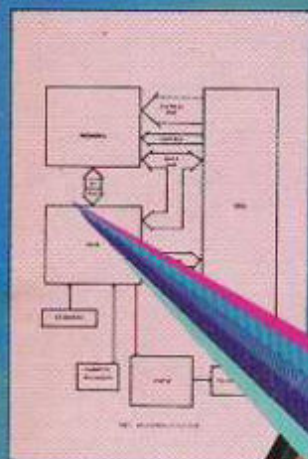


ZX Spectrum hardware-boek



SPECTRUM *Hardware* MANUAL



Adrian Dickens

ZX Spectrum HARDWARE boek



Kluwer technische boeken

Omslag: W. Niessink
Vertaling: E.R. Visser

ISBN 90 201 1762 9
D/1985/0108/167

Oorspronkelijke titel: Spectrum Hardware Manual
Uitgegeven bij: Melbourne House Publishers

©1983 A.C. Dickens

This edition is published by arrangement with Melbourne House (Publishers) Ltd.,
London

©1985 van de Nederlandse vertaling bij Kluwer Technische Boeken B.V. - Deventer

1e druk 1985

Niets uit deze uitgave mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm of op welke andere wijze ook, zonder voorafgaande schriftelijke toestemming van de uitgever.

No part of this book may be reproduced in any form, by print, photoprint, microfilm or any other means without written permission from the publisher.

Ondanks alle aan de samenstelling van de tekst bestede zorg, kan noch de redactie noch de uitgever aansprakelijkheid aanvaarden voor eventuele schade, die zou kunnen voortvloeien uit enige fout, die in deze uitgave zou kunnen voorkomen.

Inhoud

1	Inleiding.	9
2	De Spectrum in het algemeen.	11
2.1	Deel A.	11
2.2	Deel B.	11
3	De voeding	16
3.1	De +5 volt voeding	16
3.2	De +12 volt voeding	18
3.3	De -5 volt voeding	19
3.4	Stroombegrenzings	19
3.5	De -12 volt voeding	19
3.6	Extra +5 volt voeding	21
4	De Z80A Centrale Verwerkingseenheid (CPU)	24
4.1	Z80A CPU-penbeschrijvingen	26
5	Video- en programmeergeheugen.	29
6	Het BASIC 'alleen te lezen' geheugen (ROM)	32
7	Het toetsenbord.	35
8	Het niet-gebonden logische netwerk (ULA)	38
8.1	ULA-penbeschrijvingen	38
8.2	De versteltransistor in model 2	42
8.3	Video-uitvoer	42
8.4	Iets over onderbrekingen	43
8.5	Klokken	44
8.6	Toetsenbord en cassette-invoer	45
8.7	Zoemer-, cassette- en randkleuruitvoer	45
9	De geheugenuitbreiding.	46
10	Het videocircuit	50

11	Regel het videocircuit af voor een betere beeldkwaliteit.	52
11.1	Verwijdering van de bewegende streepjes naast de letters	52
11.2	Het veranderen van de kleur of de grijsschaal	52
12	De randconnector	56
12.1	De aansluitingen aan kant A.	56
12.2	De aansluitingen aan kant B.	58
13	Het opsporen van fouten	61
14	Experimenteren met de randconnector	63
14.1	Wat heeft u nodig	63
14.2	Herstelschakelaar	65
14.3	HALT-experiment	66
14.4	NMI-test.	66
14.5	INT- en IORQGE-proeven	66
14.6	Het toevoegen van 128 I/U-poorten	68
15	Het toevoegen van een Z80A-PIO-chip	71
15.1	Proeven met LED's en schakelaars	75
16	Het toevoegen van uw eigen toetsenbord.	76
16.1	Inleiding.	76
16.2	Beschrijving van de schakeling	76
16.3	Bouwaanwijzingen	80
17	Stuurknuppels aangepast aan de Sinclair	81
17.1	Beschrijving 'Apollo'-programma	84
17.2	Programma Apollo	85
18	Acht-kanaals analoog-naar-digitaal omzetter	87
18.1	Bouwbeschrijving	87
18.2	Hoe werkt het	89
18.3	Tekenprogramma	92

Appendices.	94
A Verklarende woordenlijst	94
B Referenties	97
C Onderdelenlijst van de Spectrum	98
D Onderdelenopstelling	100
E Volledig schema.	102
Index.	103

1. Inleiding

De ZX Spectrum wordt geleverd met een introductieboekje en een BASIC-programmeerboek. Deze twee boeken bevatten geen informatie over de hardware van de Spectrum. Dit boek wil deze lacune opvullen door dieper op de werking van de Spectrum in te gaan. Voor de echte beginner die niets van computerhardware begrijpt, vormt dit boek een goede inleiding in de werking van de computer. Voor de gevorderde elektronicus zal dit boek van veel waarde zijn door de vele schema's en werkbeschrijvingen.

Het boek begint met een verhandeling over de basisprincipes van de Spectrum, daarna wordt verder gegaan met een gedetailleerde beschrijving van elk afzonderlijk deel van de computer en hoe die delen samenwerken. Na een volledige verklaring van alle signalen die op de randconnector aanwezig zijn, worden er eenvoudige tot de verbeelding sprekende toepassingen gegeven die gebruik maken van deze signalen. De rest van het boek bevat praktische schakelingen die u kunt nabouwen. De beginner zonder ervaring wordt geholpen met bouw-aanwijzingen, tekeningen en foto's.

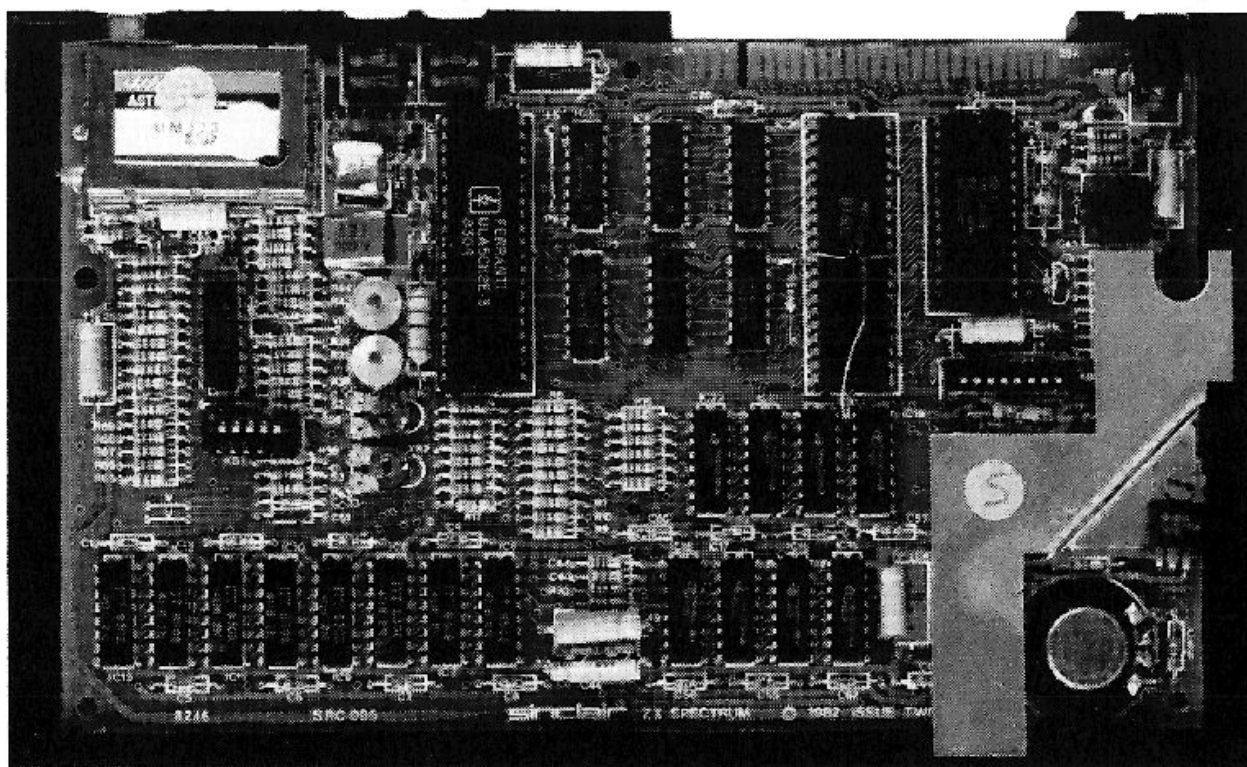
Alle Spectrum-schema's bevatten onderdelennummers zoals TR4. Dit zijn de codes zoals ze op de printplaat van de Spectrum gedrukt staan. In appendix D staan de onderdelenopstellingen van de Spectrum model 2 en model 3 en in appendix E staat het volledige schema.

Wanneer u de plaats van een onderdeel zoekt kunt u van deze appendices gebruik maken. De onderdelencode welke bij de hardware-projecten wordt gegeven, verwijst naar de onderdelenlijst van dat project en niet naar de Spectrum-onderdelenlijst.

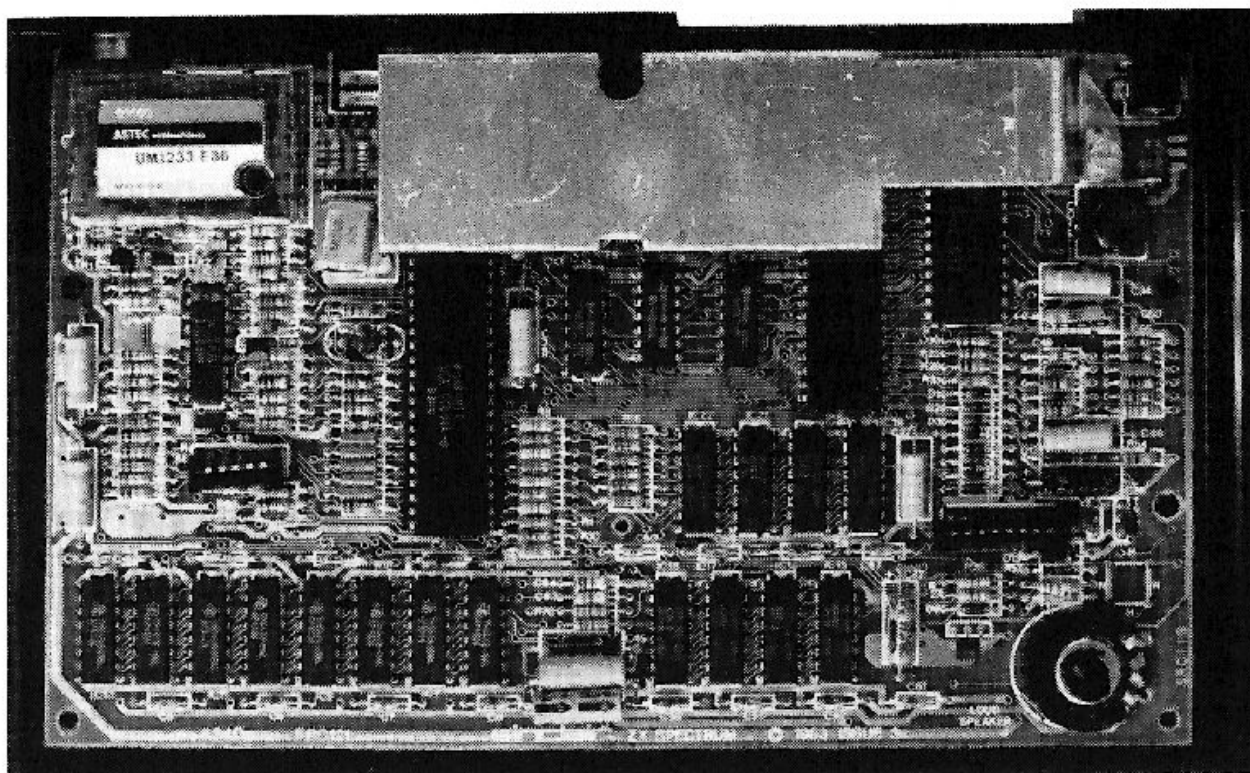
Voor we dieper in de ingewikkelde hardware duiken eerst een waarschuwend woord. De Spectrum bevat veel gevoelige en dure onderdelen. Deze zullen niet beschadigd worden mits u voldoende voorzorgen treft. Wanneer u een interface op de Spectrum wilt aansluiten moet u eerst de voedingsspanning uitschakelen. Schakel de voedingsspanning van de Spectrum voor of tegelijkertijd in met die van het interface. Als u verschillende spanningen gebruikt binnenin de Spectrum of op de randconnector, zorg er dan voor geen kortsluiting te veroorzaken. Controleer en controleer nog eens alles wat u doet!

DENK ERAAN – hoewel er alles aan gedaan is om ervoor te zorgen dat de gegeven informatie in dit boek juist is, kan er geen verantwoordelijkheid worden genomen voor eventuele fouten. Alle Spectrumschema's vallen onder het auteursrecht van Sinclair Research Limited. Andere ontwerpen en programma's vallen onder het auteursrecht van de schrijver.

Model 2



Model 3



Afb. 1.1 Foto van de Spectrum als de kast geopend is

2. De Spectrum in het algemeen

Dit hoofdstuk bestaat uit twee delen. Het eerste deel legt de beginselen van binaire (tweetallige) getallen uit. Het tweede deel vertelt wat over elk hoofddeel van de Spectrum.

2.1 Deel A

Computers kennen maar twee toestanden. Voor het gemak worden deze twee toestanden weergegeven door een 0 en een 1. Op het eenvoudigste niveau werkt de computer met enen en nullen om de antwoorden te maken. Laten we als voorbeeld eens een optelling bekijken. Deze kan worden voorgesteld als een zwarte doos met twee ingangen, A en B, en twee uitgangen C en D. A en B kunnen bij elkaar worden opgeteld en de som wordt voorgesteld door C en D. De optelling wordt uitgevoerd door enkele transistors in de zwarte doos. De optelfunctie wordt binair als volgt gedefinieerd:

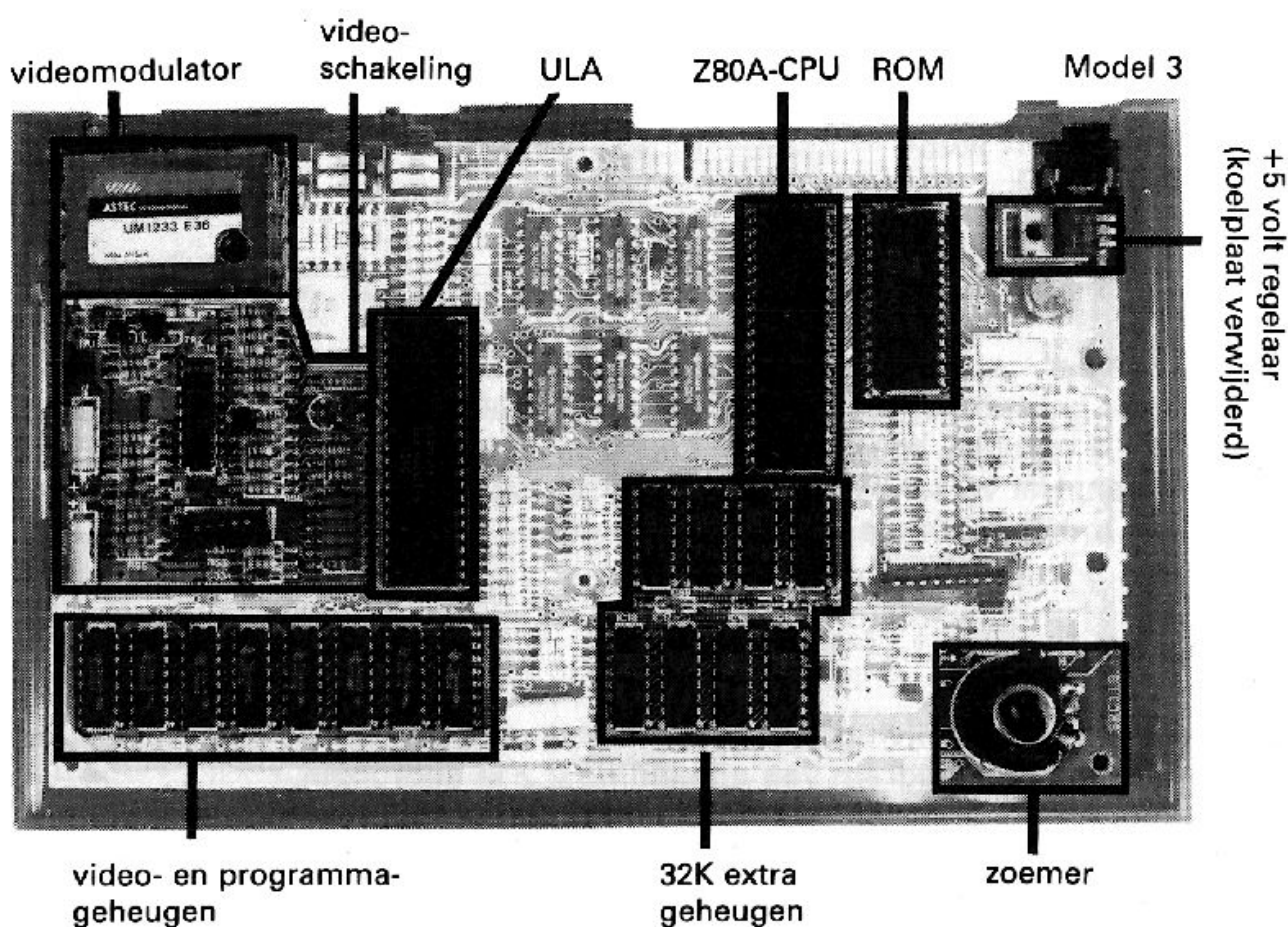
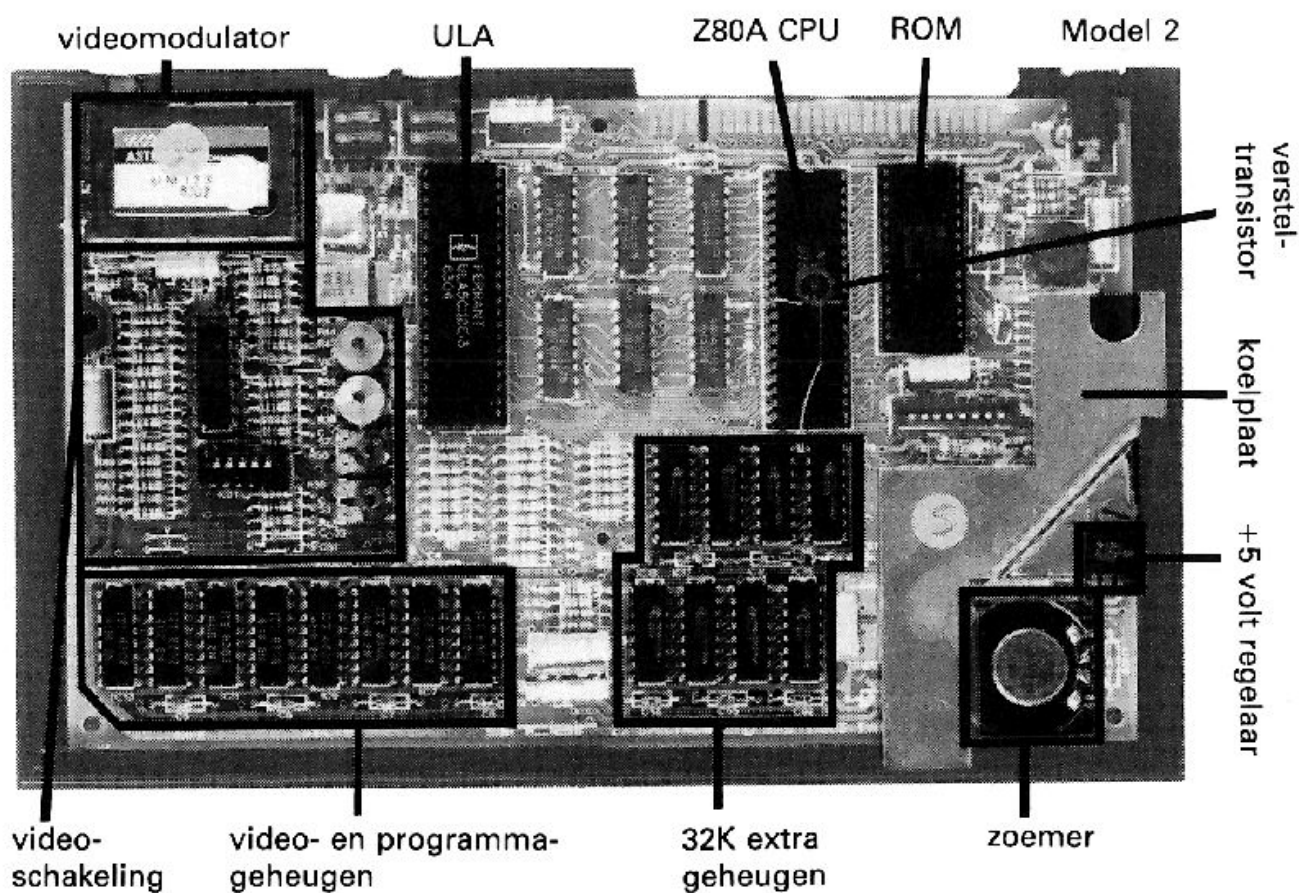
A + B = D (som)	&C (carry)
0 + 0 = 0	0
0 + 1 = 1	0
1 + 0 = 1	0
1 + 1 = 0	1

Denk eraan dat $1 + 1$ niet gelijk aan 2 kan zijn in het binaire talstelsel, alleen de 0 en 1 kunnen gebruikt worden. Het carry-bit is hetzelfde als in het decimale stelsel. Als we $8 + 9$ optellen, zou dat in het decimale stelsel 7 en 1 onthouden (Engels: carry) geven. Het verschil is dat een 'carry' in het binaire stelsel twee voorstelt terwijl het in het decimale stelsel tien voorstelt.

2.2 Deel B

Bij de volgende beschrijvingen van de verschillende delen in de Spectrum, is het handig om het blokdiagram van afbeelding 2.2 te gebruiken.

