hardware is er (nodig)? En hoe ziet het systeem er dan uit? Of liever gezegd: kan er uitzien? Want er "moet" niets.

Bus, kaart en buskaart

Vooral voor amateurgebruik bedoelde computersystemen zijn er in twee soorten: bussystemen en "single board"-systemen. Bij een bussysteem is elk onderdeel op een kaart (= print) ondergebracht. De kaarten zijn onderling even groot (bijvoorbeeld eurokaart: 100 x 160 mm) en doorverbonden via een "bus", dat is een raamwerk van leigingen waarbij punten met gelijke positie (bijvoorbeeld konnektorpennen) onderling zijn doorverbonden. De praktische uitvoering is soms via draadverbindingen, soms via een print: de buskaart. De lengte van de bus is *in principe* onbeperkt. Het bussysteem is universeel; het kan (gaandeweg) zeer uitgebreid worden met als gevolg een computer met vele gebruiksmogelijkheden.
Bij een "single board"-systeem is alles op één kaart gezet. Het hangt er maar helemaal van af wat er op zit (en dus hoe groot de kaart is) als het gaat om de vraag wat de gebruikersmogelijkheden ervan zijn. Bij de behoefte aan uitbreidingen is dat "alles" op één kaart" vaak een blok aan het been. Men heeft verkeerd gegokt bij de aanschaf: dat krijg je als je alles op één kaart zet.
"Single board" wordt vooral toegepast bij oefen/leersystemen en bij relatief eenvoudige procesbesturingen (waarbij de computer permanent is ingeschakeld). In zijn standaardvorm is de junior-computer een "single board"-computer. En bedoeld voor de al genoemde toepassingen. Duizenden gebruiken hem zo. Er is kennelijk sprake van een "Campari"-effekt: je hoort en leest zo veel over "chips" en "computers" dat het interessant wordt om er zelf iets mee te gaan doen, al is het maar één keer in je leven. Vele applikaties (gebruik van de poorten) van de junior-computer zijn ons bekend, variërend van A/D-omzetters tot de sturing van een opdampproces bij de halfgeleiderfabrikage. Er is niets op tegen om het te te houden bij de (legio) mogelijkheden van de standaard-junior-computer en gewoon te vergeten dat er een uitbreidingskonnektor is.
Toch is die uitbreidingskonnektor er niet zo maar. Hij biedt de mogelijkheid om de meest voor de hand liggende (lees: de meest noodzakelijke) uitbreidingen te realiseren, te weten: voorzieningen voor het gebruik van de magnetische geluidsband als achtergrondgeheugen (zodat een RAM met geheugenverlies geen RAMp meer hoeft te betekenen), de mogelijkheid om de maximale hoeveelheid geheugen (64K) te gebruiken en alles wat nodig is om randapparatuur zoals bijvoorbeeld een video-terminal aan te sluiten. Eventuele extra I/O is mooi meegenomen.
Zodra de uitbreidingskonnektor wordt gebruikt is de junior-computer niet meer "single board". Wil dat zeggen dat het nu zo'n in meerdere opzichten "eindeloos" bussysteem gaat worden? Nee. Er is al iets gezegd over afrondende hardware. Dat is iets anders dan "eindeloos". Verder lijkt de fysieke opzet van het uitgebreide systeem helemáál niet op een zuiver bussysteem. Al de reeds genoemde uitbreidingen (op het grootste deel van de geheugenuitbreiding na) zitten op de interface-kaart, die net zo groot is als de basiskaart (hoofdprint) van de junior-computer, en daaronder wordt geplaatst. Aan de interface-kaart

kan een bestaande SC/MP-buskaart worden gekoppeld, die als drager dient voor één of meerdere geheugenkaarten.
Ondanks de mogelijke aanwezigheid (mogelijk, want hij kan achterwege blijven) van een buskaart is de uitgebreide junior-computer geen bussysteem. Maar wat dan wèl? Gezien de sandwich-konstruktie van interface-kaart en basiskaart kun je misschien nog het beste spreken van "double single board".
Nòg belangrijker dan de vraag welke naam het beestje moet krijgen is de beschrijving van het beestje. En wel nu.



De interface-kaart

Het woord "interface" betekent koppeling, aanpassing. In ons geval de koppeling van de junior-computer met de buitenwereld via extra I/O, de koppeling met de magnetische band (cassette-interface), de koppeling met randapparatuur (RS-232-interface) plus een interne interface van en naar de standaardkaart via buffering.
Alle elektronica op de interface-kaart is terug te vinden in de figuren 1 en 2. Die elektronica is omvangrijker dan die van de basiskaart en zit toch op een print met gelijke afmetingen. Dat is geen eenvoudige zaak geweest, neem dat van ons aan. De diverse onderdelen zullen nu stuk voor stuk worden besproken.

Buffering: tegen een stootje kunnen

Links in figuur 1 zien we alle aansluitingen van de ingangskonnektor (INPUT CONNECTOR). Via deze konnektor en via vijf verbindingen met de poortkonnektor (zie figuur 2) verloopt de kommunikatie met de standaard-junior-computer. Op de leidingen EX en K1 . . . K6 na — die zijn uitsluitend van belang voor de interface-kaart — vinden we alle leidingen terug bij de uitgangskonnektor (OUTPUT CONNECTOR), rechtsboven in figuur 1. De uitgangskonnektor dient voor de koppeling met de buskaart, die op zijn beurt zorg draagt voor de koppeling met één of meer geheugenkaarten.
Adreslijnen A zien we terug als AB en datalijnen D als DB. De B staat voor: gebufferd. Waarom buffering? Om twee redenen. Ten eerste is buffering noodzakelijk om elektrische overbelasting als gevolg van een te groot aantal aansluitingen te voorkomen. Ten tweede omdat het in het geval van de datalijnen nu mogelijk is om de datastroom tussen de standaard-junior-computer en alles daarbuiten (interface-kaart, één of meer geheugenkaarten) te blokkeren als dat wenselijk is.
De adresbuffers zijn N1 . . . N16. Omdat adressering altijd vanuit de 6502 op de standaardkaart plaatsvindt, staan de driehoeken in de aangegeven richting: ingang links, uitgang rechts. De databuffers zijn N17 . . . N32. Hier is sprake van een even genummerde groep buffers met ingang rechts, uitgang links, en een groep oneven genummerde buffers met ingang links, uitgang rechts. Er zijn drie situaties mogelijk:

1. De oneven genummerde poorten N17 . . . N32 zijn aktief, de even genummerde inaktief. Dat is het geval als $\overline{\text{WRITE}}$ *en* READ "0" zijn. Er is een datastroom mogelijk van de standaardkaart naar de uitbreidingen. Dat is het geval bij het schrijven van data in *niet*-standaardgeheugen of —I/O.
2. De oneven genummerde poorten N17 . . . N32 zijn inaktief ("tri-state" oftewel hoogohmig aan ingang en uitgang), de even genummerde aktief. Dat is zo als $\overline{\text{WRITE}}$ en READ "1" zijn. Er is een datastroom mogelijk van de uitbreidingen naar de standaardkaart. Dat is het geval bij het lezen van data uit niet-standaardgeheugen of —I/O.
3. Alle poorten N17 . . . N32 zijn inaktief, als $\overline{\text{WRITE}}$ "1" is en READ "0". De datastroom is geblokkeerd. Deze situatie is relevant bij het lezen of schrijven van data op standaardgeheugen of —I/O. De signalen $\overline{\text{WRITE}}$ en READ komen uit de PROM IC17. Daarover later meer.

N.B. De vierde mogelijkheid:
$\overline{\text{WRITE}}$ = "0" en READ = "1" houdt in dat alle poorten N17 . . . N32 tegelijkertijd aktief zijn. Deze situatie is ongewenst.