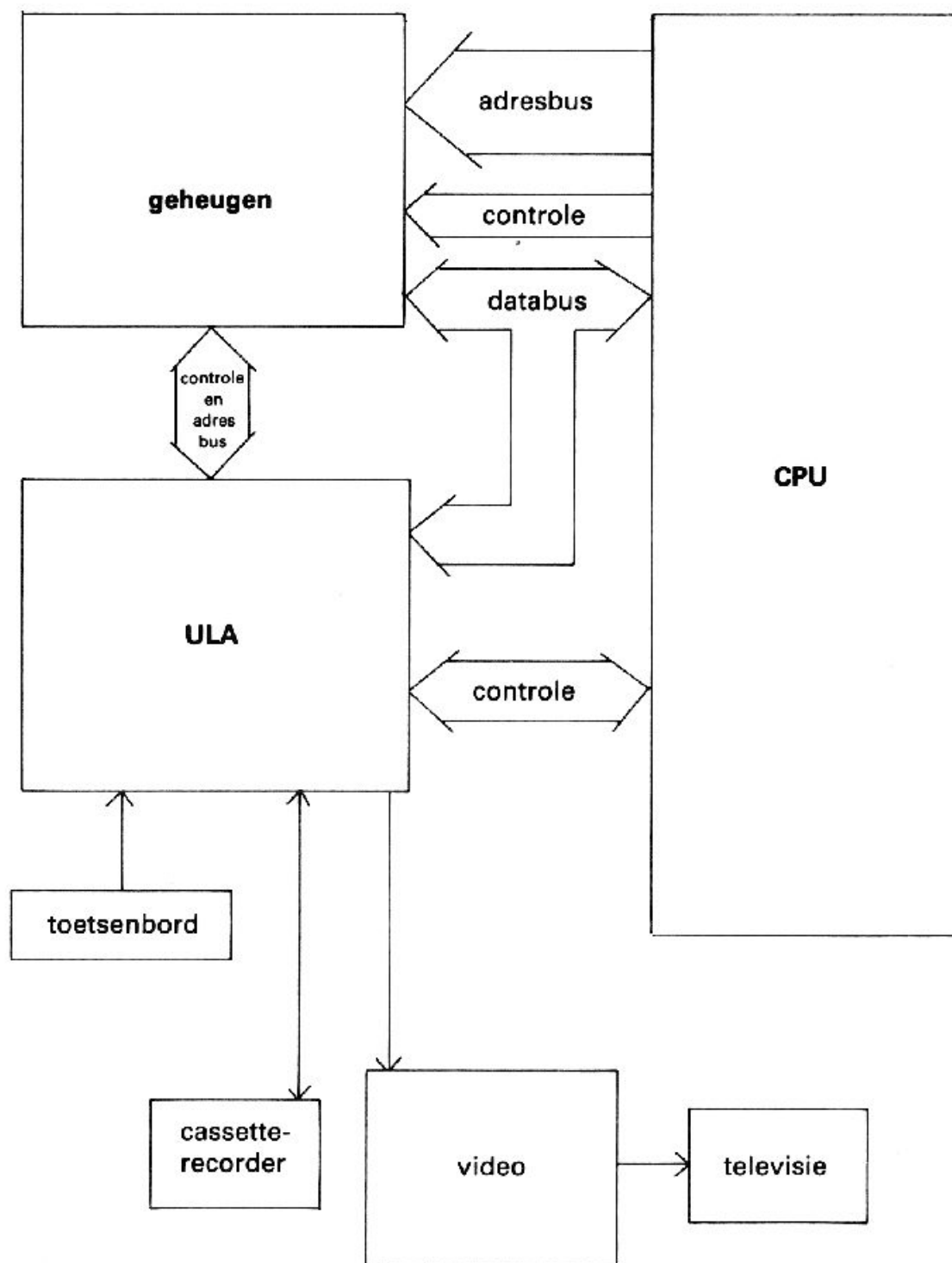
De centrale verwerkingseenheid (CPU) is het hart van de Spectrum. Hij is verbonden met de andere delen van de computer door middel van data-, controle- en adresbussen (hierover later meer). De CPU in de Spectrum is de Z80A en bevindt zich in de grote chip IC2 in het midden van uw Spectrum-print. Deze processor is een 8-bit instrument wat betekent dat er 8 verbindingen op de databus zijn. De



Afb. 2.1 Foto van het Spectrum-bord met de onderdelengroepen

CPU kan informatie naar andere onderdelen in de Spectrum sturen over de databus en die onderdelen kunnen via dezelfde bus gegevens terugsturen. Omdat er acht lijnen in de databus zitten die elk 1 of 0 kunnen zijn, kan elk getal tussen 0 (alles 0) en 255 (alles 1) over de databus verzonden worden ($255 = 2^8 - 1$).

U kunt zich afvragen hoe de CPU grote decimale getallen of tekst kan begrijpen die ingetikt worden als u BASIC gebruikt. Als u HALLO intikt op het toetsenbord van de Spectrum is het moeilijk te begrijpen hoe dit door een getal tussen 0 en



Afb. 2.2 Blokschema van de Spectrum

255 wordt voorgesteld. In feite is het antwoord niet moeilijk. De CPU heeft steeds maar met een heel klein deel van de totale opdracht te maken. Om HALLO te begrijpen, houdt de CPU zich eerst bezig met de H (de CPU ziet dat als 72 decimaal), daarna houdt hij zich met de A bezig (gezien als 65 decimaal) enzovoorts. In appendix A van het BASIC-handboek voor de Spectrum vindt u een volledige lijst van de letters en tekens en hun decimale waarde. Grote decimale getallen worden op een overeenkomstige manier behandeld. Elk decimaal getal wordt in het Spectrum-geheugen opgeslagen in 5 bytes. Hoofdstuk 24 van het BASIC-handboek vertelt hoe deze bytes gebruikt worden.

Voordat de CPU echt iets kan doen, moet hem verteld worden wát hij moet doen. De BASIC-opdrachten staan in het geheugen. Het BASIC-besturingssysteem (operating system) bevat alle informatie die de CPU nodig heeft om BASIC te begrijpen. Dit besturingssysteem is geschreven in machinetaal en begint te lopen zodra de Spectrum wordt ingeschakeld. Het besturingssysteem is opgeslagen in het 'alleen te lezen'-geheugen (Read Only Memory = ROM) IC 5. Het ROM-geheugen kan niet veranderd worden door de CPU en blijft, ook als de spanning uitgeschakeld wordt, in de geheugenchip aanwezig. Het BASIC-programma dat u intikt gaat naar het 'willekeurig toegankelijk'-geheugen (Random Access Memory = RAM). Anders dan bij de ROM kan dit geheugen veranderd worden door de CPU. Als de voedingsspanning uitgeschakeld wordt vergeet de RAM alles dat erin was opgeslagen. Daarom moet u uw programma's op cassettes schrijven voordat u de computer uitschakelt. Als u dit niet doet zal het programma verloren gaan.

Nadat de CPU in de ROM ontdekt heeft wat hij moet doen, krijgt hij invoer via het toetsenbord of via de cassette en geeft hij uitvoer naar het TV-scherm of de cassette. Het 'niet gebonden logische netwerk' (Uncommitted Logic Array = ULA) helpt de CPU in de verbinding met de buitenwereld.

De ULA krijgt de informatie rechtstreeks van het toetsenbord of van de cassette. Deze informatie wordt dan naar de CPU gezonden. Als de CPU een programma op de cassette wil schrijven of de zoemer wil laten zoemen, dan geeft hij de ULA opdracht dit te doen. Uitvoer naar het TV-scherm is echter ingewikkelder. De ULA kopieert de schermuitvoer 50 maal per seconde vanuit het video-geheugen. Dit geeft de indruk van een continue afbeelding. Alles wat de CPU moet doen wanneer hij uitvoer naar het TV-scherm wenst is de informatie in het video-geheugen schrijven. De ULA doet dan de rest.

Tot nu toe is de overdracht van informatie tussen de CPU, het geheugen, de ULA, het toetsenbord en dergelijke als vanzelfsprekend aangenomen. Hoe gebeurt het werkelijk?

Alle gegevens worden overgebracht via de databus. Het soort overdracht wordt aangegeven met controlesignalen op de controlebus. Bijvoorbeeld, de CPU geeft een leessignaal af als hij gegevens uit de ULA of uit het geheugen wenst te lezen. Hierdoor weet de ULA of het geheugen dat er gegevens naar de CPU moeten. Als de CPU iets in het geheugen op wil slaan geeft het een schrijfsignaal af, daardoor weet het geheugen dat het de gegevens van de databus op moet slaan. Om het geheugen te laten weten waar het de gegevens moet opbergen zet de CPU tevens een adres op de 16-bit adresbus. De adresbus stelt de CPU in staat

data naar $2 \uparrow 16 = 65536$ verschillende geheugenplaatsen te verzenden of daaruit te lezen.

De bussen kennen slechts de twee logische toestanden 0 en 1 voor elk van hun lijnen. In de praktijk worden deze logische toestanden weergegeven met spanningen. De afspraak is (hierdoor kunnen de moderne computer-IC's met elkaar worden uitgewisseld) dat een logische 0 weergegeven wordt met een spanning tussen 0 en 0,8 volt en de logische 1 wordt weergegeven met een spanning tussen +2 en +5 volt (de maximum spanning voor logische chips). Als de spanning tussen 0,8 en +2 volt ligt, gaat het signaal van 0 naar 1 of van 1 naar 0. Alle overdracht van logische data gebeurt tussen deze spanningsgrenzen. De chips zijn zo ontworpen dat ze geen gegevens lezen als deze kunnen veranderen.

U kent nu enigszins de blokken waaruit uw Spectrum is opgebouwd. De volgende hoofdstukken zullen deze blokken meer in detail behandelen. Elk blok wordt op zich bekeken, maar u moet in gedachten houden hoe het met de rest van het systeem verbonden is. Het volledige schema van de logische onderdelen die uw Spectrum vormen vindt u in appendix E. U kunt dit schema altijd raadplegen als u niet meer weet hoe de verschillende circuits met elkaar verbonden zijn.

3. De voeding

De voeding van elke computer is waarschijnlijk het onderdeel waaraan de minste aandacht wordt besteed door de gebruiker. De computer is verbonden met het lichtnet en hij werkt. Het wordt gewoon gevonden dat de juiste spanningen en stroomsterktes worden gevormd. Wanneer u uw eigen hardware gaat maken is het nodig dat u de stroomvoorziening van de computer kent. Daarom bekijken we in dit hoofdstuk dit vergeten onderdeel eens nauwkeuriger.

Het laat zien hoe we zo goed mogelijk gebruik kunnen maken van de Spectrumvoeding en geeft ook schema's voor extra voedingsspanningen die we kunnen gebruiken bij de externe schakelingen.

In de Spectrum komt de voeding binnen als +9 volt met een stroomsterkte tot 1,2 ampère. Deze energie komt uit een ZX-energievoorziening (power supply). Helaas gebruikt geen van de chips in de Spectrum een voedingsspanning van 9 volt. De meeste chips, inclusief de CPU werken op +5 volt. De ULA vraagt daarnaast ook +12 volt. De videoegeuegenschap is de onhandigste van alle en vraagt zowel +12, +5 als -5 volt.

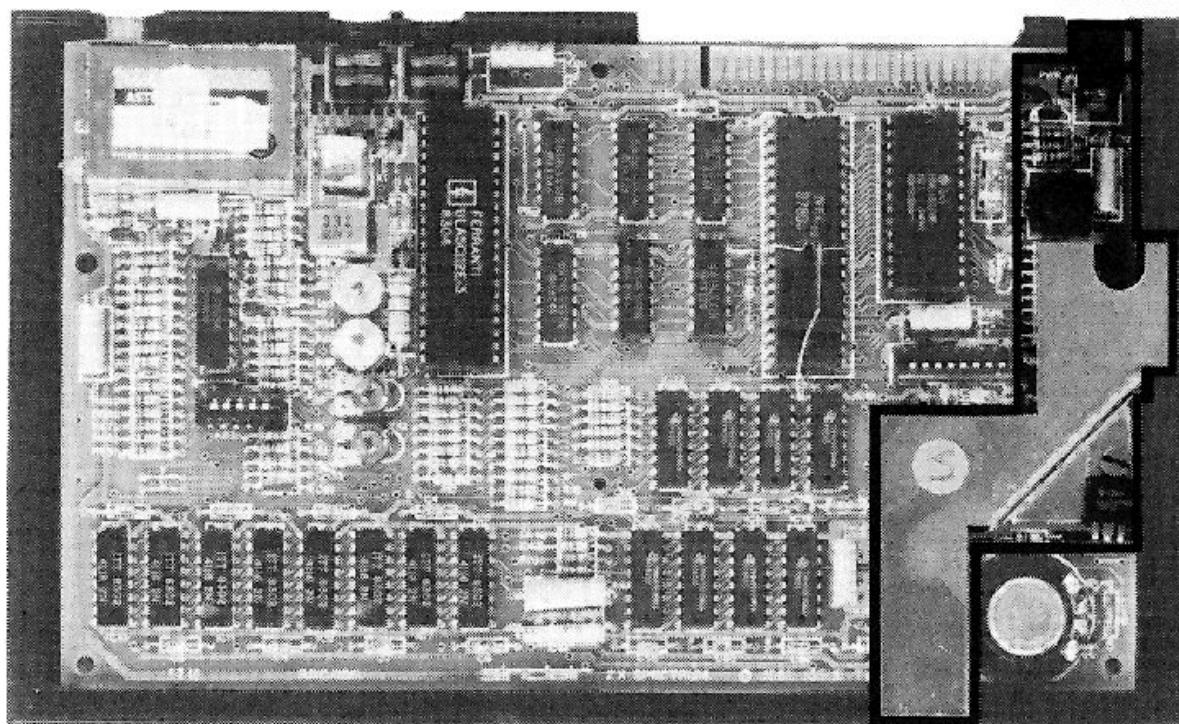
Het probleem is niet om een spanning te maken die constant van ongeveer de juiste waarde is. De +5 volt voeding moet binnen 5% van +5 volt zijn en de +12 en -5 volt voedingen moeten binnen 10% nauwkeurig zijn in de totale schakeling en te allen tijde. Zelfs een spanningsval gedurende één microseconde kan een ramp betekenen. Hoe worden deze stabiele spanningen nu gemaakt?

3.1 De +5 volt voeding

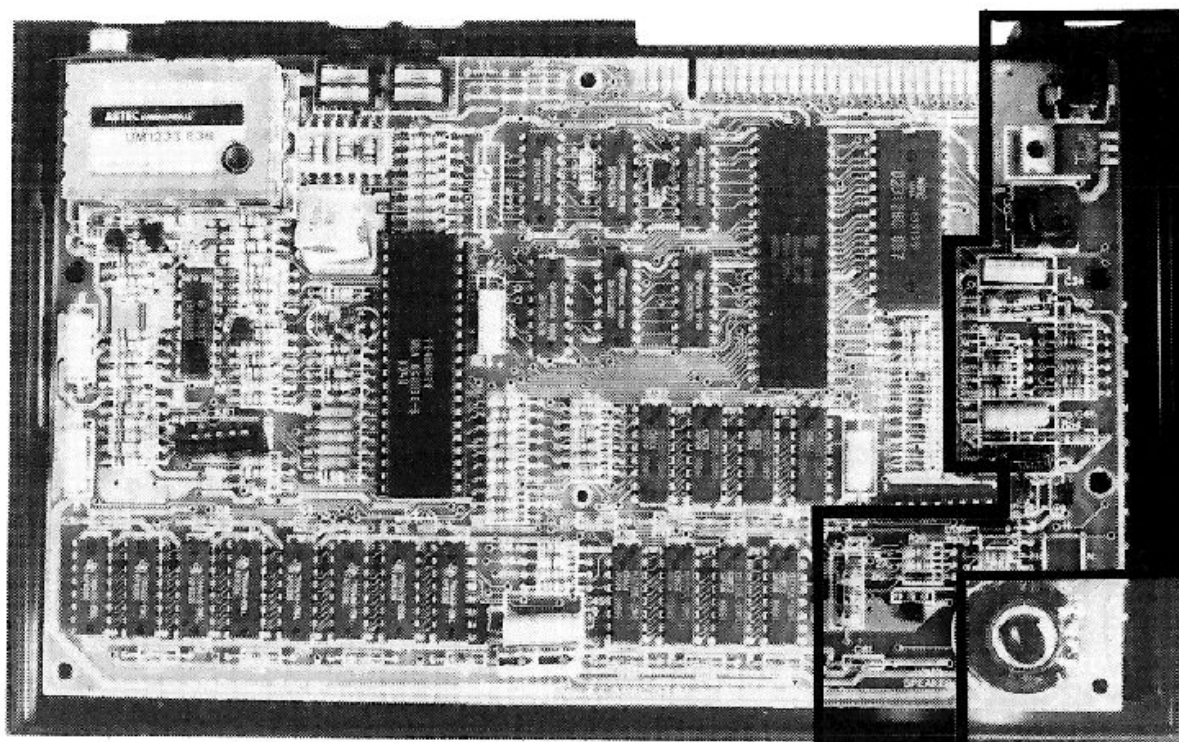
Dit is de belangrijkste spanning in de Spectrum. In de 48K uitvoering werkt hij op de toppen van zijn kunnen om de volle 1 ampère stroom te leveren. Bekijken we de spanningsregelaar dan zien we dat hij op een groot stuk aluminium geschoefd zit. De kleine +5 volt regelaar met slechts drie aansluitingen naar de rest van de schakeling bevat een ingewikkelde regelschakeling.

In afbeelding 3.2 zien we dat de 7805-regelaar +9 volt op de IN-pen krijgt toegevoerd. Het interne regelcircuit reduceert dit tot +5 volt op de OUT-pen. Het feit dat de ingang +9 volt is doet niet terzake (met uitzondering voor de hoeveelheid uitgestraald vermogen). Het mag elke spanning tussen de +7 en +25 volt zijn, de uitgang zal stabiel op +5 volt blijven. U kunt zich afvragen waar die overige 4 volt gebleven is? Die wordt als warmte uitgestraald door de aluminium koelplaat. Als de regelaar 1 ampère levert dan wordt er 4 watt door de koelplaat uitgestraald

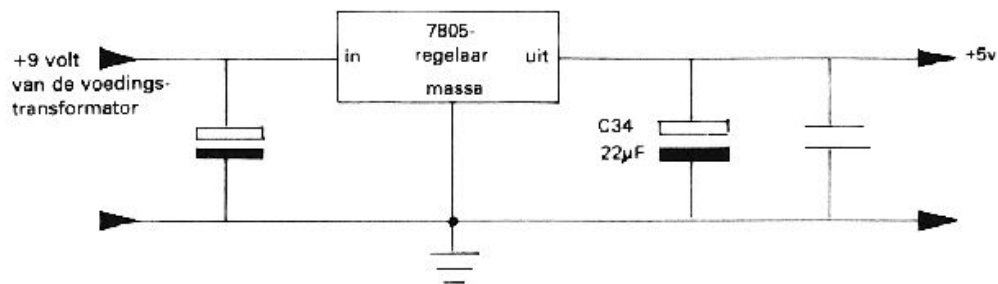
Model 2



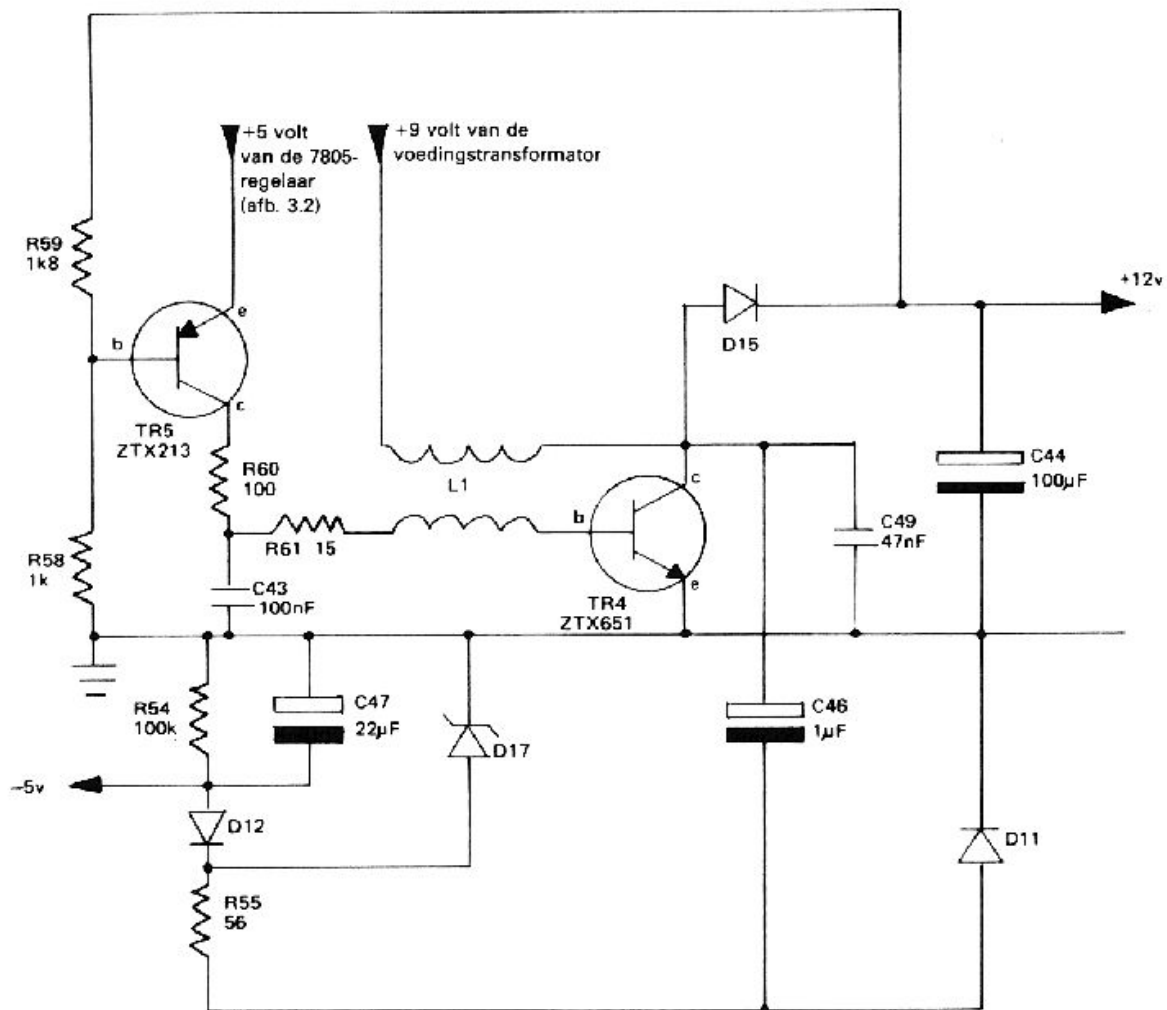
Model 3



Afb. 3.1 Foto van het Spectrum-bord met de voeding aangegeven



Afb. 3.2 Schema van de +5 volt voeding



Afb. 3.3 Schema van de +12 volt en de -5 volt voeding

(ongeveer een kwart van een kleine soldeerbout). Daarom wordt de Spectrum na het aanzetten nogal warm.

3.2 De +12 volt voeding

De +5 volt voeding was betrekkelijk eenvoudig te verzorgen. De spanningsregelaar werkt 4 volt weg. Om +12 volt te maken met slechts +9 volt aanwezig is ingewikkelder.

Bekijken we afbeelding 3.3, dan zien we dat de TR5 en TR4 met de bijbehorende onderdelen de 12 volt voeding verzorgen. TR5 vormt een stroomterugkoppeling voor de oscillator die door C43, R61, C1 en TR4 (de spanning-stuurtransistor) gevormd wordt. De werking van de schakeling berust op de geïnduceerde omgekeerde spanning over L1 die optreedt bij elke trillingsperiode. Deze omgekeerde spanning brengt de collector van TR4 boven de 9 volt tot maximaal 13 volt. Op dit niveau komt D15 in geleiding en wordt de +12 volt voedingscondensator C44 opgeladen.

De ontlading van C44 levert een constante +12 volt spanning voor de ULA, de geheugens en het videocircuit als D15 niet geleidt. Als de +12 volt te laag wordt dan gaat TR5 meer geleiden. De oscillatiefrequentie neemt toe, waardoor het voltage op zijn oude niveau komt.

3.3 De -5 volt voeding

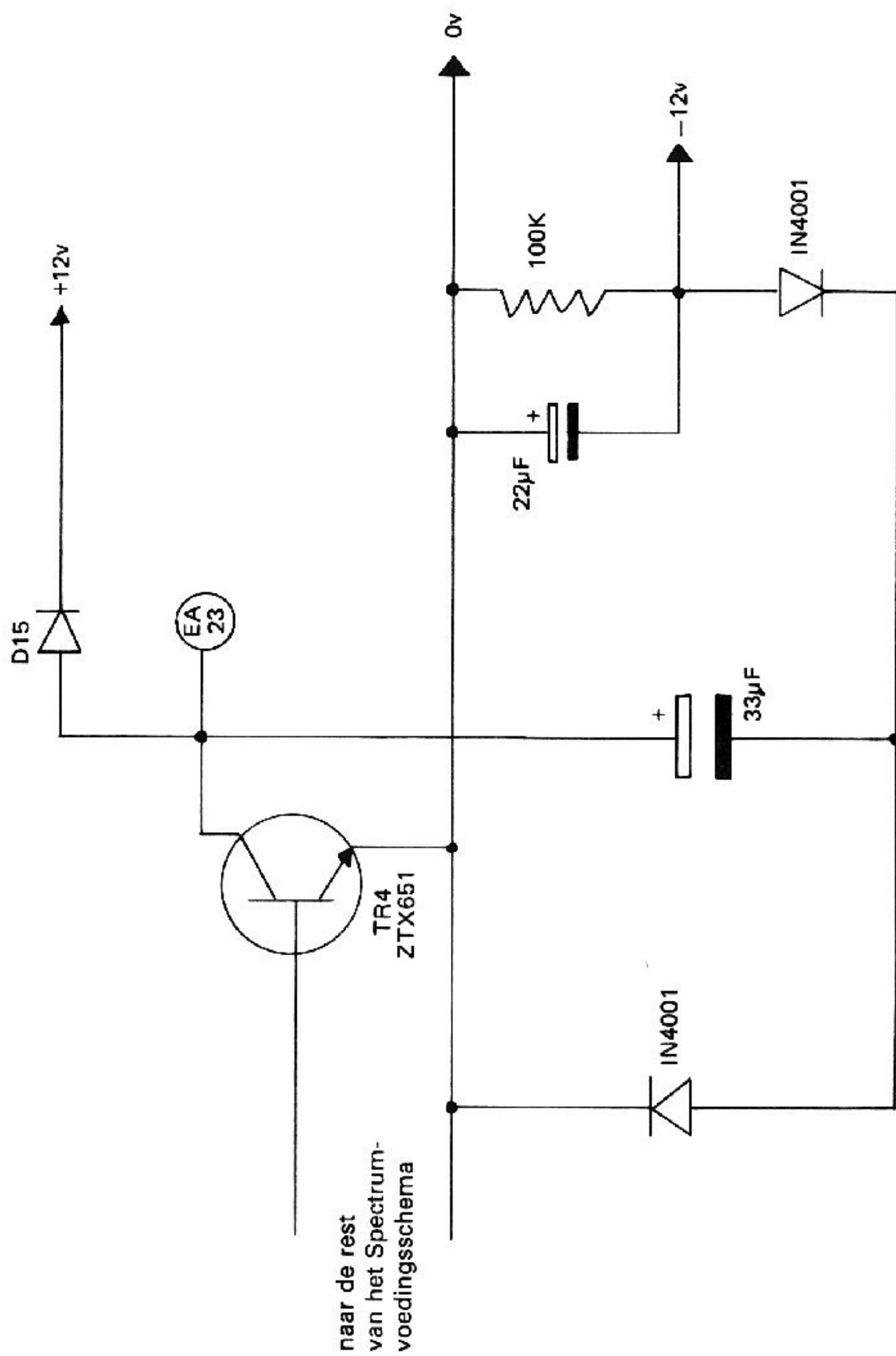
Bekijk hiervoor ook afbeelding 3.3. Het -5 volt deel bestaat uit C46, D11, R55, D12, D17, R54 en C47. Deze schakeling werkt volgens het 'ladingspomp'-principe. U herinnert zich dat de collector van TR4 in het 12 volt circuit snel wisselt tussen de +13 volt en de 0 volt. Als hij stijgt naar 13 volt wordt C46 via D11 tot maximaal 12 volt opgeladen (0,7 volt gaat verloren over de siliciumdiode bij geleiding). Als de collector van TR4 naar 0 volt daalt, wordt de negatieve pool van C46 -12 volt. C47 wordt dan via D12 en R55 uit C46 opgeladen. De spanning over C47 wordt door de zenerdiode D17 constant gehouden op -5 volt. C46 wordt dan weer opgeladen in de volgende oscillatorcyclus enzovoorts.

3.4 Stroombegrenzings

De stromen die uit de interne voeding van de Spectrum gehaald kunnen worden voor gebruik in externe schakelingen zijn nogal beperkt. Bij de 16K Spectrum kan ongeveer 300 mA extra uit de +5 volt voeding worden afgenomen. De absolute grenzen zijn moeilijk vast te stellen. In het algemeen gezegd, hoe meer u van de voeding vraagt hoe slechter de stabilisatie wordt. Slechte stabilisatie kan een computer regelmatig op hol laten slaan, maar veroorzaakt geen schade mits u binnen de boven aangegeven grenzen blijft. Bij de 48K Spectrum verdient het aanbeveling om een aparte +5 volt voeding voor de externe schakelingen toe te voegen. De stroombegrenzings op de andere voedingsspanningen zijn gelijk aan die van de 16K uitvoering.

3.5 De -12 volt voeding

U kunt zich afvragen wat er met de -12 volt voeding gebeurd is. Hij staat aangegeven bij de randconnector in het Sinclair BASIC-handboek. Hij is echter niet aangegeven in het hoofdschema. In feite bestaat hij niet. De '-12 volt' aansluiting is fout gemerkt. In werkelijkheid moet er staan: 'ongestabiliseerde +12 volt',



Afb. 3.4 Aanvullende schakeling voor -12 volt

want hij is verbonden met de collector van TR4. Gelukkig is het eenvoudig om met de oscillerende +12 volt voeding een -12 volt voeding te maken door het toevoegen van een paar onderdelen.

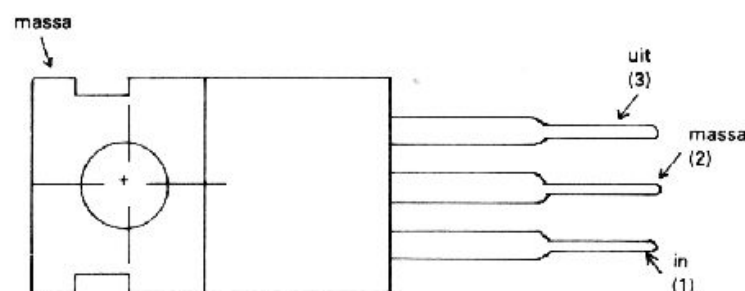
Bekijk afbeelding 3.4, u zult direct een schakeling herkennen die erg veel lijkt op die voor de -5 volt voeding in de Spectrum. Het werkt op hetzelfde 'ladingspomp'-principe. Er is geen zenerdiode in de schakeling opgenomen omdat de 33 μ F condensator in elk geval tot 12 volt oplaadt en dat is de juiste spanning. De regeling van dit type schakelingen is nooit erg goed bij belasting. De spanning stijgt tot ongeveer -10,5 volt bij een stroom van 20 mA, dit is de aanbevolen maximum stroomsterkte. Wanneer u besluit de schakeling te bouwen zorg er dan voor dat de twee condensatoren van een 16 volt type of meer zijn. Verder is geen van de onderdelen kritisch.

3.6 Extra +5 volt voeding

Voor alle schakelingen bij de 48K versie, uitgezonderd de schakelingen die zeer weinig stroom vragen (zoals de eenvoudige weerstand- en schakelaarexperimenten in dit boek), moet u een extra voeding maken. Hetzelfde geldt voor de 16K uitvoering als u meer dan 300 mA van de +5 volt voeding vraagt. In beide gevallen heeft u twee mogelijkheden:

1. Gebruik maken van de ZX-voedingseenheid

Er is wat extra stroomcapaciteit aanwezig in uw +9 volt Spectrum-voeding, zeker als de +12 volt voeding niet volledig is belast. De schrijver kon gemakkelijk tot 500 mA extra uit de ZX-voeding halen. Het schema om dit te bereiken staat in afbeelding 3.6. Alleen het deel rechts in het schema is nodig. Dit schema is in principe gelijk aan dat wat in de Spectrum wordt gebruikt. Verbind de 7805-regelaar zoals aangegeven in afbeelding 3.5, de ingangspen wordt verbonden met de +9 volt. De tekening laat de regelaar zien aan de plastic zijde waar het typenummer is aangebracht. Vergeet niet om een koelplaat aan de regelaar te bevestigen. Kleine vinvormige koelplaten zijn bij de onderdelenwinkel te koop. Een goedkope oplossing is een stuk aluminium aan de regelaar te bevestigen. Een stuk even



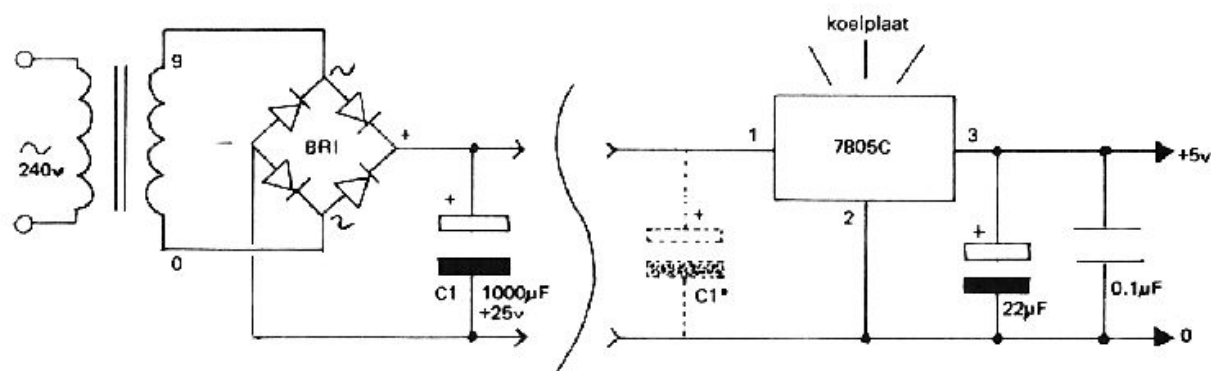
bovenaanzicht

Afb. 3.5 Aansluitpennen van de 7805C 1 amp/5 volt regelaar

groot als in uw Spectrum zal het goed doen. C1* in afbeelding 3.6 moet aangebracht worden om de stabilisatie te verbeteren. Gebruik hiervoor een 22 μF /16 volt condensator. Een tweede condensator van 22 μF moet over de uitgang worden gezet. Tevens is het verstandig enkele 0,1 μF ontkoppelcondensatoren aan te brengen. Eén bij elke twee chips is voldoende. De +5 volt die dit circuit produceert mag niet met de +5 volt voeding van de Spectrum verbonden worden. Alleen de 0 volt mag alle chips verbinden.

2. Gebruik een extra netvoedingseenheid

Het schema wordt in afbeelding 3.6 gegeven en kan ongeveer 1 ampère bij +5 volt leveren. In plaats van pen 1 van de regelaar met de +9 volt voeding van de Spectrum-voeding te verbinden, zoals in het vorige voorbeeld, moet hij met een nieuwe +9 volt voeding verbonden worden. Deze extra +9 volt voeding bestaat uit een 220 volt naar 9 volt, 1 ampère transformator, een 1 ampère bruggelijkrichtcel (BR1) en een grote 1000 μF +25 volt afvlakcondensator C1. U moet ervoor zorgen dat de transformator in een geaard metalen huis zit. De 7805 kan met het huis verbonden worden, dat als koelplaat dienst doet. Open verbindingen die de volle netspanning voeren, moeten vermeden worden om de kans op een elektrische schok te voorkomen. Behoudens de bovengenoemde veranderingen is het schema gelijk aan het in de vorige paragraaf beschreven schema. U moet ervoor zorgen dat de hoofdvoeding tegelijkertijd ingeschakeld wordt met die van uw Spectrum. Wanneer de voeding niet tegelijkertijd wordt ingeschakeld, kunnen bepaalde chips beschadigd worden, omdat ze er een hekel aan hebben wanneer er spanning op de pennen gezet wordt zonder dat hun eigen voedingsspanning aanwezig is.



Afb. 3.6 Extra +5 volt voeding

Ontkoppelcondensatoren

De voedingsspanning wordt overal op de Spectrum-print gebruikt. Als er nu een chip aan de ene kant van de print extra stroom vraagt kan er een plaatselijke spanningsdaling optreden. Als deze spanningsdaling te groot is (zelfs gedurende één microseconde), kan dat voldoende zijn om essentiële gegevens of programma's in het geheugen te vernietigen. Ontkoppelcondensatoren ten behoeve van de voeding worden daarom op strategische plaatsen op de printplaat ge-

plaatst. Zij kunnen de voeding afvlakken door gedurende een korte periode grote stromen te leveren. De spanning blijft daardoor ongeveer constant. C1 t/m C8 rondom de geheugenchips dienen hiervoor, zoals ook andere condensatoren die elders op de printplaat over de voedingslijnen staan. U moet er ook enkele opnemen in de externe schakelingen die u bouwt. Een 0,1 μ F condensator bij elke twee chips is voldoende.

4. De Z80A Centrale Verwerkingseenheid (CPU)

Dit is de microprocessorchip. Deze chip is het rekencentrum van uw Spectrum en is ondergebracht in een 40-pins plastic behuizing, aangegeven met IC2 in appendix D. De CPU kan verschillende basisfuncties uitvoeren. Deze zijn:

- Data uit het geheugen lezen.
- Data in het geheugen schrijven.
- Data lezen vanaf het invoer/uitvoer (I/U)-apparaat.
- Data schrijven naar een I/U-apparaat.
- Rekenkundige en logische bewerkingen uitvoeren met data.

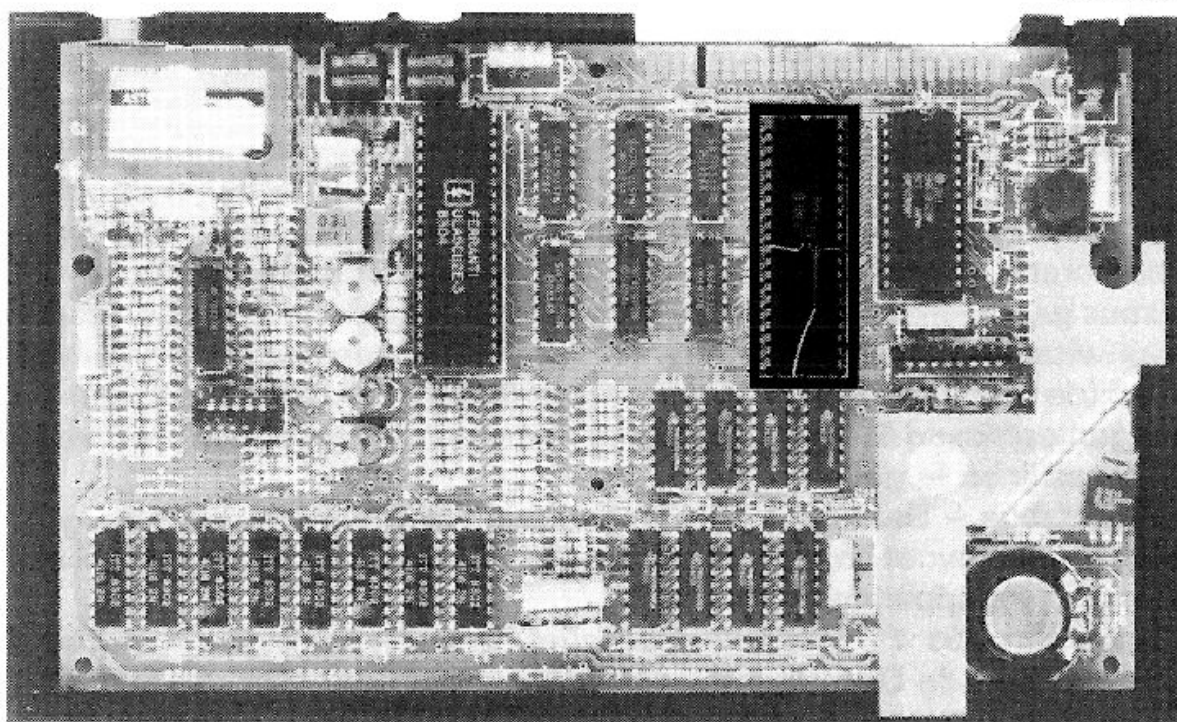
Wat de Z80A op een bepaald moment doet wordt bepaald door de instructies die hij uit het geheugen krijgt. Er zijn 158 verschillende soorten instructies die de Z80A kan begrijpen. Het programmeren van de Z80A in machinetaal is een onderwerp op zich, als u hier meer over wilt weten zult u daarop gespecialiseerde boeken moeten raadplegen. Wellicht heeft u andere computers gezien waarin de Z80-chip is gebruikt en zich afgevraagd wat het verschil is tussen de Z80 en de Z80A. Beide zijn gelijk in hun werking. Het enige verschil is dat de Z80A met snelheden tot 4 MHz kan werken terwijl de Z80 slechts tot 2,5 MHz gaat. Kort geleden is er een nieuwe chip, de Z80B, geïntroduceerd. Dit is dezelfde als de Z80-chip maar zijn snelheid gaat tot 6 MHz.

Zoals eerder gezegd, de Z80A is een 8-bit microprocessor welke per keer gegevens van 8 bits verplaatst. De eerste microprocessorchips waren 4 bit-uitvoeringen maar nu zijn er microprocessoren in 16-bit-uitvoering of zelfs in 32-bit-uitvoering. Deze zijn aanzienlijk sneller dan de 8-bit omdat ze meer gegevens in dezelfde tijd kunnen overbrengen. Ze zijn echter veel duurder zodat de 8-bit microprocessoren meer toegepast worden in thuiscomputers.

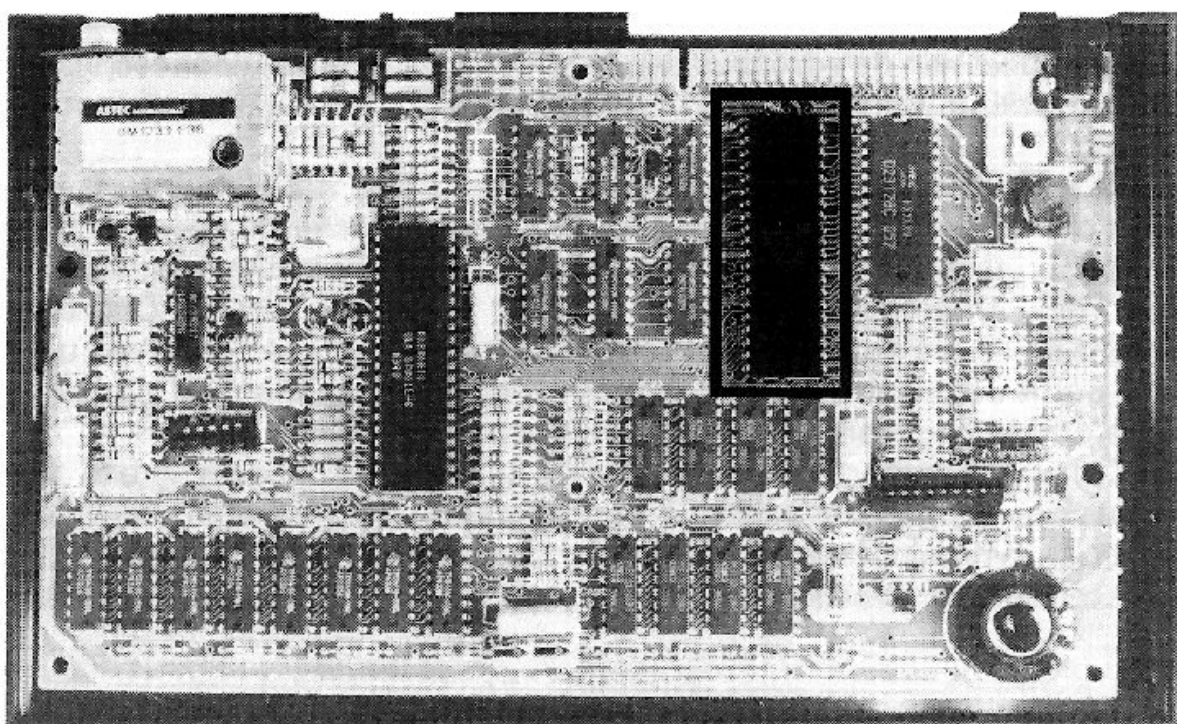
Het Z80A-kloksignaal loopt met 3,5 MHz in de Spectrum (behalve wanneer het video/programmegeheugen gebruikt wordt – zie hoofdstuk 8). Alle CPU-bewerkingen worden vergeleken met dit kloksignaal, dus de 0-1 en 1-0 overgangen moeten erg snel zijn om er zeker van te zijn dat alle bewerkingen, elke cyclus op precies hetzelfde punt beginnen. Daarvoor zijn TR3 en de bijbehorende weerstanden en diode nodig.

Het 3,5 MHz kloksignaal van de ULA heeft een onvoldoende snelle 0-1 overgang. Omdat alle interne CPU-circuits op iets verschillende voltages in het +0,8 tot +2 volt gebied kunnen beginnen wanneer de tijd waarin het kloksignaal van +0,8 naar +2 volt gaat lang duurt (in nanoseconden), is het mogelijk dat verschillende interne circuits op verschillende tijdstippen beginnen. Als er een verschil in be-

Model 2



Model 3



Afb. 4.1 Foto van het Spectrum-bord met de Z80A aangegeven

gintijd is kan de Z80A foutief werken. TR3 zorgt ervoor dat de overgang op de CPU-klokpen snel genoeg is.

4.1 Z80A CPU-penbeschrijvingen*

A0-A15 Adresbus – Driestanden (Engels: tristate)-uitgangen (0, 1 of zwevend. Als de adresbus zwevend is, kan een ander apparaat in plaats van de CPU de adresbus gebruiken). Deze bus geeft $2 \uparrow 16 = 65536$ verschillende adressen voor data-uitwisseling met het geheugen, of $2 \uparrow 8 = 256$ adressen voor in/uitvoerapparaten (de laagste acht adreslijnen worden bij I/U gebruikt). De CPU kan ook het geheugen opfrissen (zie de refresh-pen) door een geldig opfrisadres op de laagste 7 adreslijnen te geven.

D0-D7 Databus – Deze acht driestanden-tweerichtingslijnen worden gebruikt voor gegevensinvoer en uitvoer met de CPU. De overdracht gebeurt tussen geheugen of randapparaat en de Z80A.

M1 Machinecyclus 1 – uitgang, laagactief. Het signaal betekent dat de CPU op dat moment de Opcode (bewerkingscode) voor de volgende instructie die moet worden uitgevoerd uit het geheugen krijgt. $\overline{M1}$ komt ook voor te zamen met \overline{IORQ} om een onderbrekingsbevestigingscyclus aan te geven.

\overline{MREQ} Geheugenaanvraag – Driestandenuitvoer, laagactief. Dit signaal geeft aan het geheugen door dat er nu een geldig adres op de adresbus staat voor een lees- of een schrijfbewerking. Het signaal is nodig om onderscheid te maken tussen geheugen- of in/uitvoerbewerkingen.

Bijvoorbeeld, \overline{MREQ} , \overline{RD} en \overline{ROMCS} moeten allen laagactief zijn voor er een leesopdracht in de BASIC-ROM kan worden uitgevoerd (zie afbeelding 6.2).