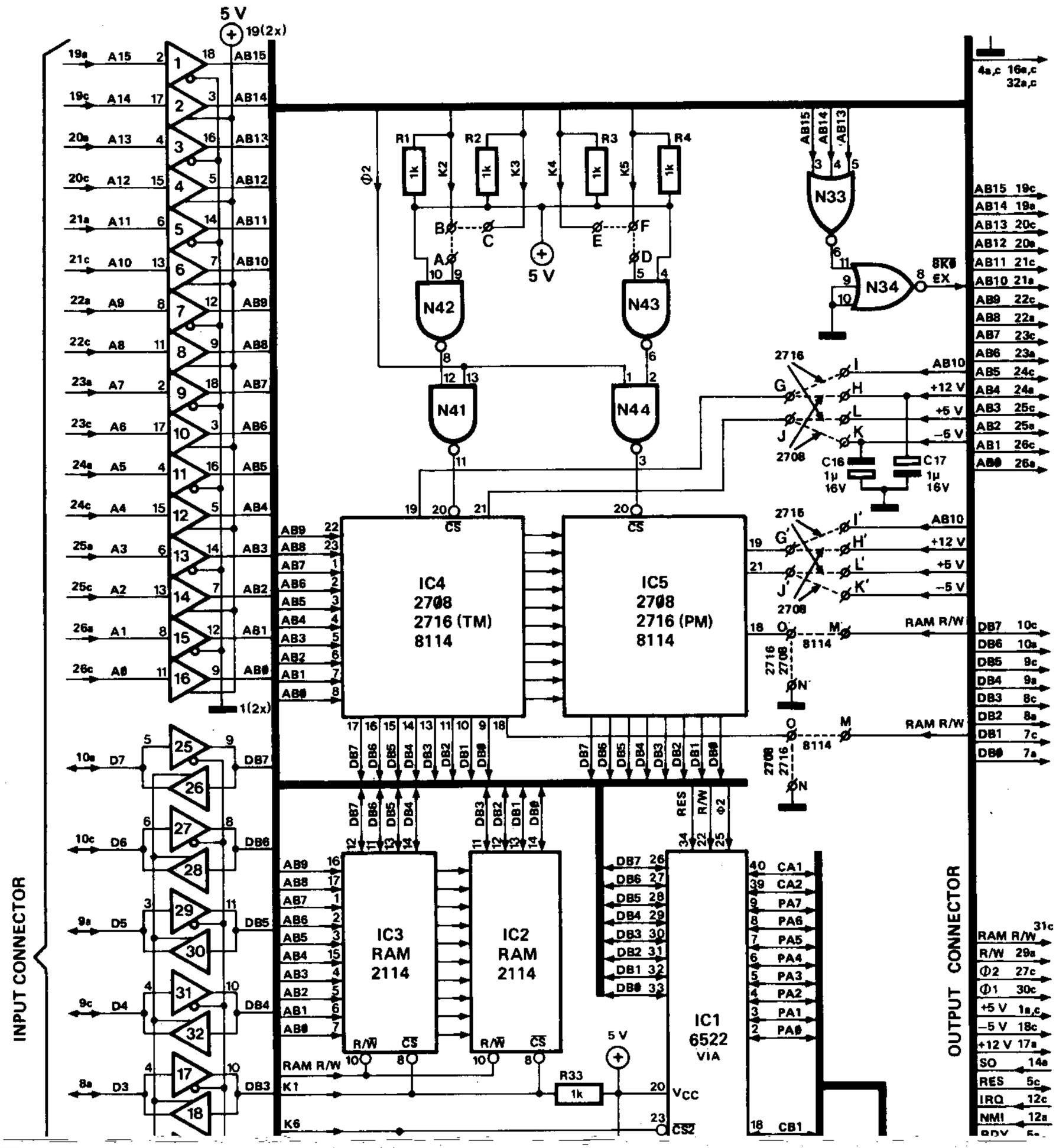
Extra I/O

De VIA IC1, van het type 6522, is een hoofdstuk apart. En wel een hoofdstuk van de tweede helft van boek 3. Hier zij opgemerkt dat deze "Versatile Interface Adapter" nog meer mogelijkheden in huis heeft dan de standaard-I/O-bouwsteen (PIA) van het 6532. De relevante verbindingen met de buitenwereld zijn te vinden bij de VIA "CONNECTOR" in figuur 1 (is geen echter konnektor, vandaar de aanhalingstekens). De 6522 wordt gestuurd vanuit de adreslijnen AB0 . . . AB3 en met diverse signalen van de stuurbus. De aanwezigheid van acht datalijnen zal ook niemand verbazen, evenals het uitgangssignaal IRQ (interrupt na de time out van een timer).
Evenals de 6532 heeft de 6522 twee chipselect-signalen: $\overline{\text{CS2}}$ is net zoals de $\overline{\text{CS2}}$ van de 6532 verbonden met het uitgangssignaal K6 van de adresdekoder IC6 op de standaardkaart van de junior-computer; CS1 ligt aan de uitgang van N35, die wordt gestuurd door K6 en AB9. Voor de aktivering van de VIA moet niet alleen $\overline{\text{CS2}}$ "0" zijn, maar ook CS1 "1". Nu is N35 een NOR-poort; de uitgang ervan is "1" als alle ingangen nul zijn, dus als AB9 en K6 nul zijn (voor de 6532 moet A9 juist "1" zijn). Als we verder bedenken dat A8 (AB8) niet is aangesloten op de 6522 en ook niet op de 6532, dan geldt voor de VIA 6522 het volgende adresbereik:
1800 = 1900 . . . 18FF = 19FF
(AB8 = X; AB9 = "0"; K6 = "1")
en voor de PIA 6532 het adresbereik:
1A00 = 1B00 . . . 1AFF = 1BFF
(A8 = X; A9 = "1"; K6 = "1")
Sluiten we de dubbeladressering vanwege A8 = AB8 = X uit, dan zijn er zoveel voor de PIA als voor de VIA 256 adressen beschikbaar. Net als bij de 6532-PIA (19 verschillende "geheugenplaatsen" die meervoudig zijn geadresseerd) zijn bij de VIA niet op al die 256 adressen 256 verschillende geheugenplaatsen aangesloten. Zoals men ziet in figuur 1 zijn de adreslijnen AB4 . . . AB7 niet aangesloten. Er zijn dus 16 verschillende VIA-geheugenplaatsen.

Extra geheugen

Op de interface-kaart is plaats voor extra geheugen. En wel zo veel (maxi-

1


5 V
19(2x)
INPUT CONNECTOR
OUTPUT CONNECTOR
A15
A14
A13
A12
A11
A10
A9
A8
A7
A6
A5
A4
A3
A2
A1
A0
AB15
AB14
AB13
AB12
AB11
AB10
AB9
AB8
AB7
AB6
AB5
AB4
AB3
AB2
AB1
AB0
1(2x)
D7
D6
D5
D4
D3
DB7
DB6
DB5
DB4
DB3
R1
R2
R3
R4
1k
K2
K3
K4
K5
Φ2
A
B
C
E
F
D
N42
N43
N41
N44
N33
N34
8K0
EX
4a,c 16a,c 32a,c
2716
2708
I
H
L
K
G
J
C16
1µ
16V
C17
1µ
16V
+12 V
+5 V
–5 V
IC4
2708
2716 (TM)
8114
IC5
2708
2716 (PM)
8114
CS
RAM R/W
M
O
N
IC3
RAM
2114
IC2
RAM
2114
R/W
CS
K1
K6
R33
1k
Vcc
CS2
RES
R/W
IC1
6522
VIA
CA1
CA2
PA7
PA6
PA5
PA4
PA3
PA2
PA1
PA0
CB1
RAM R/W
R/W
Φ2
Φ1
SO
IRQ
NMI

van buskaartgeheugen). Als dat allemaal niet het geval is moeten de vektoren uit de standaard-EPROM komen. Dat gebeurt net als vroeger door de zaak te beduvelen: isoleer de databuffers en de adreslijnen A15, A14 en A13 zijn net als vroeger niet van belang omdat ze nergens zijn aangesloten op de basis-kaart.

N.B. Zodra er buskaartgeheugen is aangesloten moet er (onder meer) op pagina FF EPROM zijn aangesloten met gespecificeerde vektor-data. Er is dan geen keuze mogelijk tussen vektoren ophalen in pagina FF en vektoren ophalen in pagina 1F. Dat laatste zou namelijk isolatie van de buffers betekenen, en dan heeft bus-kaartgeheugen geen zin.

Terug naar de adressering van de PROM IC17. De vijf adressignalen, die via $\overline{\text{WRITE}}$ en READ invloed uitoefenen op de toestand van de databuffers zijn:

A. Het signaal R/W of eigenlijk R/$\overline{\text{W}}$ (is niet konsekwent zo gedaan bij al het geschrevene over de junior-computer; het streepje betekent: aktief indien logisch nul). Hangt samen met de vraag of er sprake is van lezen of schrijven. Vrij logisch dat-ie erbij hoort.