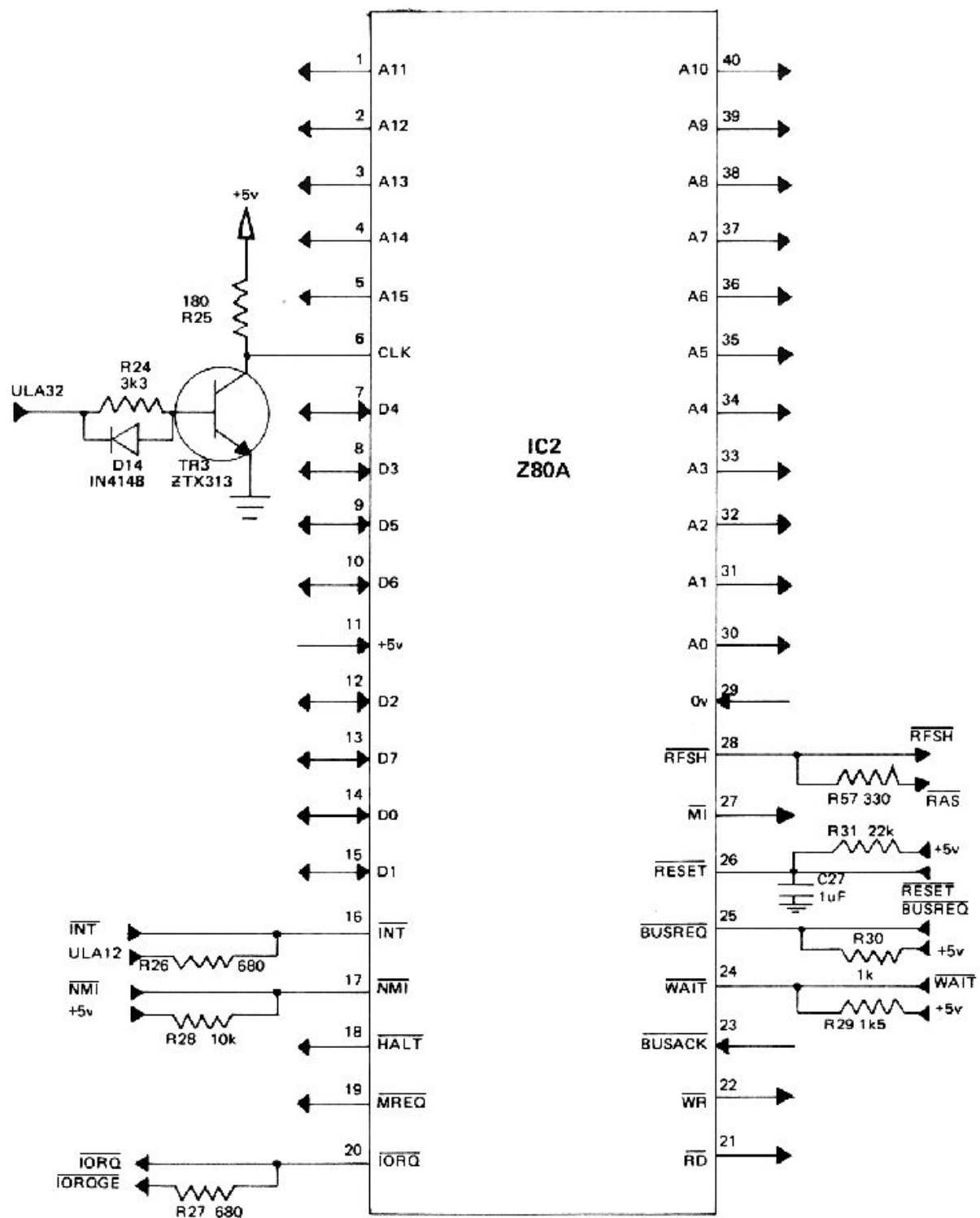
\overline{IORQ} Invoer/uitvoeraanvraag – Driestandenuitvoer, laagactief. Geeft aan dat het laagste deel van de adresbus een geldig I/U-adres bevat voor een I/U-schrijf of leesbewerking. Denk erom dat \overline{IORQ} en \overline{MREQ} nooit op hetzelfde moment actief zijn.

\overline{RD} Lees – Driestandenuitvoer, laagactief. Geeft aan dat de CPU uit het geheugen of van een I/U-apparaat wil lezen. Het geadresseerde apparaat moet dit signaal gebruiken om de juiste gegevens op de CPU-databus te zetten.

\overline{WR} Schrijf – Driestandenuitvoer, laagactief. Geeft aan dat de CPU-databus gegevens bevat die op een geadresseerde plaats opgeslagen of op een uitgekozen I/U-poort uitgevoerd moeten worden. Het geheugen dient dit signaal te gebruiken om de gegevens van de databus op te slaan.

\overline{RFSH} Opfris – Uitvoer, laagactief. Geeft aan dat de laagste 7 bits van de adresbus een opfrisadres (refresh) voor dynamische geheugens bevatten. Dit wordt door dynamische geheugens gevraagd om hun gegevens niet te verliezen. Als ze niet elke twee milliseconden opgefrist worden bestaat het gevaar dat

*Merk op dat signalen waar een streep boven staat zoals \overline{RD} 'laagactief' zijn. Dit betekent dat er een leesopdracht uitgevoerd wordt als $\overline{RD} = 0$. Gewoonlijk is 1 = waar en 0 = niet-waar, maar alle Z80A-controlesignalen met een streep erboven zijn omgekeerd dus 0 = waar en 1 = niet-waar.



Afb. 4.2 De Z80A-centralevverwerkings-chip (CPU)

de geheugeninhoud verloren gaat. Als het signaal aanwezig is bevatten de bovenste 8 adreslijnen de inhoud van het CPU-I-register. Dit geeft een fout in de Spectrum-hardware. Door het I-register een waarde tussen 64 en 127 te geven ontstaat er een interessant conflict voor het video-RAM. A15 en A14 van de CPU selecteren de 16K RAM waar de ULA prioriteit over heeft. In dit geval geldt, ofschoon het relevante adres verschijnt met $\overline{\text{MREQ}}$ actief, zal er geen $\overline{\text{RD}}$ - of $\overline{\text{WR}}$ -signaal optreden omdat het een opfrissing is. Deze combinatie van sig-

nalen verwacht de ULA waardoor de CPU-klok niet op de juiste manier gestopt wordt. Een BASIC-programma dat dit illustreert staat in hoofdstuk 8 dat over de ULA gaat.

HALT Uitvoer, laagactief. Geeft aan dat de CPU een software HALT-opdracht uitvoert. Hij verwacht een onderbreking van een ander apparaat voordat het doorgaat met zijn werking. Er staat een eenvoudige proef die de werking van de HALT demonstreert in hoofdstuk 14.

WAIT-Invoer, laagactief. Wordt gebruikt door langzame geheugens of I/U-apparatuur om de CPU aan te geven dat ze nog niet klaar zijn om data over te brengen. De CPU wacht rustig af tot het langzame apparaat aangeeft dat het gereed is.

INT Onderbrekingsaanvraag – Invoer, laagactief. Dit signaal kan door externe apparatuur opgewekt worden om de CPU een speciaal machinecodeprogramma, dat ergens in het geheugen staat, af te laten werken. Als de interne door software gecontroleerde onderbrekings-flip-flop is ingeschakeld, zal de CPU de onderbreking accepteren en bevestigen. Een flip-flop is gelijk aan één geheugenbit. De CPU gebruikt dit bit van zijn inwendige geheugen om te onthouden of het al of niet onderbrekingen van andere apparatuur zal aannemen. Meer informatie over het gebruik van onderbrekingen kunt u vinden in een boek dat de Z80 behandelt. In hoofdstuk 14 wordt een eenvoudige proef gegeven over het gebruik van de onderbreking.

NMI Niet te maskeren onderbreking – Invoer die werkt op de negatief gaande flank van het onderbrekingssignaal. De NMI wordt altijd door de CPU geaccepteerd aan het einde van de lopende opdracht. Hij dwingt de CPU een programma uit te voeren dat op geheugenadres 102 decimaal (66 hex) begint. Hoofdstuk 14 geeft een eenvoudig voorbeeld.

RESET-Invoer, laagactief. Dit signaal dwingt de CPU in de herstelstatus. In de Spectrum, als de spanning wordt ingeschakeld, houdt C27 de herstellpen laag totdat hij via R31 is opgeladen. Hierdoor komt de rest van de computer in een werkbare staat voordat de CPU het programma vanaf adres 0 gaat uitvoeren. Details hoe een drukknop aan de Spectrum moet worden verbonden om de Spectrum te herstellen staan in hoofdstuk 14.

BUSREQ Busaanvraag – Invoer, laagactief. Dit wordt door externe apparatuur gebruikt om aan de adresbus, databus en de driestandenuitvoer van de CPU controlesignalen te vragen voor eigen gebruik. De CPU geeft de controle over zodra de lopende machinecyclus beëindigd is. Om het vragende apparaat aan te geven dat de controle kan worden overgenomen, maakt de CPU de BUSACK-uitvoer laag.

BUSACK Busbevestiging – Uitvoer, laagactief. Geeft aan dat het vragende apparaat de volledige controle over alle CPU-bussen kan overnemen.

Meer informatie over deze Z80-signalen en hoe ze gebruikt kunnen worden staat in andere hoofdstukken. In het bijzonder wordt u verwezen naar de hoofdstukken 8, 12 en 14.

5. Video- en programmegeheugen

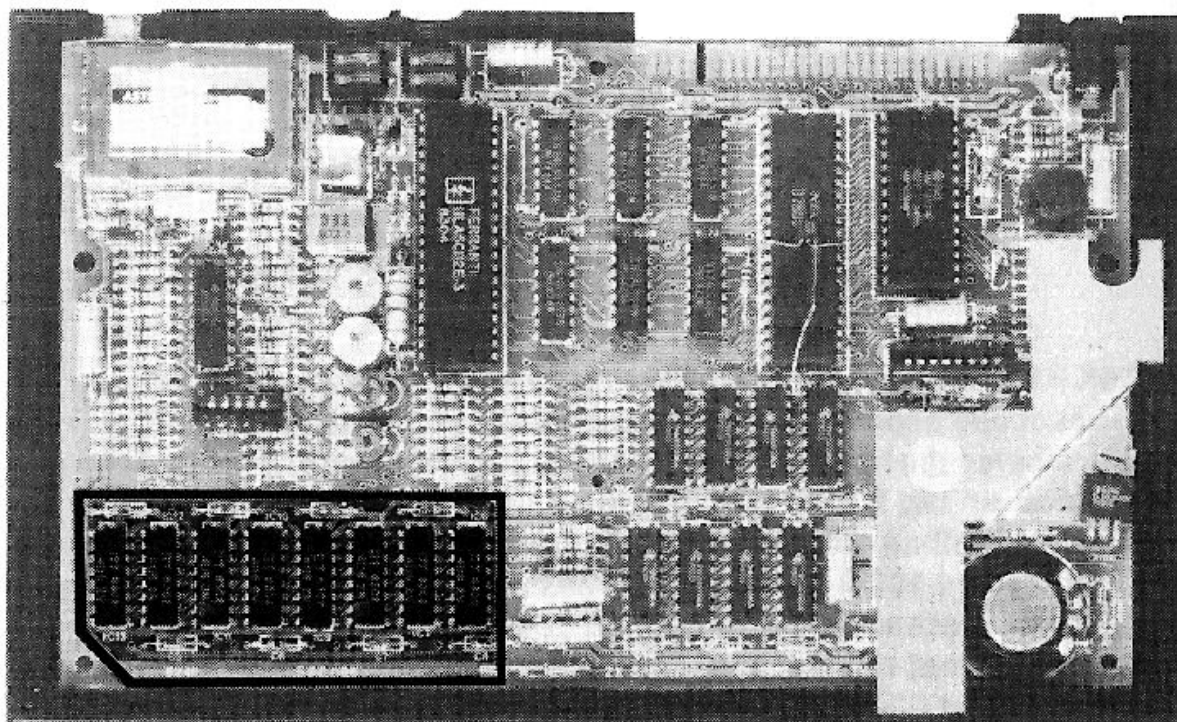
Deze 16K RAM is zowel op de 16K- als op de 48K-versie van de Spectrum aanwezig. Hij bevat alle gegevens voor het opmaken van het schermbeeld, de verschillende variabelen die het BASIC nodig heeft, de werkruimte, de gebruikersbepaalde 'graphics' en uw BASIC-programma.

Bekijken we nu afbeelding 5.2. Elk van de geheugenchips IC7 t/m IC14 kan $2 \uparrow 14 = 16K$ (K betekent 1024) bits aan gegevens bewaren. Acht zijn er nodig om de 8-bit databus te maken. Daardoor kunnen de geheugens in kleinere chips worden ondergebracht met minder aansluitingen, de adreslijnen zijn gemultiplext. Dit betekent dat eerst A0 t/m A6 aan de geheugenchip worden aangeboden, gevolgd door A7 t/m A14.

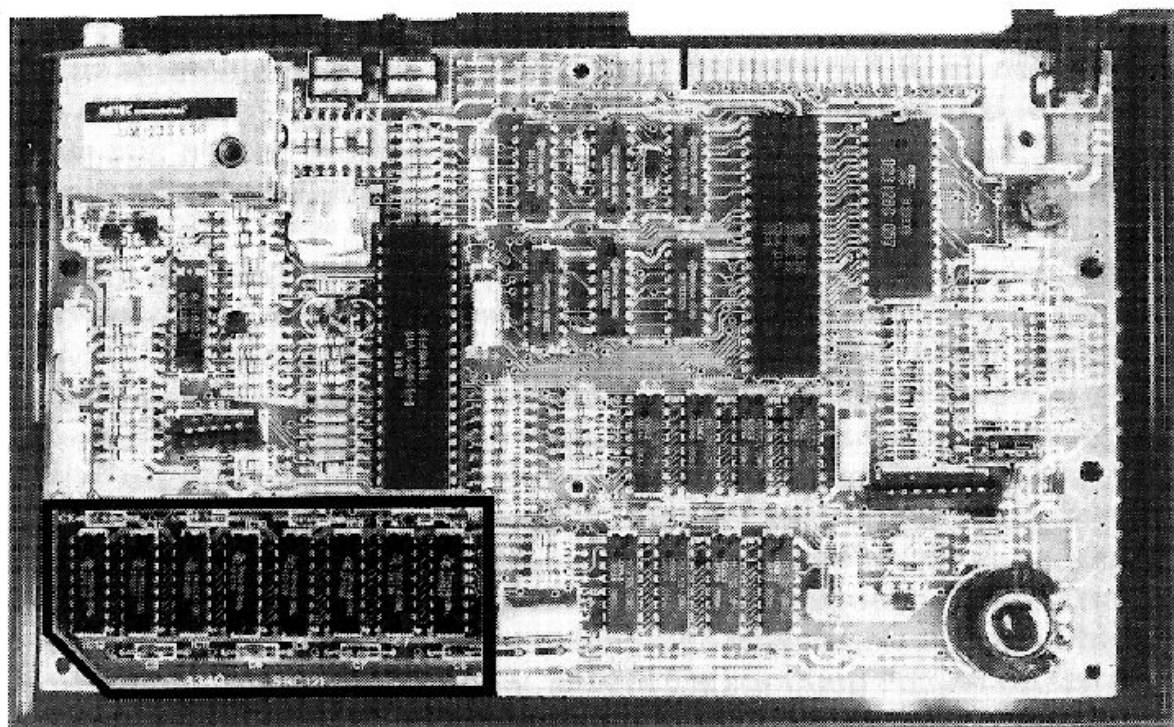
Deze twee sets van 7 bits worden binnen de geheugenchip gekoppeld, zodat die de juiste plaats kan opzoeken. IC3 en IC4 zorgen voor deze multiplexing, de gekozen adreslijnen worden verbonden met de invoer van de geheugenchip, deze wordt bepaald door de toestand van pen 1 op IC3 en IC4. DRAM A0 t/m DRAM A6 op de ULA kunnen de uitvoer van IC3 en IC4 overstemmen vanwege de 330 ohms weerstanden in serie met de multiplexeruitvoer. Deze mogelijkheid tot overstemmen stelt de ULA in staat om gegevens voor de video-uitvoer te krijgen als dat nodig is. U kunt zich afvragen wat er gebeurt als zowel de ULA als de CPU toegang tot het geheugen vragen. Het is duidelijk dat ze beiden niet twee verschillende geheugenplaatsen tegelijkertijd kunnen bereiken. De oplossing van dit conflict wordt in hoofdstuk 8 beschreven.

Het soort geheugen dat gebruikt is wordt 'dynamisch willekeurig toegankelijk geheugen' (DRAM) genoemd. De interne geheugenmatrix is opgebouwd uit cellen gerangschikt in 128 rijen en 128 kolommen. De rij-adres-puls (RAS)- en de kolom-adres-puls (CAS)-signalen worden gebruikt om de juiste adressen in de geheugenchip in te schakelen. Daarna kunnen de gegevens in die geadresseerde plaats worden geschreven of daaruit worden gelezen. Elke rij moet elke 2 ms worden ingeschakeld anders vergeet het geheugen de opgeslagen gegevens. In de Spectrum is dit opfrissen geen probleem wanneer er video-uitvoer plaatsvindt, omdat het videogeheugen regelmatig wordt uitgelezen om een continu TV-beeld te maken. Gedurende de videobeeld-terugslag (beeldveldsynchronisatie), als het geheugen gedurende 5 ms niet wordt uitgelezen, neemt de normale CPU-opfrissing het over. Geheugen waar zowel uit gelezen als in geschreven kan worden, maar dat geen voortdurende opfrissing vraagt wordt statische RAM genoemd. Bij beide types RAM gaan de gegevens verloren wanneer de spanning wordt uitgeschakeld.

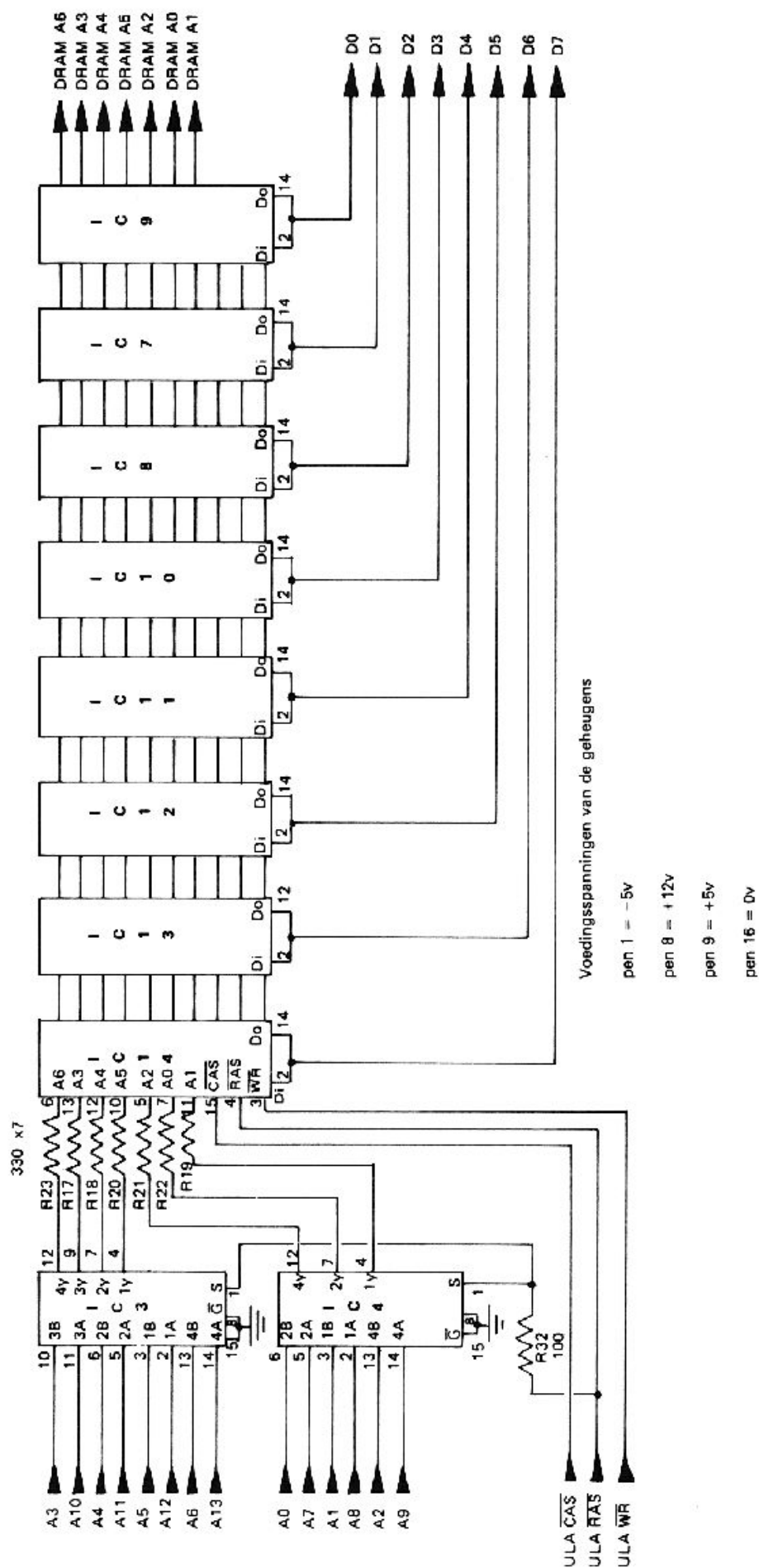
Model 2



Model 3



Afb. 5.1 Foto van het Spectrum-bord met de RAM aangegeven



Afb. 5.2 Blokschema van het video- en RAM-geheugen

6. Het BASIC 'alleen te lezen' geheugen (ROM)

IC5 is een 16K byte 'alleen te lezen geheugen' (ROM)-chip. Opgeborgen in een 28-pens behuizing is deze chip voorzien van 14 adreslijnen, 8 datalijnen, twee chipkeuzelijnen, uitvoerinschakellijn, om de CPU gegevens te kunnen laten lezen en natuurlijk de voedingsaansluitingen. De opstelling van de pennen is in afbeelding 6.2 weergegeven.

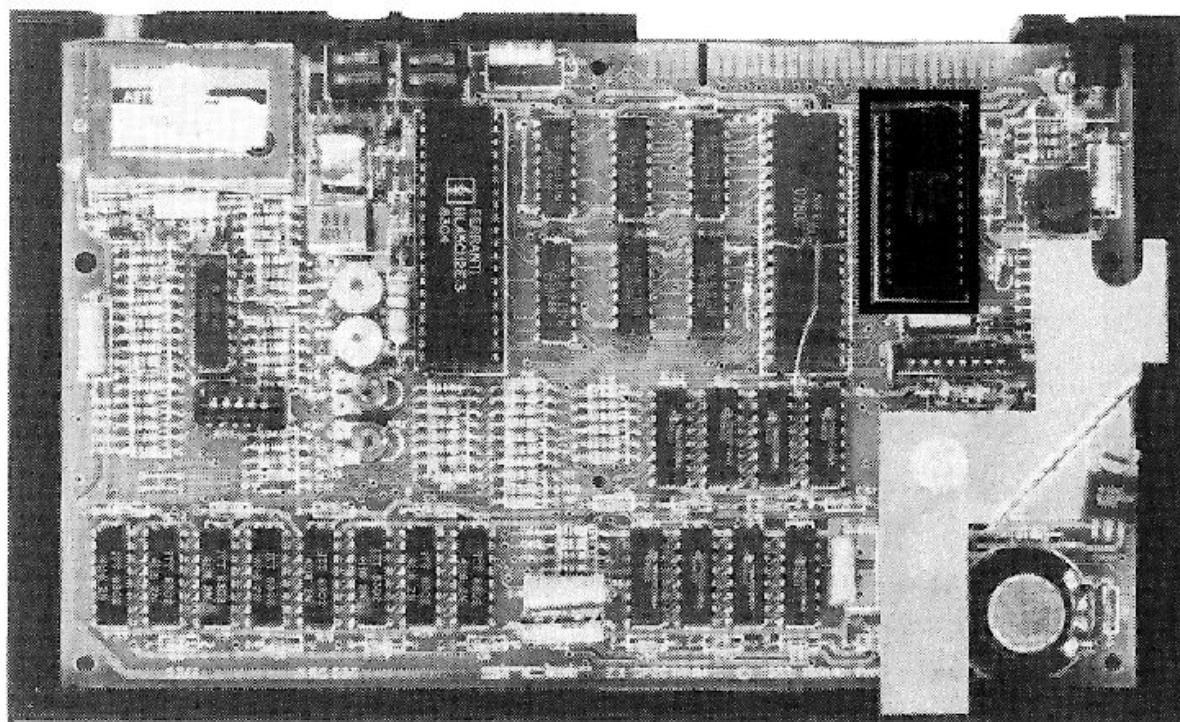
De ROM bevat het programma dat de CPU vertelt hoe het Sinclair BASIC moet worden uitgevoerd. Dit programma is tijdens de productie vastgelegd in de siliciumstructuur van de chip en kan niet worden veranderd. Als dit niet het geval was dan zou het BASIC verdwijnen wanneer de spanning wordt uitgeschakeld. Dus, omdat u het BASIC-programma niet kunt veranderen, kunt u nooit door het intoetsen van software in uw Spectrum het BASIC beschadigen.

De ROM begint op adres nul. Hij moet hier beginnen, omdat wanneer de CPU hersteld wordt (bijvoorbeeld bij het inschakelen), de CPU altijd met het machine-codeprogramma op adres nul begint.

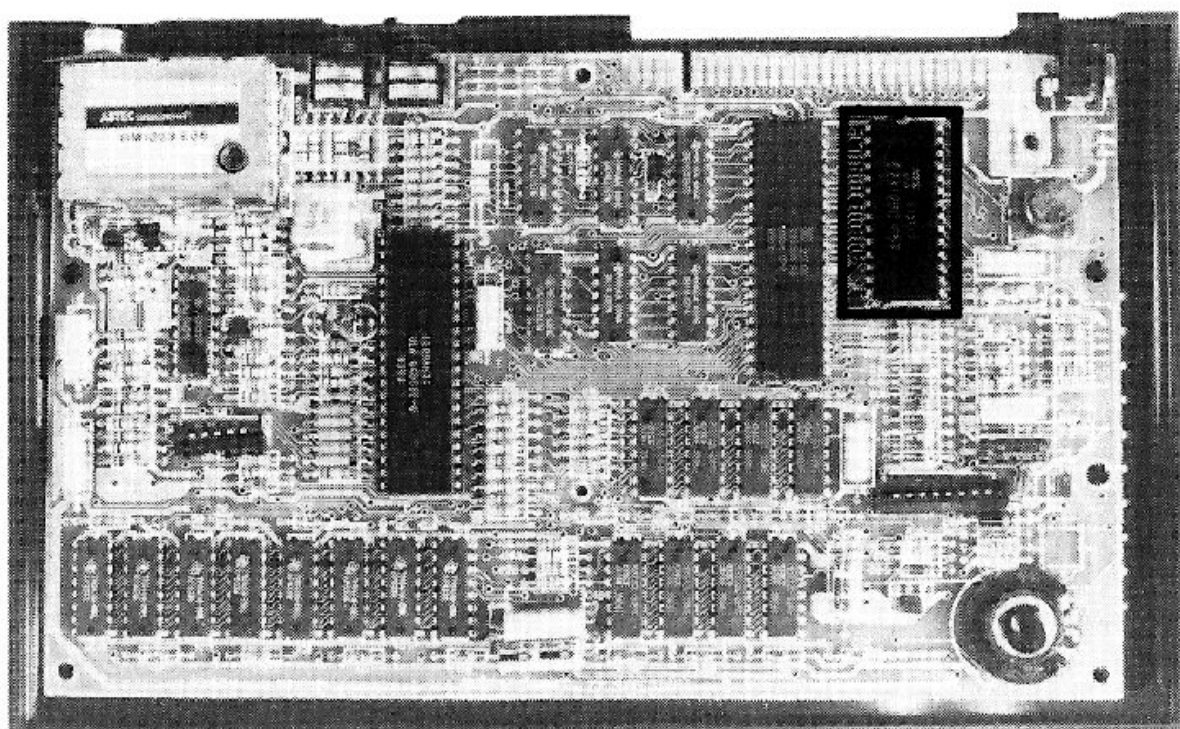
De ROMCS-aansluiting achterop de randconnector kan direct met de +5 volt verbonden worden om de BASIC-ROM te blokkeren. De ROMCS-uitvoer van de ULA is via R33 verbonden en kan daardoor het ROMCS-sigitaal niet laag maken wanneer de +5 volt verbinding is aangebracht. Dit is gemakkelijk voor toekomstige toevoegingen die hun eigen ROM of RAM hebben, verschillende talen kunnen dan op de Spectrum gebruikt worden in plaats van BASIC. Het op deze manier vervangen van de BASIC-ROM vraagt een goed begrip van de Spectrum hardware en software. Er moet dan een compleet nieuw besturingssysteem (operating system) worden geschreven.

Er zijn chips, EPROMs (uitwissbare programmeerbare 'alleen te lezen' geheugens) beschikbaar met dezelfde aansluitingen als de Sinclair-ROM. De 27128 EPROM is zo'n chip. Hij bevat 16K-byte aan geheugen. Het verschil is dat hij door de gebruiker geprogrammeerd kan worden. Is hij eenmaal geprogrammeerd dan houdt hij de gegevens vast net als een ROM, zelfs als de spanning wordt uitgeschakeld. EPROMs kunnen gewist worden met ultraviolet licht, waardoor ze opnieuw geprogrammeerd kunnen worden. Dit UV-licht schijnt door een glazen venster bovenop de chip direct op het siliciumplaatje. Deze mogelijkheid is handig omdat daardoor dezelfde chip steeds opnieuw voor verschillende programma's kan worden gebruikt.

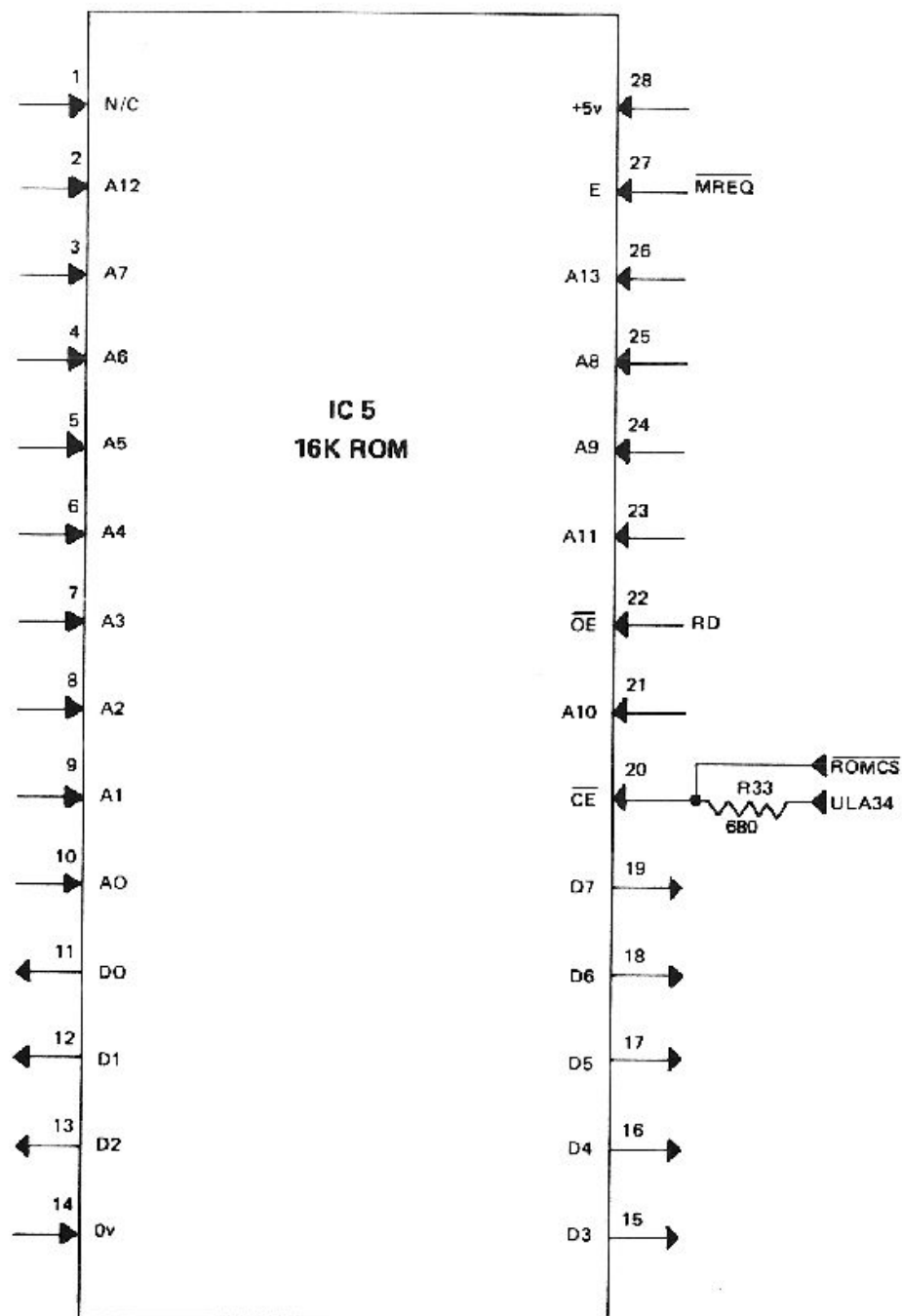
Model 2



Model 3



Afb. 6.1 Foto van het Spectrum-bord met de ROM aangegeven

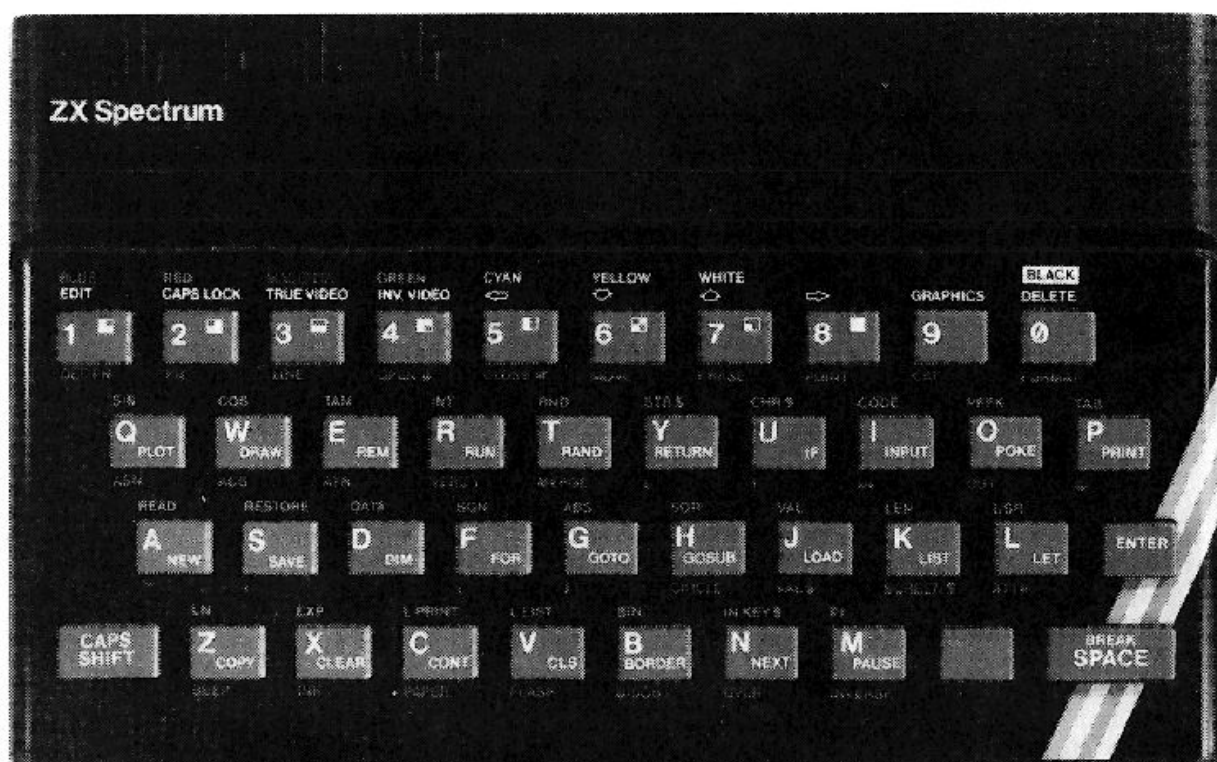


Afb. 6.2 Penaansluitingen van het BASIC-ROM

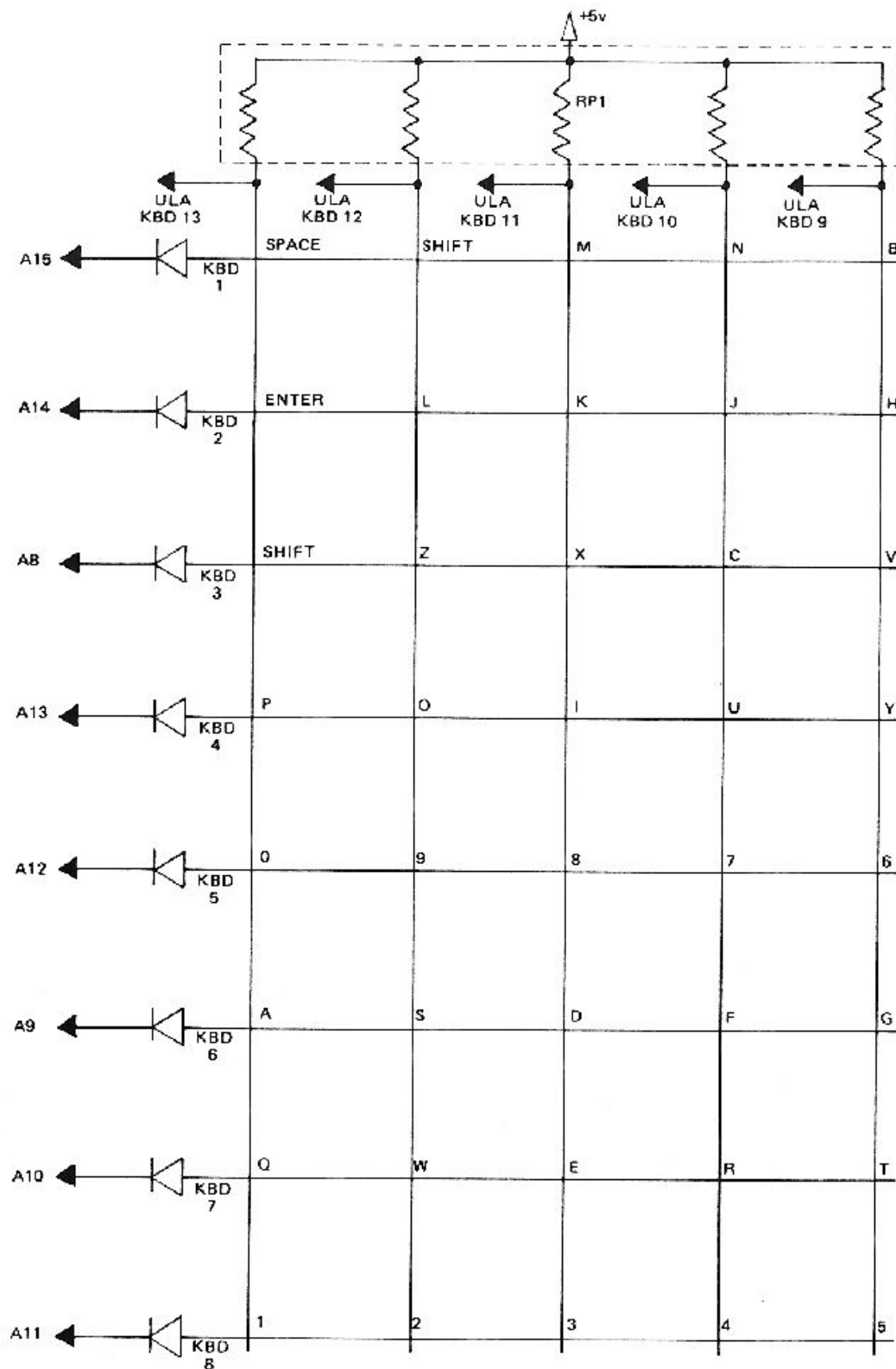
7. Het toetsenbord

Het toetsenbord is in feite een rooster van vijf bij acht draden, de kruispunten worden verbonden door op de bijbehorende toets te drukken (zie afbeelding 7.2). Elk van de acht rijen worden op hun beurt geselecteerd door de bijbehorende adreslijn laag (logisch 0) te maken, terwijl de rest logisch 1 blijft. De vijf kolommen blijven normaal logisch 1 door de weerstanden in RP1. Als één van de toetsen op een gekozen rij wordt ingedrukt, zal die kolom invoer van de ULA naar logisch 0 gaan in plaats van zijn normale logische 1-status. Zo kan de toets worden uitgelezen.

Laten we aannemen dat toets 'D' is ingedrukt. Door een IN-instructie uit te voeren op een adres waarbij alle lijnen logisch 1 zijn met uitzondering van A9, zal bit D2 van het invoerbyte 0 zijn terwijl D0, D1, D3 en D4 allen logisch 1 zijn. Het toetsenbord wordt 50 maal per seconde uitgelezen om te zien of er een toets is ingedrukt, dus alles wat u intoetst wordt door de Spectrum gezien. Het aftasten van het toetsenbord wordt door de ULA gestart door de CPU te onderbreken aan het einde van het afbeelden van een TV-beeld. De CPU leest dan het toetsenbord.



Afb. 7.1 Foto van het toetsenbord



Afb. 7.2 Schema van het toetsenbord

Als u om de een of andere reden zelf het toetsenbord wilt uitlezen, kunnen de hierna volgende adressen worden gebruikt. Denk erom dat elk van deze getallen gebruikt wordt om alle adreslijnen, behalve die afgetast wordt, op het logische

1-niveau te brengen. Door deze manier voor het uitlezen van toetsen te gebruiken, is het lezen van het toetsenbord niet meer beperkt tot 50 maal per seconde. Het kan zo vaak (binnen de grenzen van het mogelijke) of zo zelden gedaan worden als nodig is.

'IN 32766' gebruikt A15 om van SPACE tot B te lezen.

'IN 49150' gebruikt A14 om van ENTER tot H te lezen.

'IN 57342' gebruikt A13 om van P tot Y te lezen.

'IN 61438' gebruikt A12 om van 0 tot 6 te lezen.

'IN 63486' gebruikt A11 om van 1 tot 5 te lezen.

'IN 64510' gebruikt A10 om van Q tot T te lezen.

'IN 65022' gebruikt A9 om van A tot G te lezen.

'IN 65278' gebruikt A8 om van SHIFT tot V te lezen.

In de gelezen byte is:

D0 = logisch 1 bij KBD13 invoer

D1 = logisch 1 bij KBD12 invoer

D2 = logisch 1 bij KBD11 invoer

D3 = logisch 1 bij KBD10 invoer

D4 = logisch 1 bij KBD 9 invoer

D5 = niet gebruikt

D6 = logisch 1 bij gebruik van de EAR-aansluiting

D7 = niet gebruikt

Details voor het maken van uw eigen toetsenbord-interface dat op de randconnector kan worden aangesloten staan in hoofdstuk 16.

8. Het niet-gebonden logische netwerk (ULA)

De ULA (Uncommitted Logic Array) is een speciale chip van Sinclair. Hij is ontworpen om vele kleine chips te vervangen die in oudere computers gebruikt werden. De manier waarop hij werkt is vastgelegd bij de produktie en kan nooit door software veranderd worden.

De ULA heeft tot taak de CPU te ontlasten van in- en uitvoerwerkzaamheden. Hij verzorgt de vervelende uitvoer van informatie naar het televisiescherm zolang de Spectrum ingeschakeld is. Naast deze hoofdtak zorgt de ULA ook voor de uitvoer naar de zoemer en de cassette en voor de invoer vanaf het toetsenbord en de cassette.

8.1 ULA-penbeschrijvingen

DRAM A0 t/m DRAM A6 Van de geheugen-adresmultiplexerchips IC3 en IC4. Deze stellen de ULA in staat te bepalen welk adres gekozen wordt door de CPU. Ze worden door de ULA ook gebruikt om het adres voor de videogegevens uit het videogeheugen te kiezen. Als de ULA ze als uitvoer gebruikt kunnen ze de uitvoer van de adresmultiplexers overstemmen door de serie van 330 ohm weerstanden.

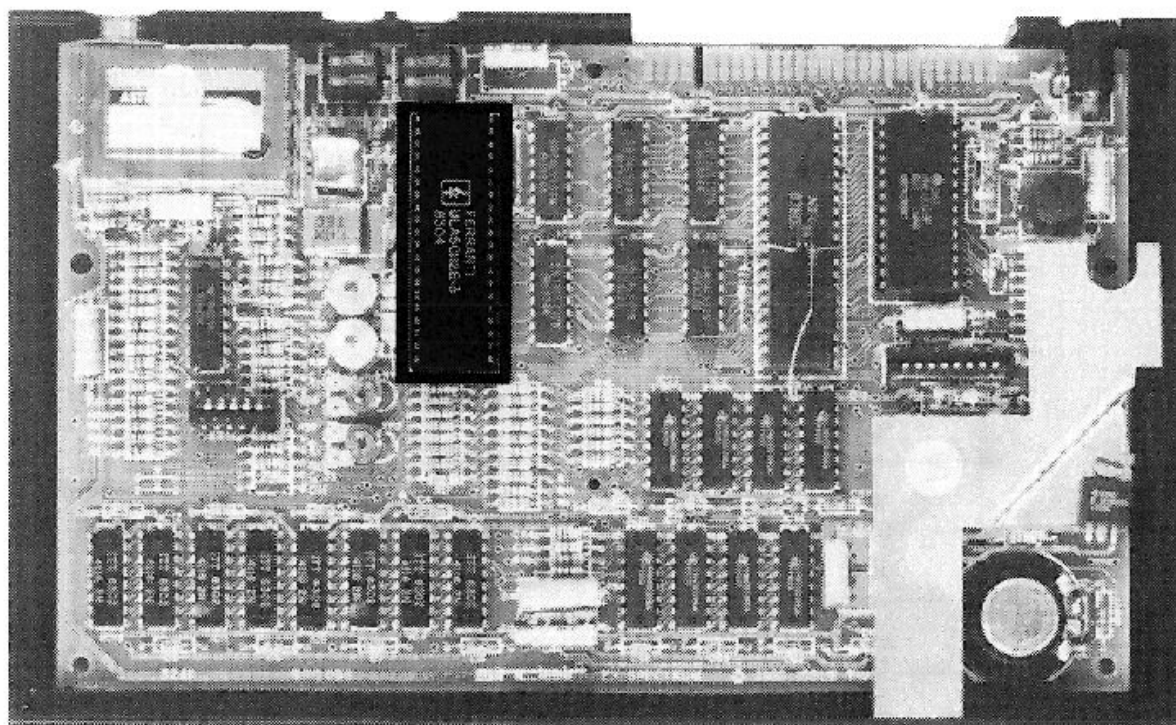
DRAMCAS Dynamische RAM-kolomadrespuls – Uitvoer. Wordt gebruikt om het kolomadres voor de dynamische RAM uit de adresmultiplexer te koppelen (zie hoofdstuk 5 voor de details).

ROMCS ROM-chipselectie – Uitvoer. Schakelt de 16K ROM-chip IC5 in als de CPU daaruit wil lezen. De ULA kan dit controleren met zijn A14 en A15 adresinvoerlijnen, waardoor hij video-informatie kan uitvoeren terwijl de CPU uit de ROM leest.