B. Het signaal VIA. Komt uit N35 en is tevens de CS1 van de VIA IC1. Is "1" als K6 en AB9 nul zijn. Dus als de VIA is geselekteerd. Van het 1K-gebied met K6 = "Ø" zit de PIA vóór de databuffers en de VIA erachter.

C. Het signaal $\overline{\text{KX}}$. Dit is afkomstig uit N36, die de uitgang van N40 akti-veert. Is nul als één van de leidingen K1 . . . K5 nul is, dus als er geheugen op de interface-kaart aan bod is.

D. Het signaal $\overline{\text{8KØ}}$ is gelijk aan EX (eerder al behandeld) en is nul als de eerste 8K wordt gedresseerd (geheugen en I/O op de basiskaart en op de inter-face-kaart) en "1" als buskaartgeheugen is geadresseerd.

E. Punt E van IC17 is of met +5 V verbonden (draadbrug RS), of met massa (draadbrug RT). Punt E *moet* nul zijn als er geen buskaartgeheugen is aangesloten (vandaar het bijschrift $\overline{\text{WITH}}$, hetzelfde als WITHOUT) en één als dat wèl het geval is (bijschrift WITH, hetzelfde als $\overline{\text{WITHOUT}}$). Waarom hier een draadbrug RS of RT? Wel, heel simpel omdat u dan de vrijheid heeft om buskaartgeheugen aan te sluiten of niet. U ziet het: niets moet, alles mag.

In tabel 1 is de inhoud van de PROM IC17 overzichtelijk samengevat (ESS508). De bits Y3 . . . Y8 zijn nul en spelen verder geen enkele rol. In prin-cipe zijn er 32 verschillende situaties mogelijk die leiden tot een keuze uit één van de drie toegestane kombinaties van READ & $\overline{\text{WRITE}}$. In werkelijkheid zijn het er veel minder. Afgezien van de opdeling in lezen en schrijven blijven er acht situaties over:

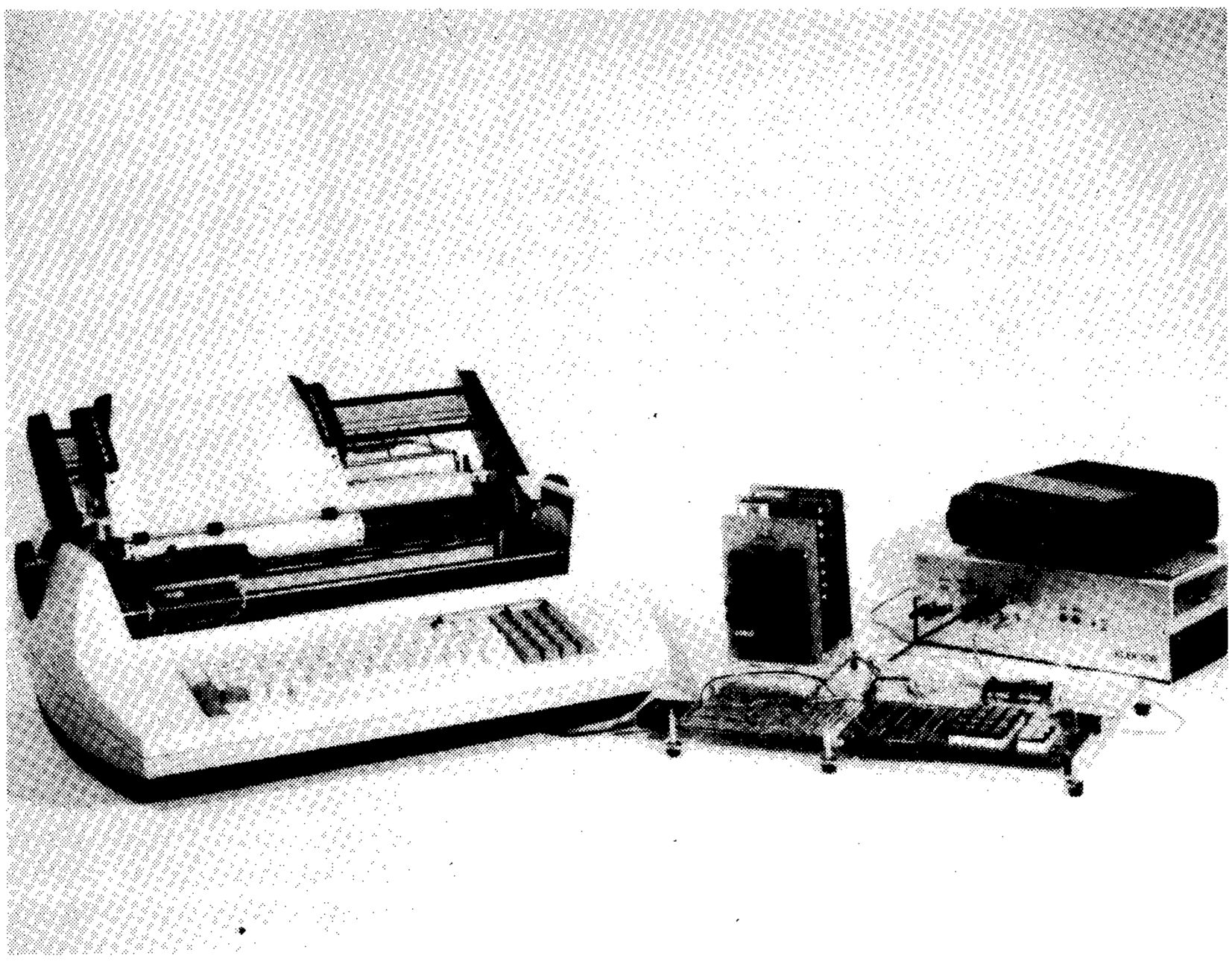
1. Schrijven of lezen van geheugen op de interface-kaart (IC2 . . . IC5).

Acht databuffers staan in de door lezen of schrijven bepaalde richting ("enabled") (PROM-adressen ØØ en Ø1).

tabel 1

PROM-adres (hex)	E = WITH of $\overline{\text{WITH}}$	D = $\overline{\text{8KØ}}$	C = $\overline{\text{KX}}$	B = VIA	A = R/W	Y8	Y7	Y6	Y5	Y4	Y3	Y2 = READ	Y1 = $\overline{\text{WRITE}}$	PROM-data (hex)
ØØ	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
Ø1	Ø	Ø	Ø	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	1	1	Ø3
Ø2	Ø	Ø	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
Ø3	Ø	Ø	Ø	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
Ø4	Ø	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	1	Ø1
Ø5	Ø	Ø	1	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	1	Ø1
Ø6	Ø	Ø	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
Ø7	Ø	Ø	1	1	1	Ø	Ø	Ø	Ø	Ø	Ø	1	1	Ø3
Ø8	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
Ø9	Ø	1	Ø	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
ØA	Ø	1	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
ØB	Ø	1	Ø	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
ØC	Ø	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	1	Ø1
ØD	Ø	1	1	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	1	Ø1
ØE	Ø	1	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
ØF	Ø	1	1	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
1Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
11	1	Ø	Ø	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	1	1	Ø3
12	1	Ø	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
13	1	Ø	Ø	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
14	1	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	1	Ø1
15	1	Ø	1	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	1	Ø1
16	1	Ø	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
17	1	Ø	1	1	1	Ø	Ø	Ø	Ø	Ø	Ø	1	1	Ø3
18	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
19	1	1	Ø	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
1A	1	1	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
1B	1	1	Ø	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
1C	1	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
1D	1	1	1	Ø	1	Ø	Ø	Ø	Ø	Ø	Ø	1	1	Ø3
1E	1	1	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ
1F	1	1	1	1	1	Ø	Ø	Ø	Ø	Ø	Ø	Ø	Ø	ØØ

Tabel 1. De inhoud van de PROM IC17. Men kan hem door Elektuur (ESS 508) laten programmeren of dat zelf doen (bijvoorbeeld met behulp van schakeling 97 uit de half-geleidergids 1980). Andere, pin-kompatibele PROM's zijn op zich bruikbaar (bijvoorbeeld de 74S188), maar kunnen niet worden geprogrammeerd via ESS 508 of met die schakeling 97.



2. Lezen van EPROM en lezen of schrijven van RAM of PIA op de basiskaart. Databuffers geblokkeerd ("disabled") (PROM-adressen Ø4 en Ø5).
3. Schrijven of lezen van de VIA. Zit op de interface-kaart, dus acht databuffers staan in de door lezen of schrijven gedikteerde richting (PROM-adressen Ø6 en Ø7).
4. Schrijven of lezen van buskaartgeheugen (PROM-adressen ØC en ØD).