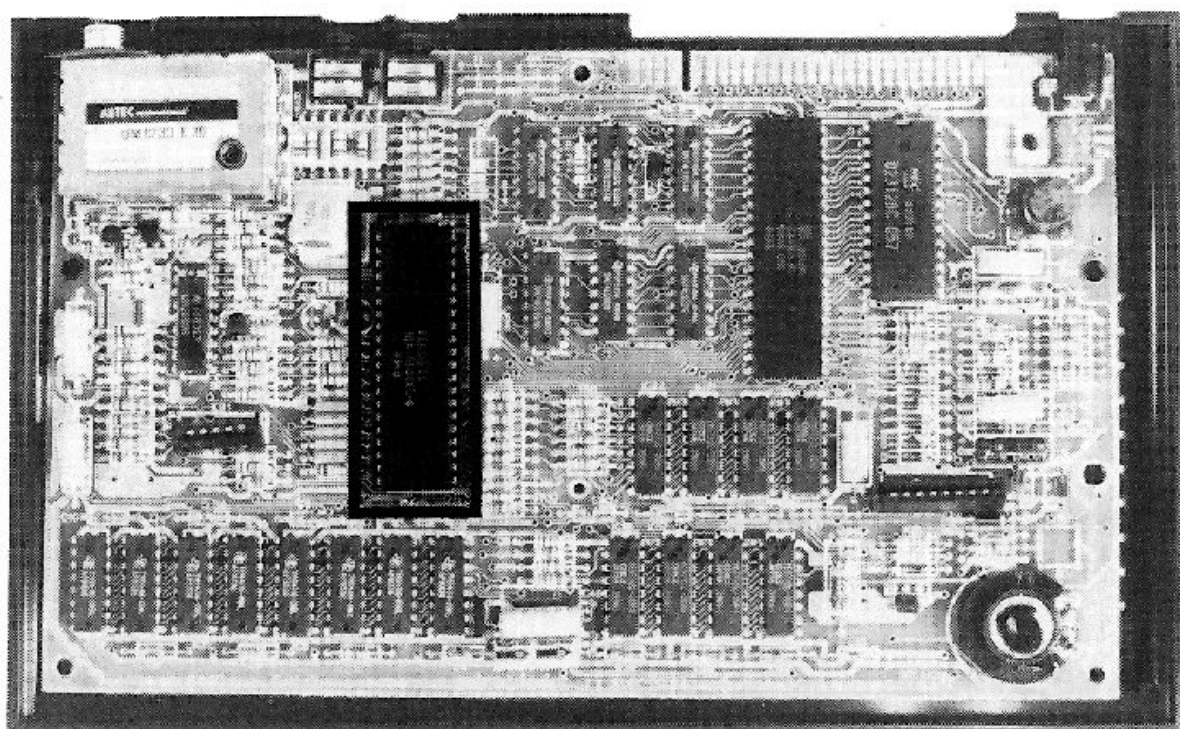
IORQGE – Invoer. Verbonden met de Z80A- $\overline{\text{IORQ}}$ -pen via R27. Wanneer deze $\overline{\text{IORQGE}}$ -verbinding hoog gehouden wordt door hem met de +5 volt te verbinden, bereikt het Z80A- $\overline{\text{IORQ}}$ -signaal de ULA niet meer. Dit kan gebruikt worden om extra I/U-apparaten toe te voegen. Zie hoofdstuk 15 voor meer details.

RAS – Gebruikt als rij-adrespuls bij de dynamische RAM-chips (zie hoofdstuk 5 voor meer details over het opfrissen van dynamische geheugens). Deze pen is ook verbonden met de ZX80A-opfrissen via R57 (330 ohm). De geheugens kunnen dan door de CPU opgefrist worden gedurende de beeldterugslagtijd, als het regelmatige gebruik door de ULA gedurende 5 ms stopt.

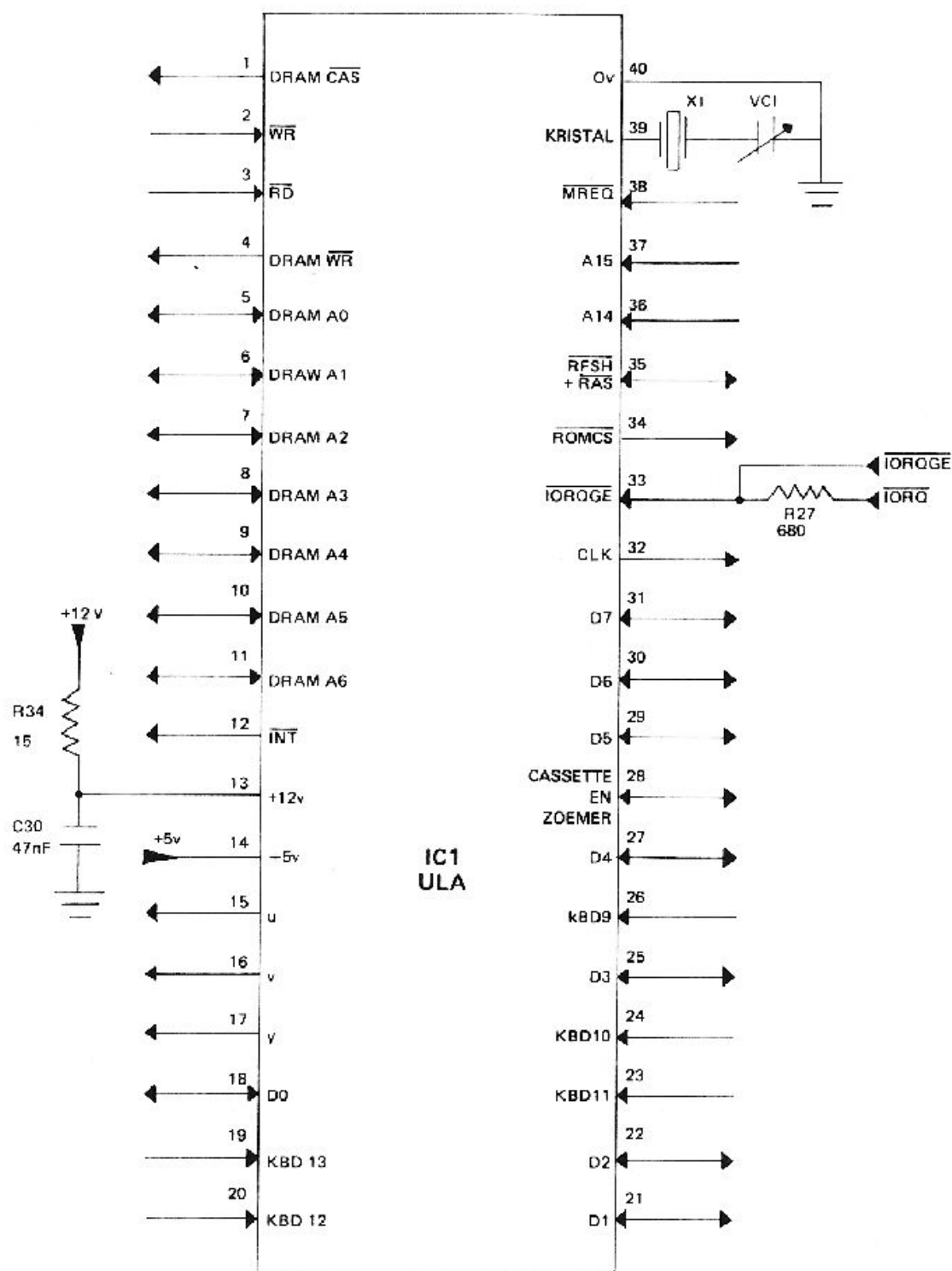
Model 2



Model 3



Afb. 8.1 Foto van het Spectrum-bord met de ULA aangegeven



Afb. 8.2 Penaansluitingen van het niet gebonden logische netwerk (ULA)

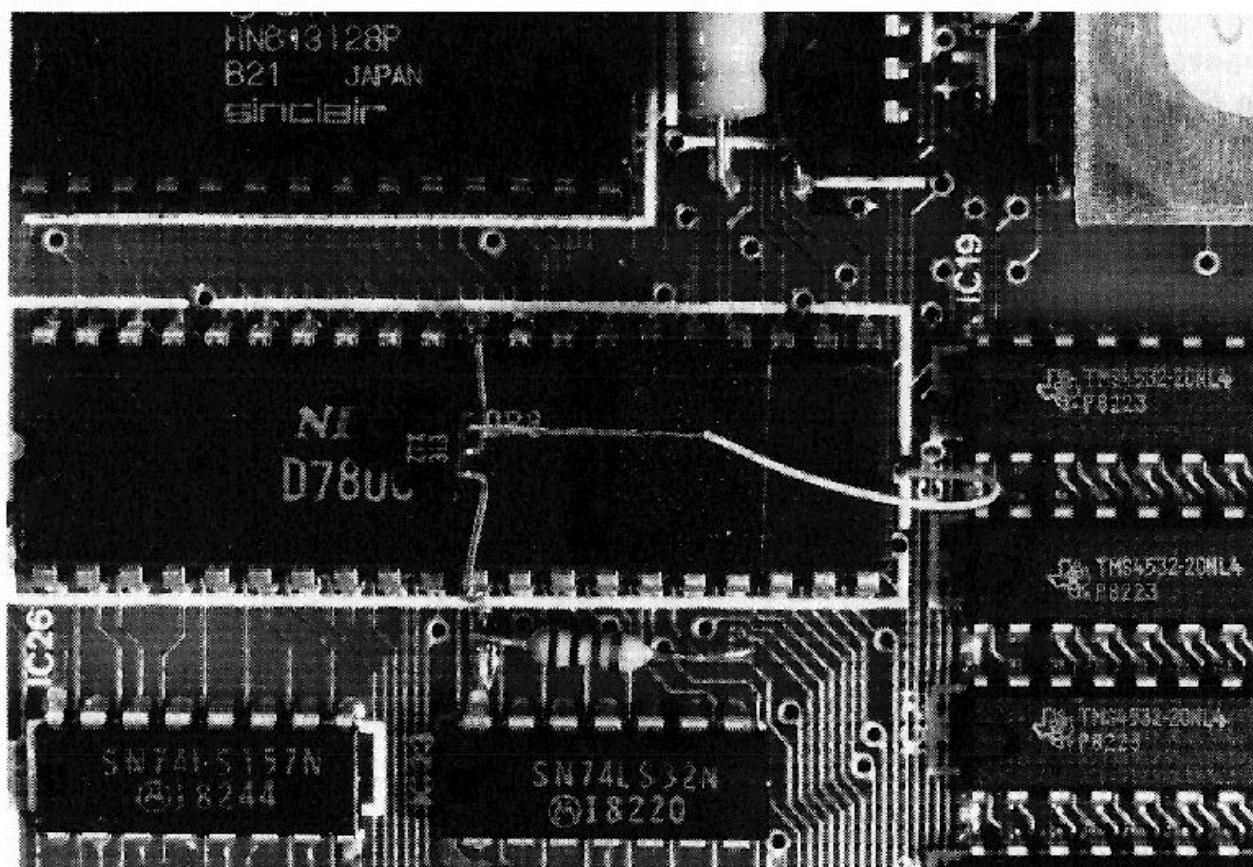
KB9 t/m KB13 – Invoer in het Spectrum-toetsenbord. Zie hoofdstuk 7 voor nadere details.

U Blauw-geel kleurverschiluitgang.

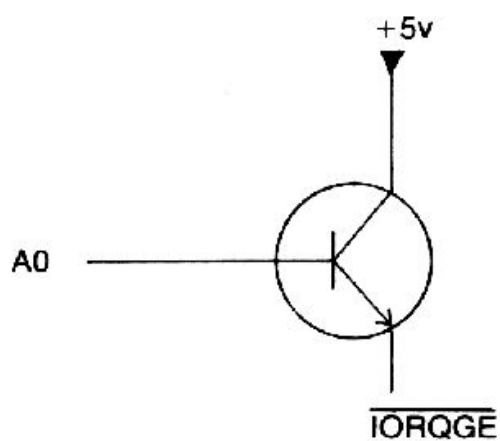
V Rood-geel kleurverschiluitgang.

Y Helderheid en synchronisatie-uitgang voor de video.

D0 t/m D7 – 8-bit tweerichtings-databus. Direct verbonden met de videogeheue-



Afb. 8.3 Foto van de versteltransistor voor model 2



Afb. 8.4 Schema van de versteltransistor voor model 2

gen-databus en via 470 ohm weerstanden met de hoofdsysteem-databus. Dit maakt de asynchrone werking mogelijk van de ULA die het video-RAM leest en de CPU die in de rest van het geheugen schrijft mogelijk. De computer werkt daardoor sneller dan wanneer de ULA elke cyclus moet stoppen.

CLK – 3,5 MHz klokuitvoer naar de Z80A-CPU. Deze klok kan nu en dan door de ULA gestopt worden om te voorkomen dat de CPU in de video-RAM schrijft terwijl de ULA erin bezig is.

\overline{WR} – Deelt aan de ULA mee dat de CPU naar een of ander apparaat schrijft. Als dat apparaat de ULA is, dan gebruikt de ULA dit signaal om de binnenkomende gegevens op te nemen.

\overline{RD} – Vertelt de ULA dat de CPU van een of ander apparaat leest. Als dat de ULA is worden de gegevens op de databus gezet ten behoeve van de CPU.

\overline{MREQ} – Deze invoer meldt de ULA dat er een geldig adres op de adresbus staat voor een lees- of schrijfbewerking. Het signaal is nodig om onderscheid te maken tussen geheugen en I/O-bewerkingen.

\overline{INT} – Onderbreking van de CPU-werking, 50 maal per seconde. Wanneer dit signaal ontvangen wordt, verhoogt de CPU een 2-byteteller in het geheugen en zoekt het toetsenbord af of er een toets is ingedrukt.

8.2 De versteltransistor in model 2

In bepaalde series van de ULA zit een fout. In de model 2 uitgave van de Spectrum wordt dit probleem opgelost door een transistor die over de Z80A gesoldeerd is.

De verbindingen met de transistor zijn afgebeeld in afbeelding 8.4. Het effect van dit 'verstellapje' is om de \overline{IORQGE} -invoer van de ULA hoog te maken wanneer de adreslijn A0 logisch 1 is. De ULA wordt daardoor alleen geselecteerd als A0 en \overline{IORQ} van de Z80A beiden laag zijn.

8.3 Video-uitvoer

Een van de belangrijkste problemen bij de meeste computers is het maken van het TV-beeld. In het bijzonder een kleurenbeeld met het oplossend vermogen zoals dat in de Spectrum gebruikt wordt; gegevens moeten voortdurend en met grote snelheid uit het videogeheugen naar het scherm gekopieerd worden. Dit geeft problemen als de CPU tegelijk met de ULA uit het videogeheugen wil lezen. Twee apparaten kunnen niet tegelijkertijd verschillende plaatsen in het geheugen adresseren.

De meeste andere computers gebruiken één van de twee manieren om dit op te lossen. Bij de eerste manier heeft de CPU voorrang. Dit veroorzaakt 'sneeuw' op het scherm, hiermee wordt video-informatie aangegeven welke niet afgebeeld kon worden omdat de CPU het geheugen gebruikte. Sneeuw is vervelend maar de CPU werkt op topsnelheid. De andere manier geeft voorrang aan het videocircuit gedurende het opbouwen van een beeld. Hierdoor kan de CPU alleen werken

gedurende de beeldterugslagtijd. Bij deze manier is er geen sneeuw, maar de computer werkt erg langzaam. Geen van deze manieren voldoet.

Uw Spectrum gebruikt een zeer vernuftige manier om het probleem te omzeilen. We nemen aan dat de ULA toegang tot het videogeheugen heeft. De CPU kan tegelijkertijd toegang hebben tot de ROM of de extra 32K RAM zonder dat er een busconflict ontstaat. De ULA en het videogeheugen hebben adres- en databussen die gescheiden zijn van de rest van het systeem door respectievelijk 330 ohm en 470 ohm serieweerstanden. Daardoor zijn er gewoonlijk twee onafhankelijk werkende systemen; de ULA die video uitvoert, de Z80A die BASIC uitvoert. Op het moment dat de CPU de 16K RAM wil gebruiken die de ULA gebruikt, omdat het de BASIC-variabelen zowel als het videogeheugen bevat, realiseert de ULA zich wat er gaat gebeuren (aangegeven door A14 en A15) en stopt snel de klok van de CPU. De Z80A merkt dit niet omdat zijn enige manier om de tijd te meten is door aan te nemen dat de klokinvoer constant is. De ULA geeft dan de CPU gedurende enkele honderden nanoseconden toegang tot het geheugen als er een kleine onderbreking in de video-uitvoer is.

Hoe heeft dit nieuwe ontwerp invloed op een programma dat u wilt schrijven? In BASIC zal het altijd duidelijk zijn voor de gebruiker, maar als u een machinetaalprogramma gebruikt in de betreffende 16K RAM is de timing van de routines niet constant. Gewoonlijk is dit niet van belang, echter routines met een nauwkeurig bepaalde tijd zoals de BEEP-routine, zullen niet goed werken. De BEEP in BASIC werkt wel goed omdat de machinecode hiervoor in de ROM staat.

8.4 Iets over onderbrekingen

Als u begint met het schrijven van machinetaalroutines, wilt u wellicht vectoronderbrekingen gebruiken. Als de CPU onderbroken wordt (bijvoorbeeld door externe hardware), kunt u hem een routine laten uitvoeren waarvan het adres gevonden kan worden op een geheugenplaats die aangegeven wordt door een 16-bit aanwijzer. Deze 16-bit aanwijzer wordt gemaakt door het I-register van de CPU en een 8-bit byte die door het onderbrekende apparaat geleverd wordt. Het I-register definieert in dit geval A8 t/m A15. Nemen we aan dat het adres van uw onderbreekroutine zich bevindt in de 16K RAM die door de ULA gebruikt wordt. Het I-register bevat dus een decimaal getal tussen 64 en 127 ($A15 = 0, A14 = 1$). Er zal nu gedurende elke instructiecyclus van de CPU een opfrissing plaatsvinden. De opfrissing zet de inhoud van het I-register op A8 t/m A15 en geeft een \overline{MREQ} . Deze combinatie van signalen brengt de ULA in verwarring, die verwacht het lezen uit of het schrijven in het videogeheugen door de CPU. Dat gebeurt niet, dus raakt de ULA in verwarring. Zo erg, dat hij de video-uitvoer vergeet, hierdoor ontstaat sneeuw op het scherm!

Probeer het volgende BASIC-programma eens, hiermee zet u het I-register en u ziet de sneeuw ontstaan. Denk eraan dat de waarden tussen 64 en 127 in het I-register sneeuw veroorzaken.

```
10 CLEAR 32499
```

```
20 INPUT 'Geef de waarde voor het I-register'; v
```

```

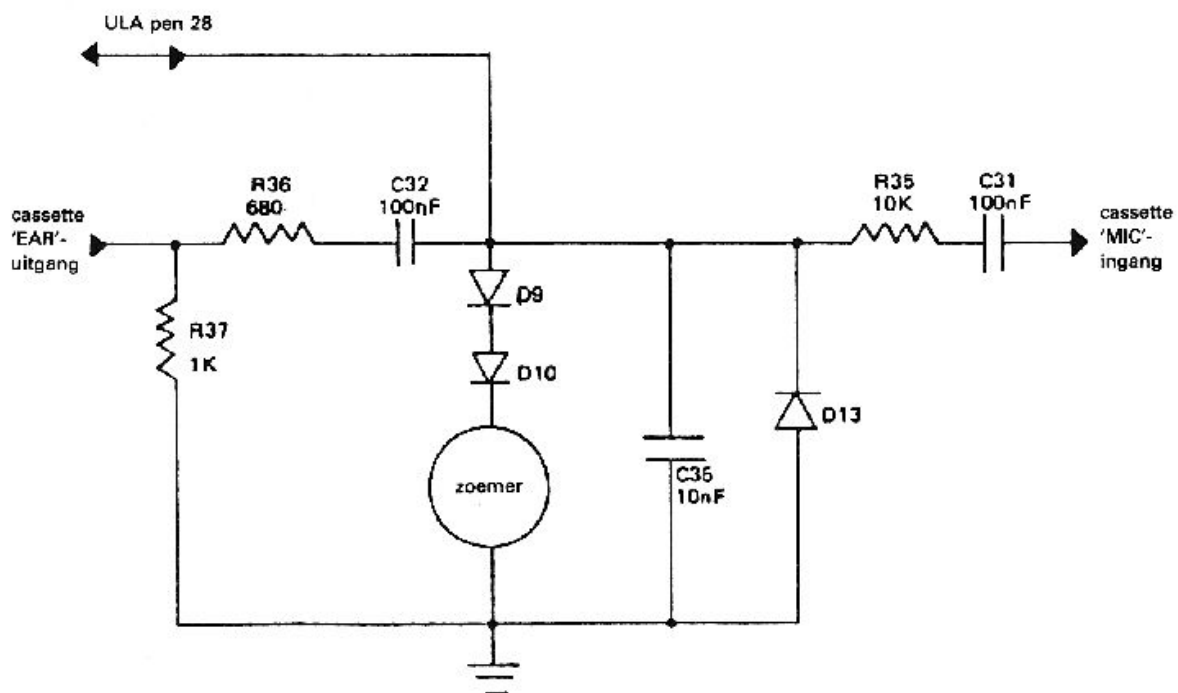
30 POKE 32500,62: REM LD A,v
40 POKE 32501,v
50 POKE 32502,237: REM LD I,A
60 POKE 32503,71
70 POKE 32504,201: REM RET
80 LET a=USR 32500
90 GO TO 20

```

Regel 10 verhindert dat BASIC het geheugen boven adres 32499 gebruikt. Regels 30 t/m 70 zetten een machinecoderoutine in het geheugen. U wordt om een waarde voor het I-register gevraagd. Het machinetaalprogramma zet zelf het I-register als het vanuit BASIC wordt aangeroepen in regel 80. De terugkeer uit de machinecode springt naar regel 90, waardoor het gehele BASIC-programma wordt herhaald.

8.5 Klokken

De ULA produceert zijn eigen moederklok. De frequentie van deze moederklok wordt door het kristal X1 constant op 14 MHz gehouden. De 14 MHz moederklokfrequentie wordt door 2 gedeeld om de juiste videostipfrequentie van 7 MHz te krijgen. Nogmaals delen door 2 geeft de frequentie van 3,5 MHz die naar de Z80A gaat. Deze 3,5 MHz klok is niet constant en kan gedurende korte perioden gestopt worden door de ULA om de CPU terzijde te schuiven bij de toegang tot het video-geheugen.



Afb. 8.5 Schema van de cassette-aansluiting en zoemer

8.6 Toetsenbord- en cassette-invoer

Het toetsenbord is in hoofdstuk 7 besproken. Samengevat, als u een byte inleest van I/U-adres 245, bevatten D0 t/m D4 het logische niveau van de KB13 t/m KB9-ingangen naar de ULA. Deze bits worden op 0 gezet als er een toets is ingedrukt en zijn 1 als dat niet het geval is. D6 heeft het ingangsniveau van de EAR-ingang van de cassetterecorder.

8.7 Zoemer-, cassette- en randkleuruitvoer

De bits D2, D1 en D0 in een uitvoerbyte naar poort 254 bepalen de randkleur. Allemaal logisch 1 geeft wit en allemaal logisch 0 geeft zwart.

Hoewel de cassette- en zoemerin- en uitvoer verbonden zijn met de ULA-pen 28 (zie afbeelding 8.5), werken zij onafhankelijk. Bit D3 van de uitvoer naar poort 254 stuurt de MIC-aansluiting, bit D4 stuurt de zoemer. De tabel legt uit hoe:

D4 (zoemer)	D3 (MIC)	spanning op ULA-pen 28
0	0	+0,75 volt
0	1	+1,3 volt
1	0	+3,3 volt

Het aansturen van de cassette met 1,3 volt levert onvoldoende spanning om de 1,4 volt spanningsval over de dioden D9 en D10 te compenseren. De zoemer werkt niet bij het laden of wegschrijven van programma's. Echter, wanneer u de cassette ingeschakeld laat en de zoemer met 3,3 volt laat werken, zal dit signaal worden opgenomen. Omgekeerd, door de MIC (hoge impedantie) of de EAR (lage impedantie) te verbinden met een audioversterker kunt u als u dat wilt een luide BEEP-toon produceren.

9. De geheugenuitbreiding

De Spectrums model 2 en model 3 hebben IC-voetjes voor de extra geheugen-chips, zodat die direct in de printplaat gestoken kunnen worden. Als u goedkope 64K chips kunt krijgen kunt u die gebruiken in plaats van de 32K chips van Sinclair.

Het schema van de 32K geheugenuitbreiding staat in afbeelding 9.2. Het werkt bijna net zo als het 16K video-RAM beschreven in hoofdstuk 5. Het grootste verschil is dat de rij-adrespuls (RAS)- en de kolomadrespuls (CAS)-signalen gemaakt worden door logische poorten te selecteren en niet door de ULA. Het circuit werkt op één van de drie manieren:

1. Uit het geheugen lezen

Als de CPU gegevens uit dit 32K geheugenblok wil lezen, zal A15 logisch 1 zijn. RD en $\overline{\text{MREQ}}$ worden beiden door de Z80A omlaag getrokken. De hoog-laag overgang van $\overline{\text{MREQ}}$ maakt RAS laag om A0 t/m A7 aan de acht dynamische RAM-chips te koppelen. De adresmultiplexers sturen A0 t/m A7 altijd naar de geheugenchips als $\overline{\text{MREQ}}$ hoog is. Het RC-netwerk, bestaande uit R2 en C2, vertraagt de $\overline{\text{MREQ}}$ -overgang naar de adresmultiplexers. Daardoor krijgen de geheugenchips de gelegenheid A0 t/m A7 te koppelen voordat de multiplexers A8 t/m A14 naar de geheugenchips koppelen. Het RC-netwerk bestaande uit R1 en C1 vertraagt daarna het signaal opnieuw voordat de $\overline{\text{CAS}}$ in werking komt. Dit geeft de adresmultiplexers de gelegenheid de adresgegevens om te schakelen. Na de $\overline{\text{CAS}}$ -bewerking is het volledige 15-bit adres aan de dynamische geheugenchips gekoppeld. Na een korte vertraging verschijnen de gewenste gegevens op de databus van de Spectrum waardoor ze door de CPU gelezen kunnen worden.

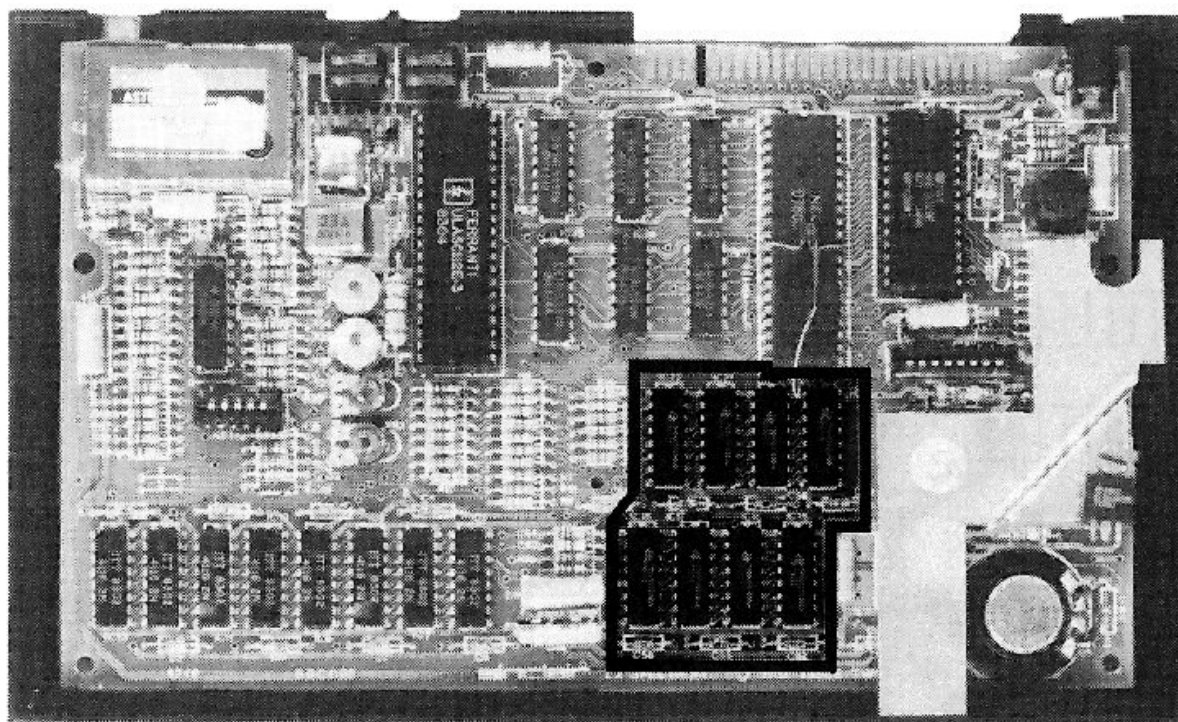
2. In het geheugen schrijven

Als de CPU gegevens in deze 32K extra geheugen op wil slaan, wordt A15 wederom logisch 1 gemaakt en worden WR en $\overline{\text{MREQ}}$ laag. Het verschil tussen schrijven en lezen is dat de WR-pen op dynamische geheugens laag gemaakt wordt door WR maar niet door RD. De geheugenchips weten daardoor dat ze gegevens moeten opnemen van de databus in plaats van ze er op te zetten.

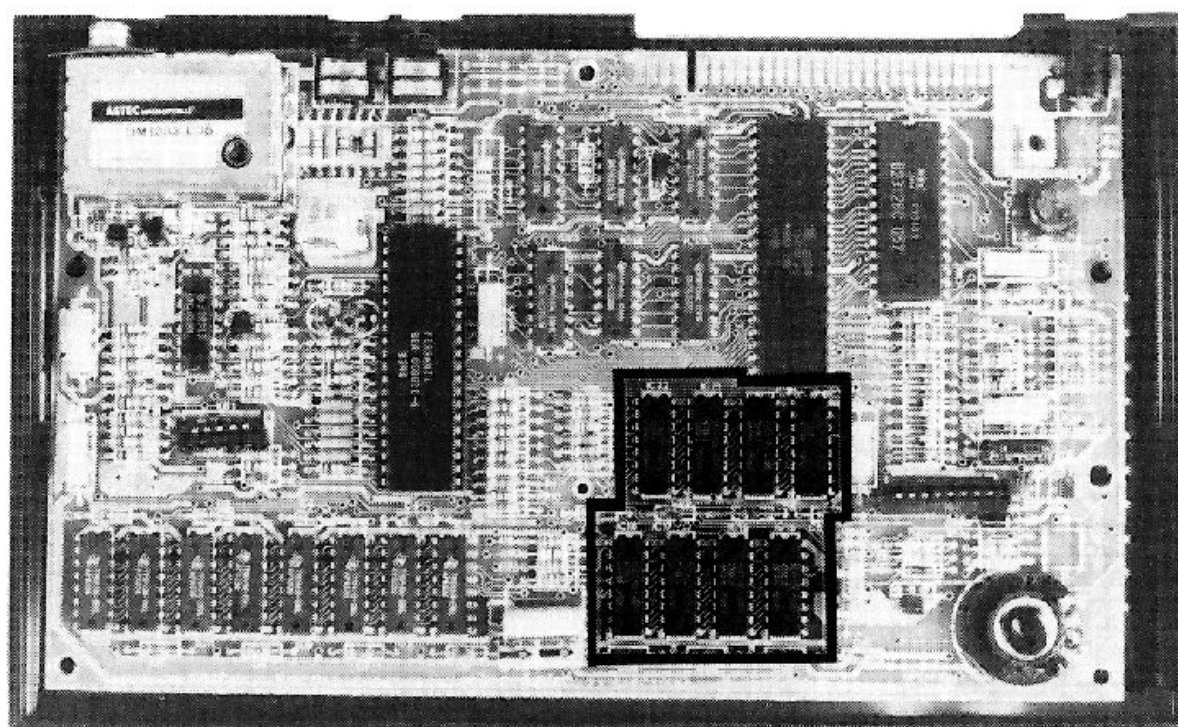
3. Geheugen opfrissen

Als er een opfriscyclus door de Z80A wordt uitgevoerd, blijven WR en RD inactief in de logische 1-toestand. De adresmultiplexers zetten daarom A0 t/m A7

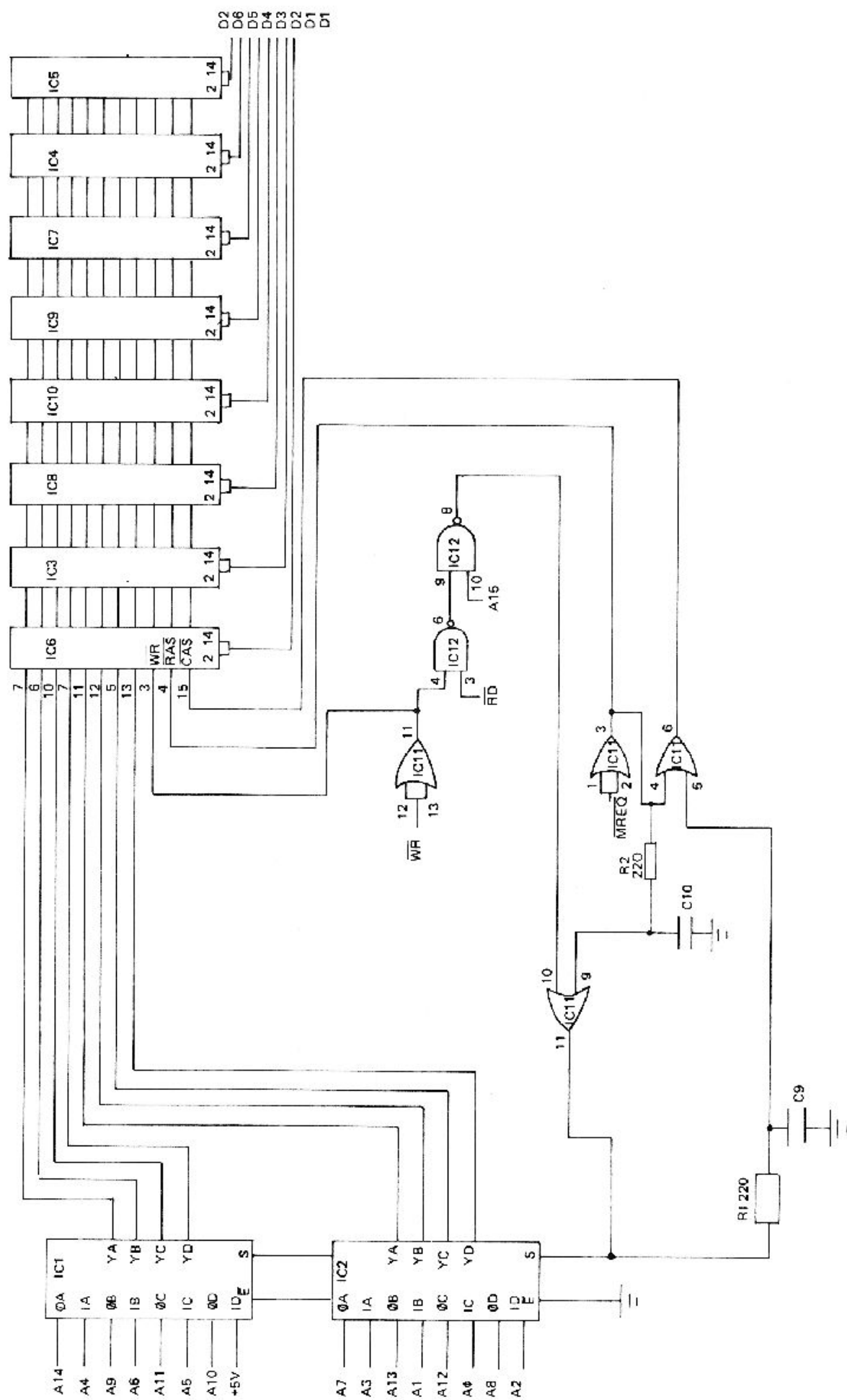
Model 2



Model 3



Afb. 9.1 Foto van het Spectrum-bord met de geheugenuitbreiding aangegeven



Afb. 9.2 Schema van de 32K geheugenuitbreiding

direct op de geheugenchips. De Z80A zet het getal van de rij die opgefrist moet worden op A0 t/m A6 (7 bits geven 128 verschillende rij-adressen). \overline{MREQ} wordt dan laag om de \overline{RAS} -lijnen van de geheugens te bewerken. Dit is alles wat nodig is voor een opfrissing. Elk van de 128 rijen wordt op deze manier tenminste elke twee milliseconden opgefrist. Voortdurende opfrissing van het videogeheugen is niet nodig met uitzondering tijdens de beeldterugslagtijd. Dit komt omdat de ULA achtereenvolgens elk van de 128 rijen van het videogeheugen uitleest voor de videogegevens.

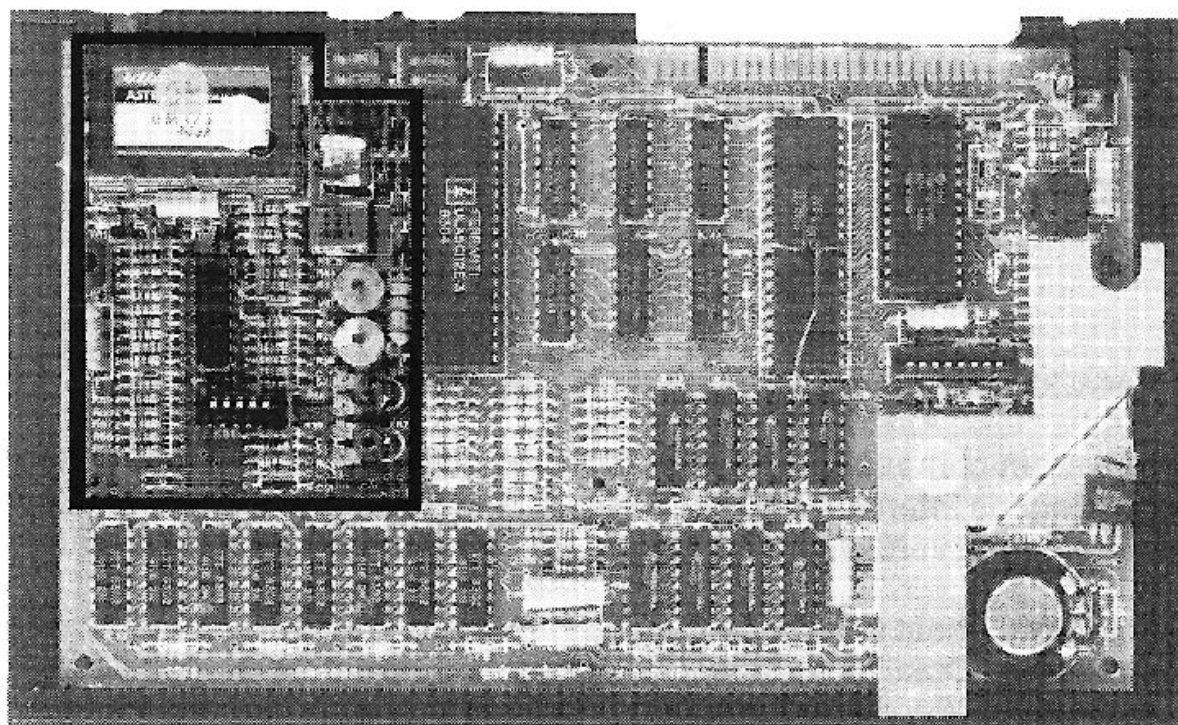
10. Het videocircuit

Het Spectrum-videocircuit is opgebouwd rondom de geïntegreerde schakeling LM1889N van National Semiconductor. Het volledige schema staat in afbeelding 11.1. Dit IC accepteert twee kleurverschilsignalen (U = blauw-geel, V = rood-geel) uit de ULA, en geeft een enkelvoudig kleuruitvoersignaal. Het is economischer om kleurverschilsignalen te gebruiken dan de aparte rood-, groen- en blauwsignalen. Slechts twee circuits zijn nodig in plaats van de drie voor RGB (rood, groen, blauw). De kleur wordt gemengd met het binnenkomende samengestelde videosynchronisatie- en helderheidssignaal (Y van de ULA) om het samengestelde kleurenvideosignaal te geven. Nadat het gebufferd is door een emittervolgerschakeling die TR2 gebruikt, wordt het signaal aan een videomodulator toegevoerd. Hierdoor kan de video-informatie op een gewoon televisietoestel zichtbaar gemaakt worden. Het is jammer dat er geen gebruik gemaakt is van RGB-video omdat enkele monitoren om deze invoer vragen. Deze matige kwaliteit van het beeld dat met een televisie verkregen wordt kan verbeterd worden door een kleurenmonitor te gebruiken die een samengestelde video-ingang heeft.

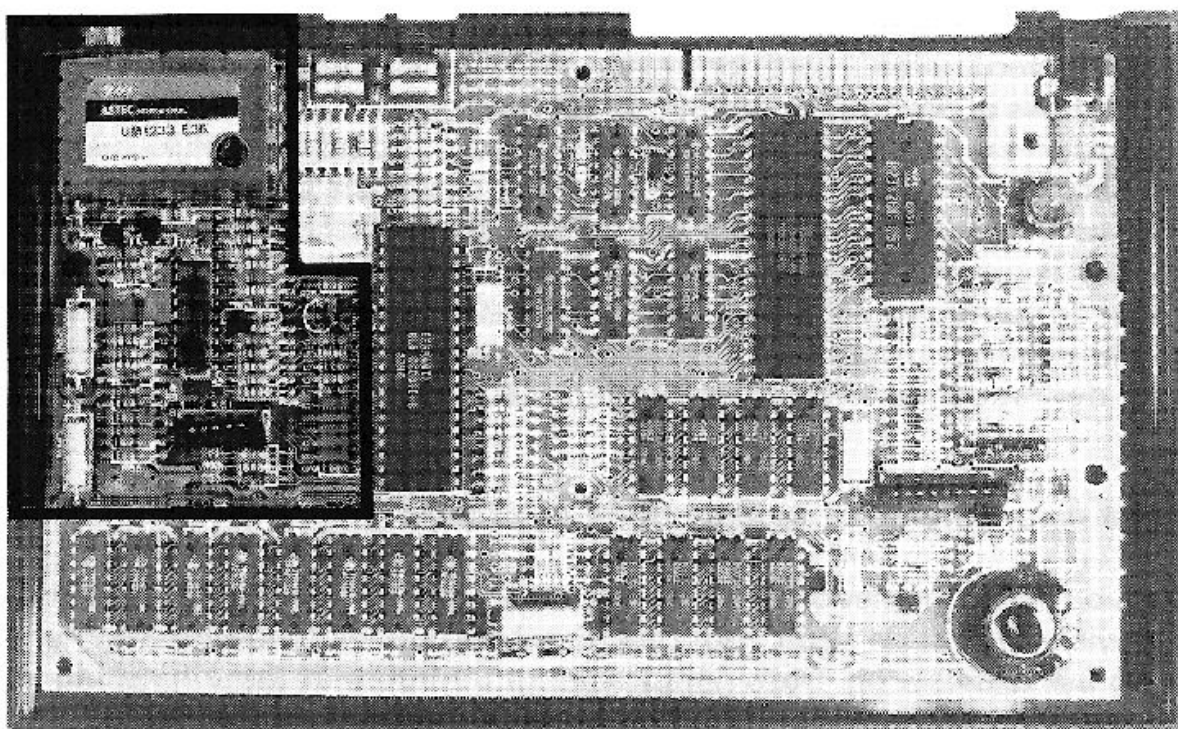
De twee instelpotentiometers VR1 en VR2 in model 2 veranderen de relatieve versterking van het rood-geel- en het blauw-geel-signaal. Door deze instelling te veranderen kunt u de kwaliteit van de kleur veranderen of van de grijsschaal. Het volgende hoofdstuk geeft aan hoe dit te doen.

Bij model 3 van de Spectrum ontbreekt deze instelmogelijkheid.

Model 2



Model 3



Afb. 10.1 Foto van het Spectrum-bord met het videocircuit aangegeven

11. Regel het videocircuit af voor een betere beeldkwaliteit

11.1 Verwijdering van de bewegende streepjes naast de letters

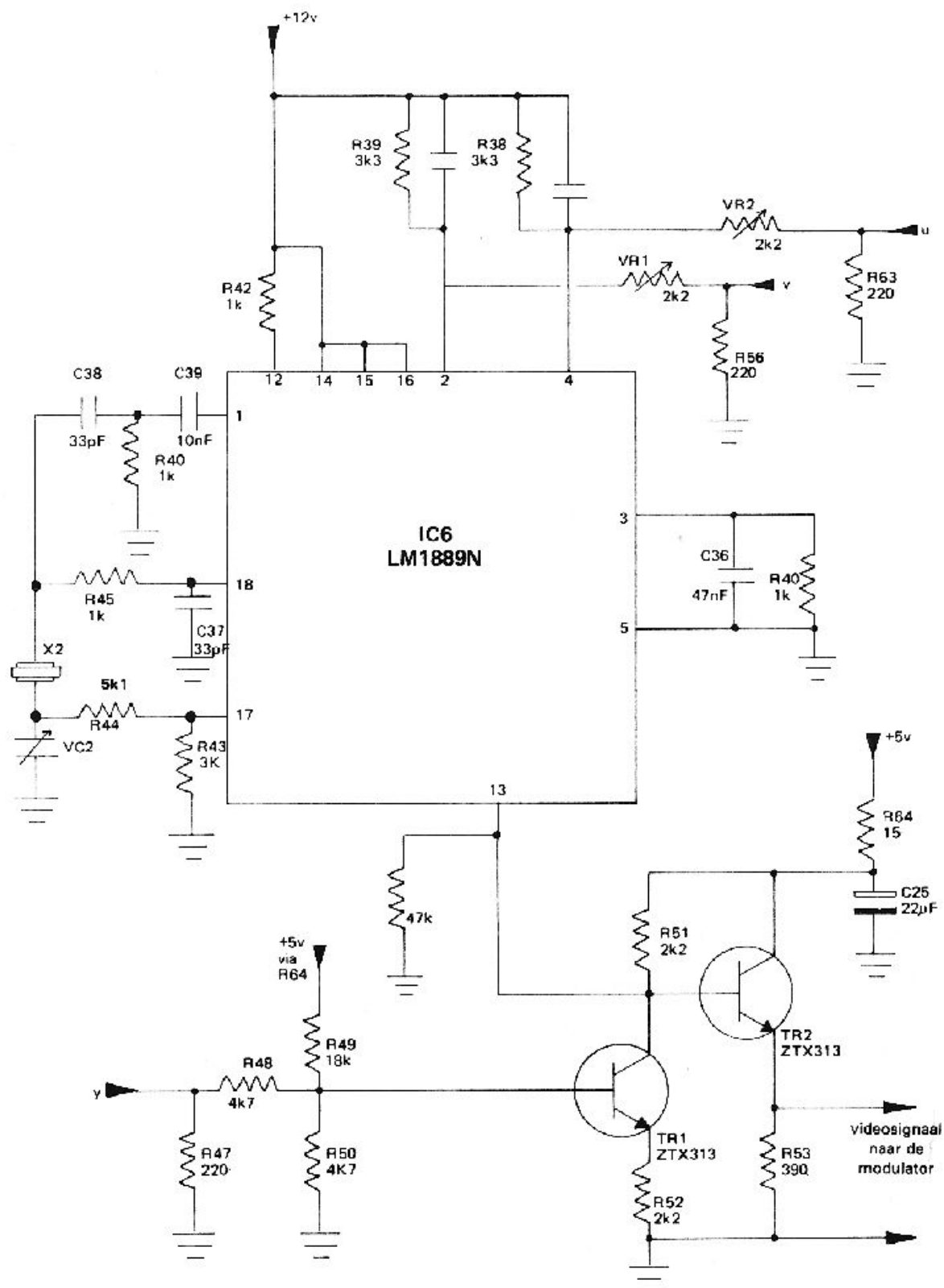
Veel Spectrum-computers hebben van die vervelende streepjes naast de letters. Dit komt door de interactie van de 14 MHz klok van de ULA met het TV-beeld. De streepjes kunnen verdwijnen als we de klokfrequentie iets wijzigen met behulp van VC1.