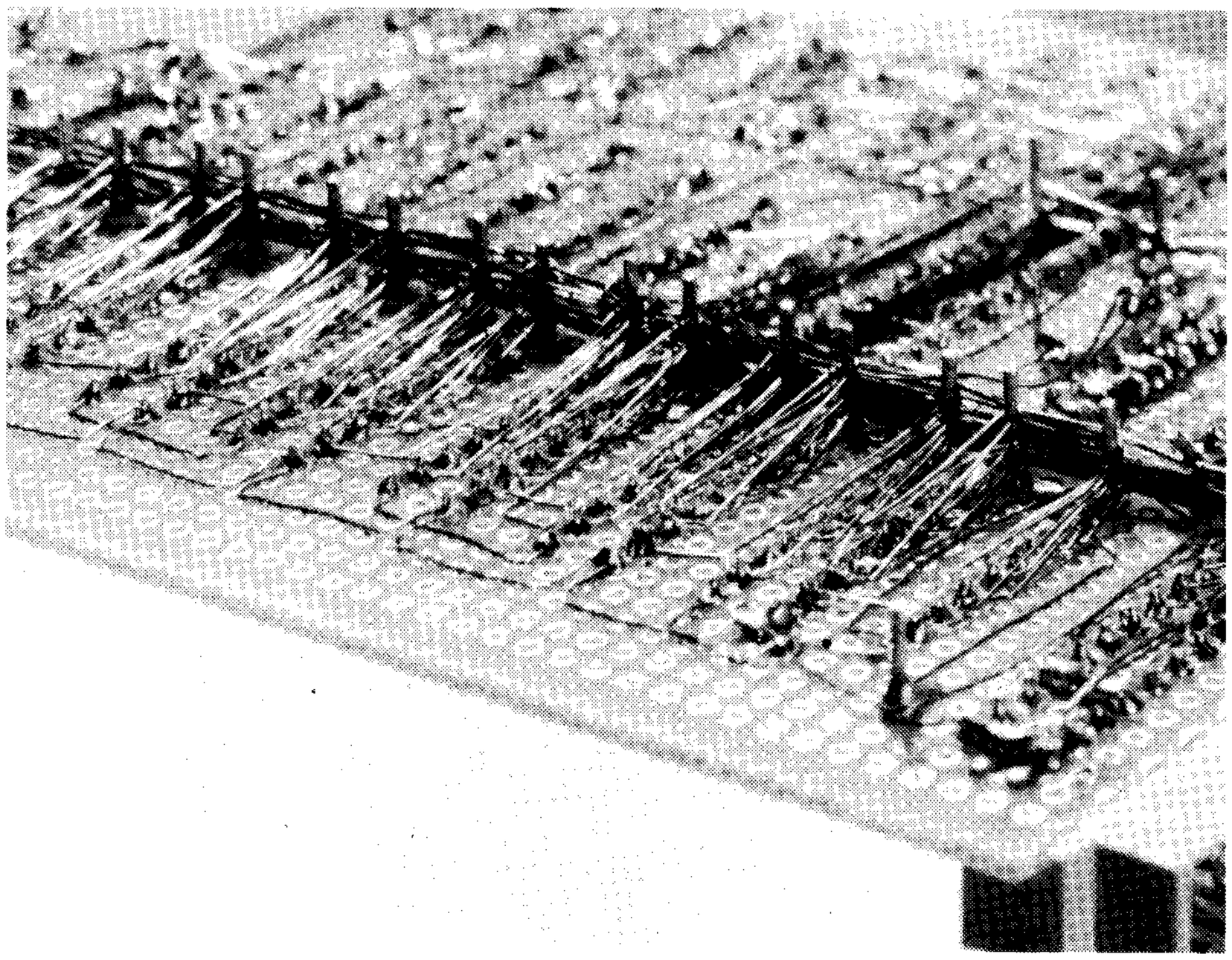
Omdat, evenals in de gevallen 1 . . . 3, E nul is ($\overline{\text{WITH}}$ = WITHOUT), is er kennelijk geen buskaartgeheugen aangesloten; de databuffers *moeten* in dit geval zijn geblokkeerd. De vektoren NMI, RES en IRQ worden automatisch uit de standaard-EPROM van de junior-computer gehaald.

5. Zie 1 (PROM-adressen 1Ø en 11).
6. Zie 2 (PROM-adressen 14 en 15).
7. Zie 3 (PROM-adressen 16 en 17).
8. Voor een deel zie 4 (PROM-adressen 1C en 1D).

Omdat, evenals in de gevallen 5 . . . 7. E "1" is (WITH = $\overline{\text{WITHOUT}}$), is er kennelijk buskaartgeheugen aangesloten. Nu moeten de databuffers wèl open staan voor dataverkeer in een bepaalde richting. Zoals al opgemerkt: de drie vektoren worden nu uit pagina FF opgehaald. Daar moet EPROM op zijn aangesloten, die op FFFA . . . FFFF de drie vektoren specificeert.
We zien dat de helft van de 32 bytes van de PROM IC17 echt nodig is. De andere 16 toestanden zijn onbelangrijk, omdat de overeenkomende kombinaties van D = $\overline{\text{8KØ}}$, C = $\overline{\text{KX}}$ en B = VIA gewoonweg niet voorkomen. Ondanks dat zijn in alle 16 overige gevallen Y1 en Y2 nul. Dat betekent dat de databuffers op schrijven staan ingesteld en dat is volkomen ongevaarlijk.
Figuur 1 is nu helemaal besproken. Nu figuur 2 nog.

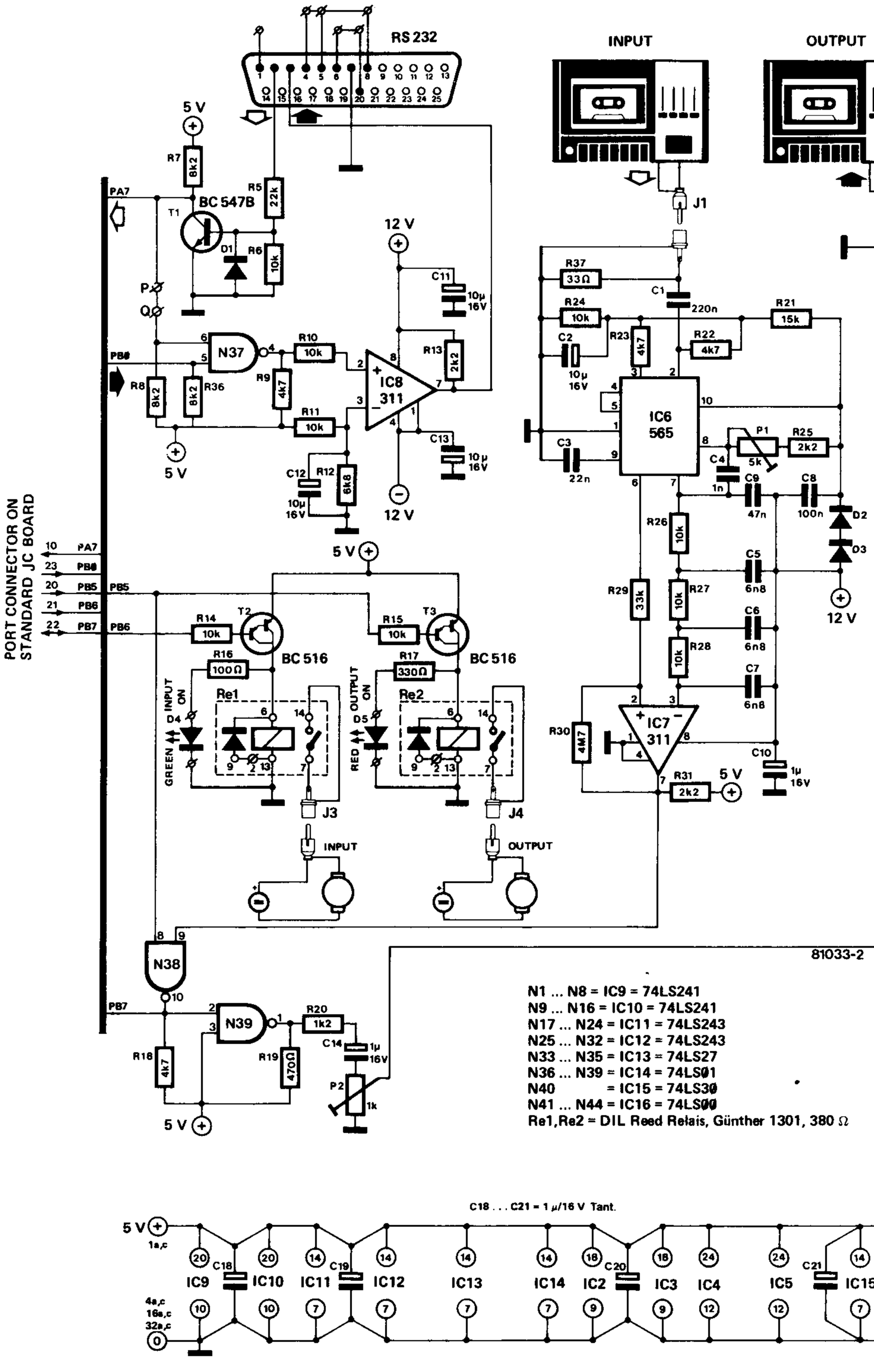
Cassette-hardware

Het grootste gedeelte van figuur 2 betreft elektronica ten behoeve van de cassette-interface: alles wat, afgezien van de software (zie een ander artikel in dit nummer) nodig is voor het zenden van data naar de cassette-recorder (schrijven) of het lezen van data van de cassette-recorder. Tijdens het lezen van data wordt data van de band weergegeven, tijdens het schrijven data op de band opgenomen.
De data gaat naar of komt van de 6502-μP via poortlijn PB7 van de poortkonnektor. Tijdens het schrijven van data, dus tijdens de subroutine DUMP/DUMPT van het systeemprogramma TAPE MANAGEMENT, is PB7 uitgang, net zoals PB5 en PB6. Poort PB5 is dan "0" en PB6 "1". Dat heeft tot gevolg dat de ingang van N38 met pennummer 8 ook laag is en de uitgang van N38 dus hoog (N38 heeft een open-kollektor-uitgang, die hoogohmig is als-ie "1" is). Omdat PB5 "0" is wordt via R15 de PNP-darlington T3 opengestuurd, met als gevolg dat de *rode* LED D5



(OUTPUT ON) oplicht en dat relais Re2 is bekrachtigd; de twee kontakten van uitgang J4 zijn met elkaar doorverbonden. Nemen we deze kontakten op in serie met de motor van de opname-recorder OUTPUT (afstandsbediening), dan kan deze worden gestart via de software (de recorder moet dan wel gereed staan voor opname).
Omdat tijdens het schrijven PB6 "1" is, is T2 gesperd, de groene LED D4 gedoofd, relais Re1 niet bekrachtigd en zijn de kontakten van J3 geopend. De uitgang van N39 voert ons via R20 en C14 naar P2, waarmee het maximale opnamenivo moet worden ingesteld. De data-uitgang is J2.
Tijdens het lezen van data van de band, dus tijdens de subroutine RDTAPE van TAPE MANAGEMENT, is PB7 als ingang geschakeld; ook nu zijn PB5 en PB6 uitgang, Poort PB5 is nu "1" en PB6 "Ø". Met als gevolg dat N38 het uitgangssignaal van IC7 geïnverteerd aan PB7 kan doorgeven. Poort N39 blijft gewoon doorverbonden. Dat betekent dat er ook signalen terechtkomen op de data-uitgang J2, maar dat kan helemaal geen kwaad.
Omdat PB5 "1" is, is de rode LED D5 gedoofd en zijn de kontakten van J4 geopend. Daarentegen is PB6 nu nul, en dat heeft tot gevolg dat T2 geleidt, de *groene* LED D4 (INPUT ON) oplicht, relais Re1 bekrachtigd is en dat de kontakten van J3 (afstandsbesturing van de weergave-recorder INPUT) gesloten zijn.
Men is niet verplicht om gescheiden recorders te nemen voor het opbergen en ophalen van data. Bij het gebruik van één recorder voor beide taken vervalt het nut van de beide relais, omdat afstandsbesturing dat niet zo nodig moet. Het kàn natuurlijk wèl: een kwestie van doorverbinden van de kontakten van J3 en J4 (*beide* relais moeten aanwezig zijn!).
De sturing van T2 en T3 levert in ieder geval nuttige visuele informatie op over wat men eigenlijk aan het doen is. De groene LED licht op tijdens het lezen van data en de rode LED doet dat tijdens het schrijven van data. De kleurkeuze is logisch; denk aan de rode lampen in opname-studio's en elders, en aan groene verkeerslichten.
Het gedeelte van figuur 2 tussen de data-ingang J1 en de ingang van N38 met pennummer 9 is nu aan bespreking toe. De schakeling rond IC6 en IC7 dus. De KIM-kenners onder u zullen ongetwijfeld bij het zien van dit gedeelte van figuur 2 zeggen: "Hé, waar heb ik dat meer gezien". Klopt. Omdat we van mening zijn dat het geen zin heeft om het wiel opnieuw uit te vinden. Wèl zijn de hoekige kanten van het wiel afgerond. Anders gezegd: het gaat om verbeterde KIM-hardware (zie ook schakeling 31 uit de halfgeleidergids 1980).
Voor de bespreking van IC6, IC7 en omstreken is het voldoende om te weten dat de data op de band staat in de vorm van een reeks van een toontje met een frekwentie van ca. 3600 Hz, afgewisseld door eentje met een frekwentie van ca. 2400 Hz. Dus een tijdje 3600 Hz, daarna een tijdje 2400 Hz, daarna weer 3600 Hz, en daarna weer 2400 Hz, enzovoorts. Bij een bit hoort een 3600 Hz-tijdje en een dito 2400 Hz.
Uit de 3600 Hz-tijdsduur ten opzichte van de 2400 Hz-tijdsduur blijkt dat het bit "1" dan wel "0" is. De schakeling rond IC6 en IC7 zorgt ervoor dat de uitgang van IC7 "1" is als er 3600 Hz op de ingang J1 staat en "0" als er sprake is van 2400 Hz. De software van RDTAPE zorgt ervoor dat uit de twee tijdsduren van een 3600 Hz/2400 Hz-cyclus een bit "0" of "1" wordt gedestilleerd (zie het

2



N1 ... N8 = IC9 = 74LS241
N9 ... N16 = IC10 = 74LS241
N17 ... N24 = IC11 = 74LS243
N25 ... N32 = IC12 = 74LS243
N33 ... N35 = IC13 = 74LS27
N36 ... N39 = IC14 = 74LS01
N40 = IC15 = 74LS30
N41 ... N44 = IC16 = 74LS00
Re1,Re2 = DIL Reed Relais, Günther 1301, 380 Ω

C18 . . . C21 = 1 µ/16 V Tant.

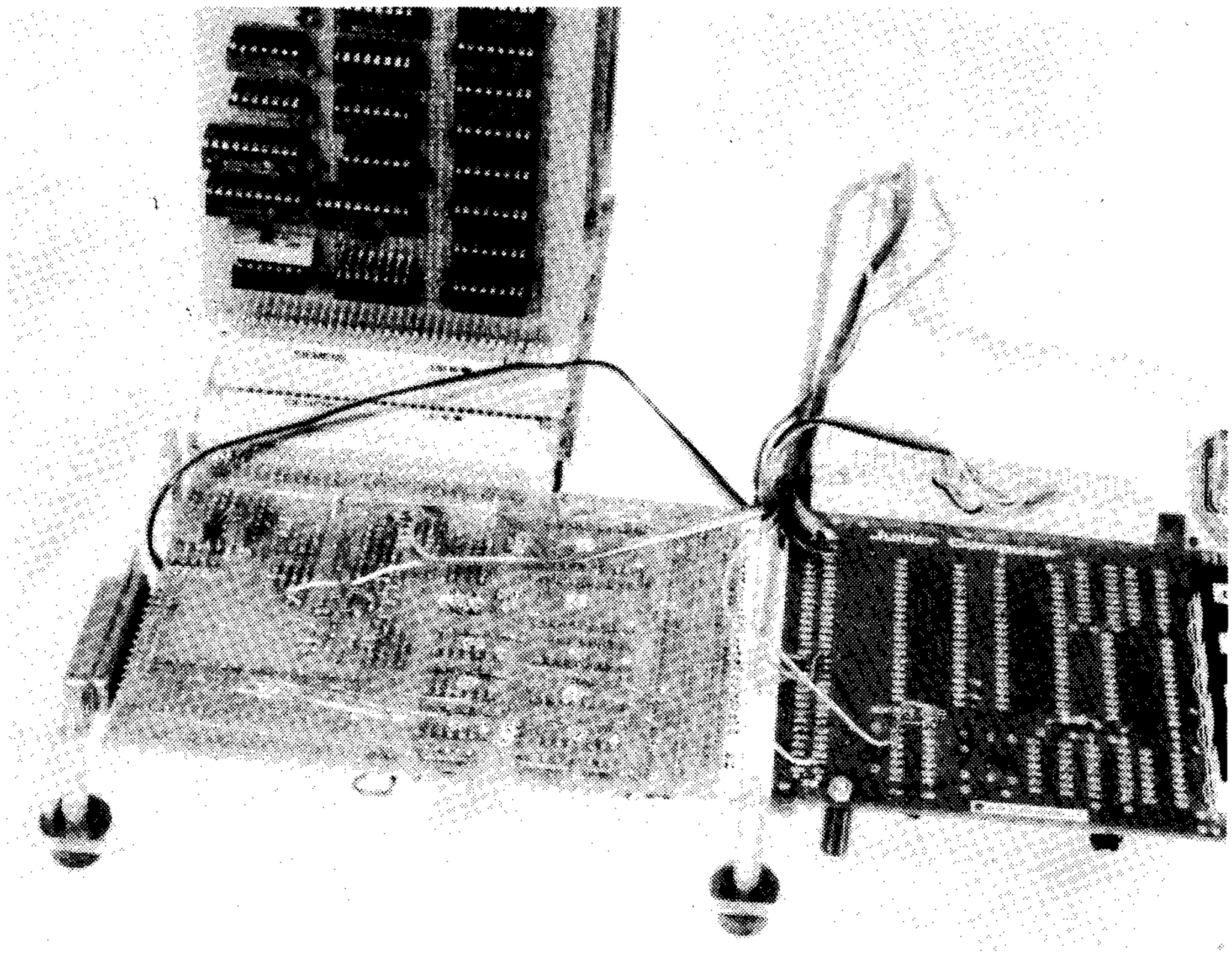
Figuur 2. Deze schakelingen van de interface-kaart hebben betrekking op het zenden of schrijven van data en het ontvangen of lezen van data ten behoeve van RS232-randapparatuur en cassette-band.

software-artikel elders in dit nummer).
De schakeling van IC6 vormt samen met de omringende komponenten een zogenaamde PLL (Phase Locked Loop: fasevergrendelde regellus). Dat klinkt erg "duur" en het is mogelijk om een hele jaargang Elektuur te vullen met formules over dat ding. Neem ons niet kwalijk als we dat niet doen en in plaats daarvan de werking uitleggen in een paar zinnen. Zie de PLL, met name in de toepassing op de interface-kaart als een *"frekwentievolger"*. Net zoals bij een emittervolger de uitgangsspanning de ingangsspanning volgt past een interne oscillator zich in de frekwentie aan aan de frekwentie van het ingangssignaal, binnen een zeker frekwentiebereik en boven een minimum-nivo van het ingangssignaal. Die interne oscillator heeft een frekwentie die afhangt van een stuurspanning (VCO). Zonder ingangssignaal is de frekwentie ca. 3000 Hz, dus precies tussen 2400 Hz en 3600 Hz. Staat er 3600 Hz op de ingang, dan gaat de frekwentie met 600 Hz omhoog; bij 2400 Hz op de ingang met 600 Hz omlaag. Voor een verlaging van de frekwentie is een verhoging van de stuurspanning nodig en voor een frekwentieverhoging een verlaging van de stuurspanning. De hoogte van de stuurspanning hangt dus samen met de

we nodig hebben. De uitgangsspanning van IC7 wordt via N38 geïnverteerd (hoog wordt laag, en omgekeerd) en op PB7 gezet.
N.B. De PLL reageert niet oneindig snel op veranderingen van de ingangsfrekwentie. Dat heeft tot gevolg dat de uitgang van IC7 niet in één keer omklapt van "1" naar "0" of van "0" naar "1" tijdens de overgang van 3600 Hz naar 2400 Hz respektievelijk van 2400 Hz naar 3600 Hz. Nee, de uitgang fietst een aantal keren heen en weer tussen het ene en het andere nivo alvorens definitief het eindnivo aan te nemen: de zogenaamde PLL-jitter, die erg veel lijkt op kontaktdender van (bijvoorbeeld) toetsen. De software van de subroutine RDTAPE weet echter wel raad met dat tijdelijke wispelturige gedrag.

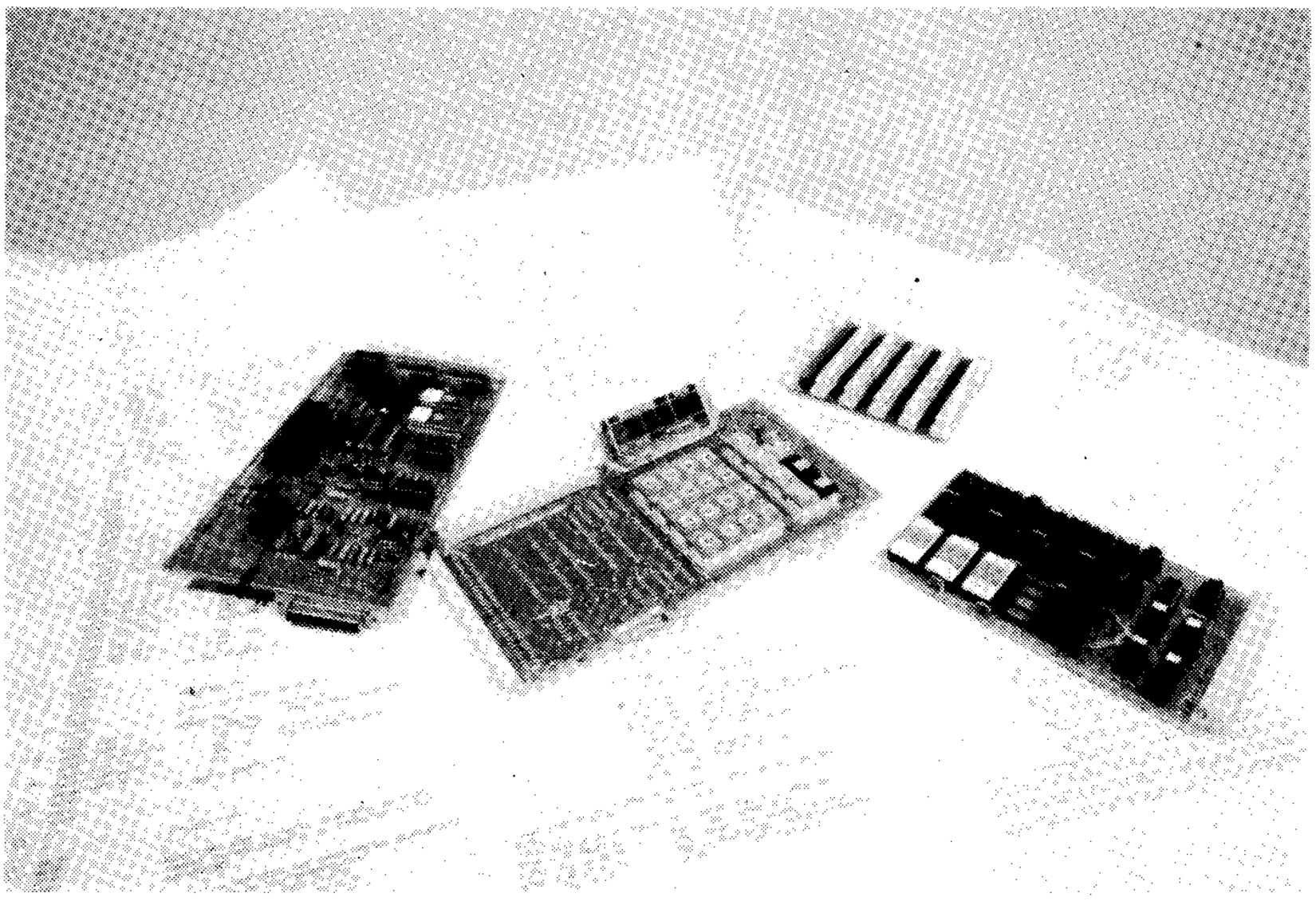
Randapparatuur via RS232C

De ter bespreking resterende schakelingen van figuur 2 (linksboven) stellen weinig voor in vergelijking tot de enorme hoeveelheid "moeilijke" elektronica, inklusief science fiction-achtige kreten als "UART", die zit in de randapparatuur, die via de voornoemde simpele schakelingen op de junior-computer kan worden aangesloten.
Het gaat om een data-zender en een data-ontvanger. Poort PA7 doet dienst als ingang, PBØ als uitgang. De ontvanger is T1 met wat spul eromheen. Het is belangrijk om vast te stellen dat T1 enen op de ingang (bovenkant R5) omtovert in nullen, en omgekeerd. . . . .
Nemen we aan dat ingang 6 van N37 aan +5 V ligt via R8 en vergeten we even wat er via de draadbrug P-Q gebeurt, dan inverteert N37 het ingangsnivo PBØ.
De uitgang van N37 stuurt via R10 de vergelijker IC8. Er wordt vergeleken met de spanning op het knooppunt R11/R12/C12. Is de uitgang van N37 "1", dan is de uitgang van IC8 ca. +12 V; in het geval "0" ca. —12 V. Ook nu stellen we vast dat er tussen



PBØ en de uitgang van IC8 — los van nivo-aanpassing — sprake is van omkering van logische nivo's.
Data-ingang (via R5) en -uitgang (IC8) zijn aangesloten op een genormaliseerde 25-polige D-konnektor, ook wel RS232-konnektor genoemd. Dat "RS232" is niet zo maar een kreet, maar een norm, afspraak voor seriële datakommunikatie: bit voor bit. Elk byte van acht bits wordt voorafgegaan door een startbit, ten teken dat begonnen is met het verzenden van een byte. Bytes zijn gekodeerd volgens de ASCII-kode.
De RS232-norm legt de twee mogelijke logische nivo's vast van een data-kommunikatieleiding, inklusief de bijbehorende spanningsnivo's. Later is daar de pinning van de D-konnektor bijgekomen. Voor een logische nul geldt een lijnnivo tussen +5 V en +15 V (versie RS232C), voor een logische één een lijnnivo tussen —5 V en —15 V. Bij "ons" is het circa +12 V respektievelijk —12 V. U ziet dat bij een "1" een lage spanning hoort en bij een "0" een hoge; dat noemen ze negatieve logika. Door nu zowel na het ontvangen als voor het zenden data (bits) te inverteren hoeft er in de computer niet met "gespiegelde" data te worden gewerkt.
De D-konnektor heeft een aantal pennen die onderling zijn doorverbonden. Andere doorverbindingen zijn mogelijk, afhankelijk van de soort aan te sluiten randapparatuur. Wat is er zo al konkreet aan te sluiten op de junior-computer via de RS232-konnektor? We denken met name aan de Elekterminal, een video-terminal, inklusief ASCII-toetsenbord (met de toetsen van een tiepmachine *plus* een aantal speciale toetsen), die is ontworpen in het kader van de SC/MP-computer, maar eveneens uitermate geschikt is voor de junior-computer.
Het systeemprogramma PRINTER MONITOR is gebaseerd op het ASCII-toetsenbord als invoer (diverse toetskommando's) en de eigenlijke Elekterminal of een geschikte printer (*niet*, althans voorlopig niet, de eerder door Elektuur gepubliceerde metaalfolie-printer) als uitvoer (display). ◀

Tot zover de beschrijving van de hardware. De aangepaste, zwaardere voeding van de junior-computer is niet besproken, evenals een paar wijzigingen van de basiskaart. Voor deze zaken, alsmede voor een korte bouwbeschrijving — mèt de printen — verwijzen we u naar de Elektuur van de volgende maand. In boek 3a zal e.e.a. nog eens dunnetjes (d.w.z. uitgebreid!) worden overgedaan.



JC als voltmeter

G. Sullivan

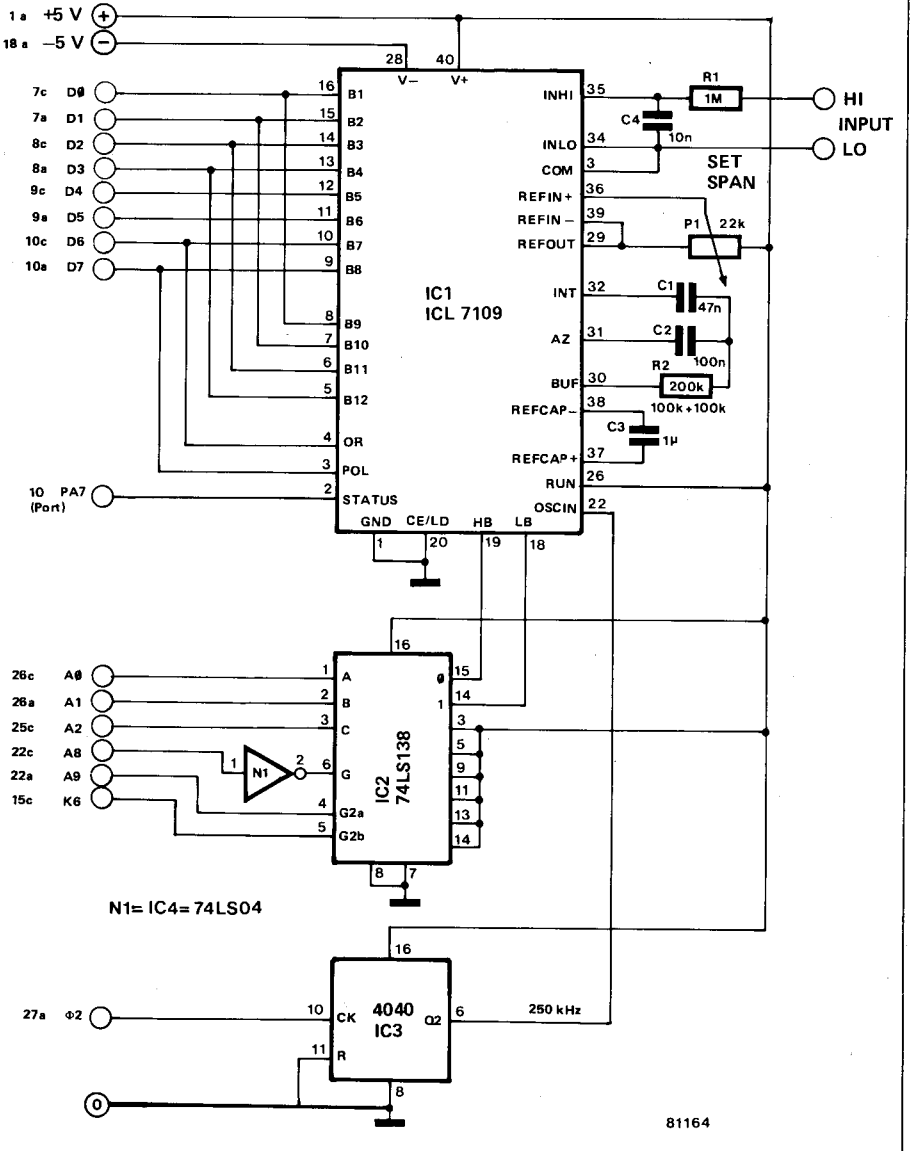
Door het toevoegen van een kleine schakeling en met behulp van het bijbehorende programma is het mogelijk de junior computer te gebruiken als digitale voltmeter. En nog een goede ook! De voltmeter heeft 3¾ digits en een automatische polariteits-aanduiding, terwijl het programma minder dan 180 bytes lang is.

Voor de elektronicus is een voltmeter een apparaat dat hij altijd wel kan gebruiken. Bezitters van een junior computer kunnen met behulp van een kleine schakeling en een klein programma hun junior ook gemakkelijk "omtoveren" tot een digitale voltmeter met uitstekende prestaties.
De schakeling werkt met een A/D-omzetter van Intersil. Dit IC heeft binaire uitgangen zodat de processor het uitgangssignaal eenvoudig kan omzetten in BCD-kode. Doordat het IC tri-state uitgangen bezit is het bijzonder geschikt voor gebruik met microprocessors.

De schakeling

Het Intersil IC ICL 7109 bevat een komplete voltmeterschakeling voor een 3¾ digit uitlezing met automatische polariteitsaanduiding, zodat de opzet van de hele schakeling erg eenvoudig kan blijven.
In figuur 1 is het schema van de komplete schakeling afgebeeld. Het voltmeter-IC bevat een 12 bit A/D-omzetter

1



Figuur 1. Het schema van de meetschakeling waardoor de junior computer in staat is gelijkspanningen te meten. IC1 verzorgt de hele omzetting van analoog ingangssignaal naar een 12 bits-kode voor de computer.

met tri-state-uitgangen zodat een rechtstreekse aansluiting op een microprocessor-systeem mogelijk is.
Het uitgangssignaal aan de uitgangen B1 . . . B8 wordt uitgelezen in twee bytes. Welk byte wordt uitgelezen wordt bepaald door de adresdekoder IC2. Het minst signifikante byte bevat de acht laagste bits en het meest signifikante byte de vier hoogste bits plus een over-range bit en een polariteitsbit.
Met de gegeven dimensionering is de ingangsspanning voor volle schaal-uitslag 4,096 V. De omzettingssnelheid is ongeveer 30 omzettingen per sekonde. Het ingangsspanningsbereik kan men aanpassen door het veranderen van R2 en de referentiespanning die met P1 wordt ingesteld.

Hiervoor geldt: $R2 = \frac{U_{volle\ schaal}}{20\ \mu A}$ en

$U_{volle\ schaal} = 2 \cdot U_{ref}$.
De waarde van C1 en C2 wordt bepaald door de gebruikte oscillator-frekwentie, waarbij $C1 = \frac{2048 \cdot periodetijd \cdot 20\ \mu A}{3{,}5\ V}$

en $C2 = 2 \cdot C1$.
De gebruikte frekwentie van 250 kHz voor de konverter wordt afgeleid van de clock van de microprocessor. Daartoe is IC3 als vierdeler geschakeld. Indien gewenst kan men, door een andere uitgang van IC3 te nemen, de konverter op een andere frekwentie laten lopen. Daarmee verandert dan ook het aantal omzettingen per sekonde. Door de hoge ingangsimpedantie van de omzetter is het vrij eenvoudig ingangsverzwakkers toe te passen om zo meerdere spanningsbereiken te verkrijgen.

Het programma

Het programma voor het gebruik van de junior computer als digitale voltmeter is afgedrukt in tabel 1. Dit zorgt er voor dat de twee door de konverter afgegeven bytes worden ingelezen, waarna een binair-naar-decimaal-omzetting volgt en dit resultaat op de displays zichtbaar wordt gemaakt. In het geval van een negatief ingangssignaal wordt de stand van de polariteitsvlag gedetekteerd en een min-teken uitgelezen op het display. Als de maximale ingangsspanning van de konverter wordt overschreden geeft het display het teken OL (overload), samen met de polariteit.
In het schema van figuur 1 is de omzetter geschakeld in de zogenaamde "free run mode", wat inhoudt dat de konverter na het beëindigen van een analoog-naar-digitaal-omzetting meteen weer begint met de volgende omzetting. Voor normaal gebruik werkt dit prima, maar in sommige gevallen kan het wenselijk zijn om te detekteren wanneer een omzetting beëindigd is en dan de data in te lezen om te vermijden dat er wordt ingelezen terwijl de data verandert.
Dit kan men doen door de statusuitgang van IC1 te verbinden met PA7 van de poortkonnektor en de neergaande flank van deze uitgang te gebrui-

Tabel 1.

```
LINE  LOC   OBJECT      SOURCE

0001  0000               ;DIGITAL VOLTMETER PROGRAM
0002  0000               ;FOR INTERSIL ICL 7109.
0003  0000               ;
0004  0000               ;AUTHOR G.SULLIVAN
0005  0000               ;
0006  0000                       * = $00D0
0007  00D0               ACUM    *=*+2
0008  00D2               SUM     *=*+1
0009  00D3               DELAY   *=*+1
0010  00D4               TMPX    *=*+1
0011  00D5               ;
0012  00D5               ;DEFINE A/D CONVERTER.
0013  00D5                       * = $1800
0014  1800               HB      *=*+1           ;HIGH NIBLE + FLAGS
0015  1801               LB      *=*+1           ;LOW BYTE
0016  1802               ;
0017  1802               ;DEFINE PIA
0018  1802                       * = $1A80
0019  1A80               PRA     *=*+1           ;DATA A REG.
0020  1A81               DDRA    *=*+1           ;A DIRECTION REG.
0021  1A82               PRB     *=*+1           ;DATA B REG.
0022  1A83               DDRB    *=*+1           ;B DIRECTION REG.
0023  1A84               ;
0024  1A84               SCAND1 =$1DCC           ;DISPLAY 1 BYTE
0025  1A84               ;
0026  1A84                       * = $0200
0027  0200               ;
0028  0200               ;MAIN DISPLAY ROUTINE
0029  0200  2C 00 18     MAIN    BIT HB          ;TEST OVERANGE BIT
0030  0203  70 06                BVS OL
0031  0205  20 24 02             JSR DISVLT      ;SHOW VOLTS
0032  0208  4C 12 02             JMP NOL
0033  020B  A2 0C        OL      LDX ##0C        ;DISPLAY OL MESSAGE
0034  020D  A0 03                LDY ##03
0035  020F  20 49 02             JSR DISTAT
0036  0212  AD 00 18     NOL     LDA HB          ;TEST POLARITY BIT
0037  0215  30 07                BMI NOT
0038  0217  A2 08                LDX ##08        ;DISPLAY MINUS
0039  0219  A0 00                LDY ##00
0040  021B  20 49 02             JSR DISTAT
0041  021E  20 73 02     NOT     JSR HEXBCD      ;CONV. BINARY TO BCD
0042  0221  4C 00 02             JMP MAIN
0043  0224               ;
0044  0224               ;VOLT DISPLAY SUBROUTINE
0045  0224  A9 7F        DISVLT LDA ##7F         ;SET PIA TO OUTPUT
0046  0226  8D 81 1A             STA DDRA
0047  0229  A2 0C                LDX ##0C        ;ADDRESS OF FIRST BYTE
0048  022B  A0 FF                LDY ##FF
0049  022D  C8          LOOP    INY
0050  022E  B9 D0 00             LDA ACUM,Y      ;GET BYTE
0051  0231  20 CC 1D             JSR SCAND1      ;LIGHT DISPLAY
0052  0234  E0 14                CPX ##14        ;TEST IF TWO BYTES YET
0053  0236  D0 F5                BNE LOOP
0054  0238  AD 00 18             LDA HB          ;TEST POLARITY
0055  023B  30 07                BMI NXT         ;SHOW - IF NEG.
0056  023D  A2 08                LDX ##08
0057  023F  A0 00                LDY ##00
0058  0241  20 49 02             JSR DISTAT
0059  0244  E6 D3        NXT     INC DELAY       ;DELAY ON DISPLAY
0060  0246  D0 DC                BNE DISVLT
0061  0248  60                  RTS
0062  0249               ;
0063  0249               ;DISPLAY '-' ,'OL'
0064  0249  A9 7F        DISTAT LDA ##7F         ;SET PIA TO OUTPUT
0065  024B  8D 81 1A             STA DDRA
0066  024E  B9 6D 02             LDA CHART,Y     ;GET MESSAGE BYTE
0067  0251  30 14                BMI ENDD
0068  0253  8D 80 1A             STA PRA         ;LIGHT SEGMENTS
0069  0256  8E 82 1A             STX PRB         ;SELECT DIGIT
0070  0259  86 D4                STX TMPX
0071  025B  A2 FF                LDX ##FF
0072  025D  CA          DLY2    DEX             ;DELAY
0073  025E  D0 FD                BNE DLY2
0074  0260  A6 D4                LDX TMPX
0075  0262  E8                  INX
0076  0263  E8                  INX             ;NEXT DIGIT
0077  0264  C8                  INY
0078  0265  D0 E2                BNE DISTAT
0079  0267  A9 00        ENDD    LDA ##00        ;DISPLAY OFF
0080  0269  8D 82 1A             STA PRB
0081  026C  60                  RTS
0082  026D  7F          CHART   .BYTE $7F,$3F,$80
0082  026E  3F
0082  026F  80
0083  0270  40                  .BYTE $40,$47,$80
0083  0271  47
```