Om dit te doen, moet u het scherm vullen met tekst, bijvoorbeeld een oude programma-listing. Draai nu de computer om zodat hij op het toetsenbord rust. Als u een ouder model Spectrum heeft ziet u een klein gaatje aan de rechterkant waardoor u een schroefkop kunt zien. Als uw Spectrum dit gaatje niet meer heeft zult u hem open moeten maken.

Hiervoor moet u aan de onderkant 5 schroeven verwijderen en het toetsenbord zover mogelijk optillen. Zoek nu de kleine schroef in het midden van de printplaat en verwijder die. De printplaat is nu los van de onderzijde van de Spectrum. Draai de printplaat om. U ziet nu een klein gat met daarachter een schroefkop aan de rechterkant van de printplaat. Met een kleine schroevendraaier kunt u de schroef voorzichtig verdraaien. Hierdoor merkt u het effect op het beeld. Nu moet u zoveel naar links of naar rechts draaien totdat het beeld de beste kwaliteit heeft. Als u de schroef een volle slag ronddraait heeft u weer de oorspronkelijke instelling. Soms helpt het als u de kanaalinstelling van de televisie bijregelt. Helaas, wanneer de computer warm wordt zet ook het kristal uit en zal de frequentie veranderen. De streepjes naast de letters kunnen daarom terugkomen als de temperatuur van de computer verandert. Daarom moet u hem eerst warm laten worden voor u hem bij gaat regelen. Als de streepjes weer erger worden is de enige oplossing de frequentie weer bij te regelen met behulp van VC1.

11.2 Het veranderen van de kleur of de grijsschaal

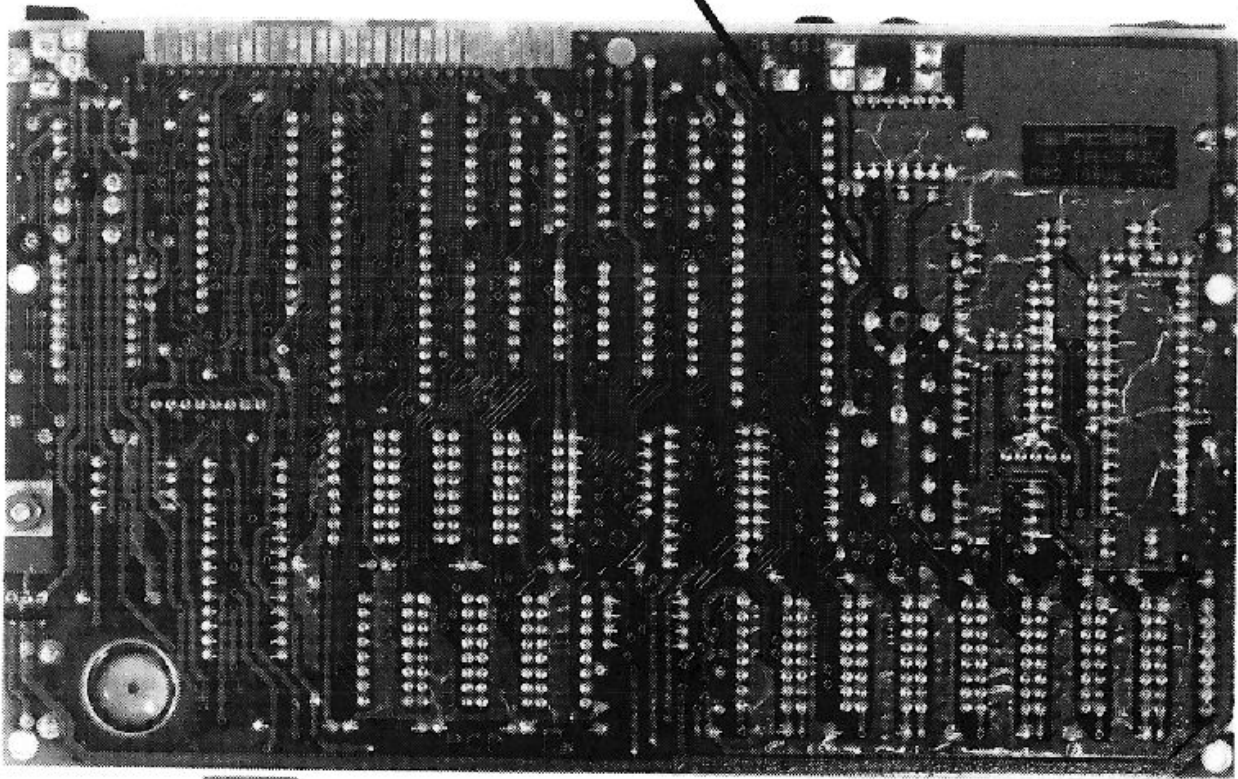
De eenvoudigste manier om de kleur te veranderen is met de kleurenregelaar op uw televisie. Hetzelfde geldt voor de grijsschaal, stel de helderheid en het contrast op uw televisie bij. Deze paragraaf legt uit hoe u de kleurinstelling in uw Spectrum kunt veranderen. Dit geldt ALLEEN voor het VERANDEREN van de kleur (of de grijsschaal). Als er helemaal geen kleur is lees dan hoofdstuk 13. Om de kleurinstelling of de grijsschaal van uw computer bij te regelen moet u uw Spectrum open maken door de 5 schroeven aan de onderzijde te verwijderen.



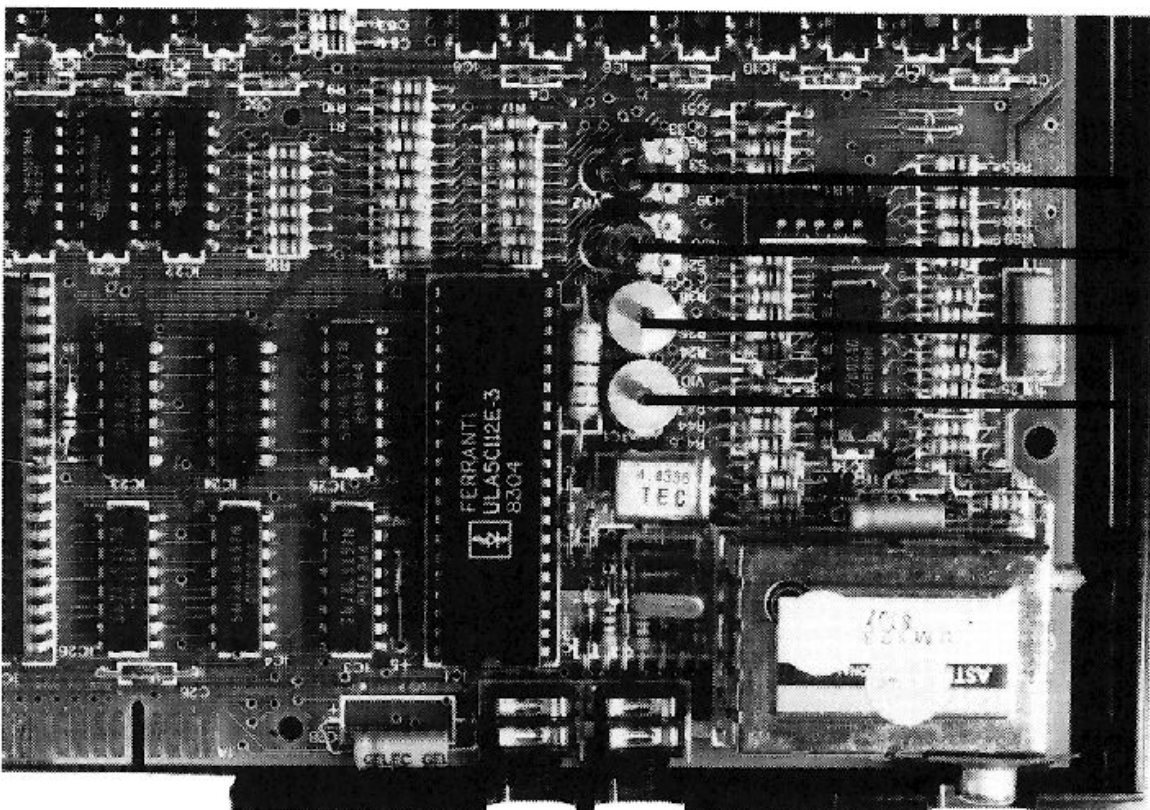
Afb. 11.1 Schema van het videocircuit

schroefgat voor VC1

Model 2



Afb. 11.2 Foto van de schroefkop van VC1



VC1

VC2

VR1

VR2

Afb. 11.3 Videocircuit met de afregelplaatsen

Nadat u de 5 schroeven verwijderd heeft moet u een schermbeeld maken met alle kleuren en grijstinten. Het programma op pagina 81 van uw BASIC-handboek is hiervoor geschikt. Nu kunt u voorzichtig VR1 en VR2 met een schroevendraaier bijregelen. De plaats van deze twee instelpotentiometers staat in appendix D en in afbeelding 11.3. Bekijk de kleuren op het scherm wanneer u de instelling verandert. Dit geeft u een indruk van de verandering. VR1 verandert de rood-geel-amplitude en VR2 verandert de blauw-geel-amplitude. VR1 moet samen met VR2 verdraaid worden om het groen-niveau te wijzigen. Hoe u tenslotte de kleuren (of de grijsschaal) instelt is een kwestie van persoonlijke smaak.

12. De randconnector

Dit hoofdstuk bevat de beschrijving van de signalen die op de contacten van de randconnector achterop uw Spectrum beschikbaar zijn. Enkele tips en ideeën om uw eigen schakelingen te maken zijn ook opgenomen. Voor het uitvoeren van enkele eenvoudige experimenten zie hoofdstuk 14.

De aansluitingen gemerkt met 28A en 1A zijn aangegeven aan de bovenkant van de Spectrum-print. 28B is onder 28A en 1B is onder 1A. Afbeelding 12.1 geeft het achteraanzicht van de Spectrum weer met de juiste stand van de randconnector. Denk erom dat hij in het BASIC-handboek ondersteboven is afgebeeld.

12.1 De aansluitingen aan kant A

1A -A15 van de adresbus. Deze kan gebruikt worden als uitvoer uit de Spectrum om verschillende externe apparaten te selecteren in samenhang met de rest van de adresbus. Hij kan ook gebruikt worden als invoer voor de Spectrum vanuit elk extern apparaat dat de controle van de CPU-bussen overneemt door het BUSREQ-signaal te gebruiken.

2A -A13 van de adresbus.

3A -D7 van de databus. Deze tweerichtings 8-bit bus kan gebruikt worden om gegevens naar of van de Spectrum te transporteren.

4A -niet aangesloten.

5A -slot. Insteekpunt voor het juist positioneren van de randconnector.

6A -D0 van de databus.

7A -D1 van de databus.

8A -D2 van de databus.

9A -D6 van de databus.

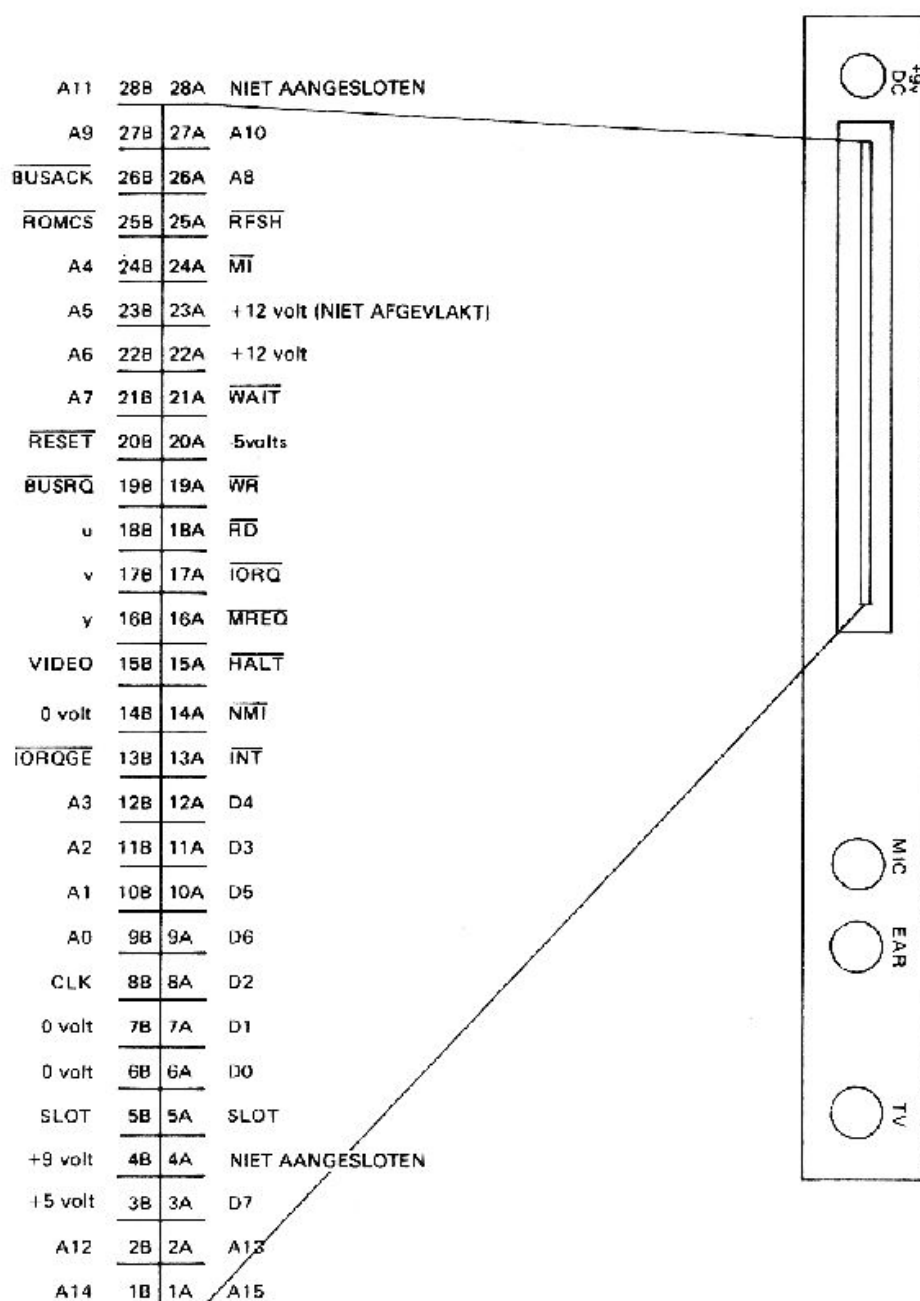
10A -D5 van de databus.

11A -D3 van de databus.

12A -D4 van de databus.

13A - $\overline{\text{INT}}$, Z80A-onderbreeklijn en via R26 (680 ohm) met de ULA- $\overline{\text{INT}}$ -pen. Deze kan gebruikt worden door een extern apparaat om een onderbreking van de Spectrum aan te vragen, of als hij met de +5 volt verbonden is verhindert hij de ULA de Z80A-CPU elke 20ms te onderbreken. In hoofdstuk 14 staat een experiment dat de $\overline{\text{INT}}$ gebruikt.

14A - $\overline{\text{NMI}}$, de Z80A niet-maskeerbare onderbreking. Normaal op logisch 1 gehouden door R28 (10K ohm) verbonden met de +5 volt. Als hij door een extern



Afb. 12.1 De randconnectoraansluitingen

apparaat laag wordt gemaakt dan wordt de Z80A gedwongen om naar adres 102 decimaal (66 hex) te springen en de machinecode die daar staat uit te voeren. In hoofdstuk 14 staat een experiment dat de $\overline{\text{NMI}}$ gebruikt.

15A $\overline{\text{HALT}}$, de CPU- $\overline{\text{HALT}}$ -uitvoer. Geeft aan dat de CPU een software HALT-instructie uitvoert. De CPU wacht op een onderbreking vanuit een extern apparaat voor hij doorgaat met de uitvoering van het programma. Een eenvoudig experiment met de HALT staat in hoofdstuk 14.

16A $\overline{\text{MREQ}}$, de CPU-geheugenaanvraaguitvoer. Geeft aan dat de adresbus nu een geldig adres bevat voor het schrijven in of het lezen uit het geheugen. Dit signaal wordt ook gebruikt bij een opfrissing van het geheugen.

17A - $\overline{\text{IOREQ}}$, de invoer/uitvoer-aanvraag-uitvoer van de Z80A. Hij geeft aan dat de laagste helft van de adresbus een geldig I/U-adres bevat voor een I/U-bewerking. De bovenste helft van de adresbus bevat de inhoud van het A-register van de CPU als er een IN A,n of een OUT n,A gebruikt was in een machinetaalprogramma, n komt op de laagste helft van de adresbus. Als er een indirecte register-I/U-bewerking wordt uitgevoerd, komt het CPU-C-register op A0 t/m A7 en het CPU-B-register op A8 t/m A15. Vanuit BASIC kan een volledig 16-bit I/U-adres opgegeven worden. Dit komt op A0 t/m A15 als $\overline{\text{IORQ}}$ actief is.

18A - $\overline{\text{RD}}$, de Z80A-leesuitvoer die aangeeft dat de CPU gegevens wil lezen vanuit het geheugen of vanaf een I/U apparaat. Het geadresseerde apparaat gebruikt dit signaal om de relevante gegevens op de databus te zetten.

19A - $\overline{\text{WR}}$, de schrijfwitvoer van de Z80A geeft aan dat de databus van de CPU gegevens bevat die opgeslagen moeten worden op de geadresseerde plaats.

20A -voedingsslijn -5 volt. Bekijk het voedingsschema voor de details.

21A - $\overline{\text{WAIT}}$, de Z80A-wachtinvoer, wordt via R29 (1K5 ohm) hoog gehouden op +5 volt. Deze invoer kan door langzame externe apparaten gebruikt worden om de CPU te laten wachten tot ze klaar zijn om gegevens over te geven. U moet er voor zorgen dat deze lijn niet te lang (meer dan 1 ms) achter elkaar gebruikt wordt want het dynamische geheugen wordt gedurende de wachtstatus NIET opgefrist. Het programma of de gegevens kunnen dus als de $\overline{\text{WAIT}}$ te lang is gebruikt worden beschadigd of kunnen verloren gaan.

22A -voedingsspanning +12 volt (zie het voedingsschema voor meer details).

23A - +12 volt NIET de -12 volt zoals in het BASIC-handboek staat aangegeven! Deze pin is in feite verbonden met de collector van TR4 en geeft daarom een ongestabiliseerde +12 volt. U moet deze niet als een directe +12 volt voeding gebruiken. Gewoonlijk moet de +12 volt voeding van 22A gebruikt worden. Een schema om deze spanning naar -12 volt om te zetten is in hoofdstuk 3 gegeven.

24A - $\overline{\text{M1}}$, de Z80A-machinecyclusuitvoer. Deze geeft aan dat de CPU nu de Op-code voor de volgende uit te voeren opdracht uit het geheugen haalt.

25A - $\overline{\text{RFSH}}$, geheugenopfrissignaal van de Z80A. Gedurende een geheugen opfrissing komt het R-register van de CPU op A0 t/m A7 en het I-register van de CPU komt op A8 t/m A15. Dit kan grappige gevolgen hebben als het I-register een waarde tussen 64 en 127 bevat. Verdere details staan in hoofdstuk 8.

26A -A8 van de adresbus.

27A -A10 van de adresbus.

28A -niet aangesloten.

12.2 De aansluitingen aan kant B

1B -A14 van de adresbus.

2B -A12 van de adresbus.

3B - +5 volt voeding van de logische chips.

4B - +9 volt ongeregelde gelijkspanning uit de nettransformator.

5B -slot. Insteekpunt voor het juist positioneren van de randconnector.

6B -nul volt aansluiting voor de voeding. Er zijn twee aansluitingen omdat deze lijn de som van de anderen transporteert.

7B -nul volt voeding. U moet beide verbindingen aansluiten bij externe schakelingen om te voorkomen dat een aansluiting overbelast wordt.

8B -CLK, het 3,5 MHz kloksignaal van de ULA. Kan gebruikt worden om de werking van verschillende Z80A-ondersteuningschips te synchroniseren met de Z80A. Denk er aan dat de ULA deze klok kan stoppen als de CPU de eerste 16K RAM wil gebruiken. Meer details staan in hoofdstuk 8.

9B -A0 van de adresbus.

10B -A1 van de adresbus.

11B -A2 van de adresbus.

12B -A3 van de adresbus.

13B - $\overline{\text{IORQGE}}$, de $\overline{\text{IORQ}}$ -invoer van de ULA. Hij is ook verbonden met de $\overline{\text{IORQ}}$ -uitvoer van de Z80A via R27 (680 ohm). Als $\overline{\text{IORQGE}}$ met de +5 volt verbonden wordt ontvangt de ULA zijn $\overline{\text{IORQ}}$ -signaal van de Z80A niet. Dit kan makkelijk zijn om het aantal apparaten uit te breiden voor gebruikers-I/U, A7 kan gebruikt worden om het $\overline{\text{IORQGE}}$ -signaal uit te schakelen als hij laag is. U kunt dan A0 t/m A6 in elke combinatie gebruiken om één van de 128 I/U-apparaten te adresseren. Een schema hiervoor te samen met een experiment waarin de $\overline{\text{IORQGE}}$ wordt gebruikt staat in hoofdstuk 14.

14B -nul volgt nog een nul volt aansluiting, bedoeld om te gebruiken te samen met videosignalen.

15B -VIDEO. Videosignalen uit de Spectrum. Ze zijn niet rechtstreeks verbonden met de aangegeven signalen. U moet dit zelf doen als u deze

16B -Y signalen wilt gebruiken. VIDEO kan gebruikt worden om een mo-

17B -V nitor te sturen.

18B -U

19B - $\overline{\text{BUSRQ}}$, de Z80A-busaanvraaginvoer en door R30 (1K ohm) hoog gehouden op +5 volt. Externe apparaten kunnen hiervan gebruik maken om het gebruik van alle CPU-bussen aan te vragen. De controle wordt, na het afhandelen van de lopende machinecyclus overgegeven. De Z80A geeft het beschikbaar zijn van de bussen aan bij externe apparaat aan door $\overline{\text{BUSACK}}$ laag te maken.

20B - $\overline{\text{RESET}}$, de Z80A-herstelpen. Deze lijn is, bij het inschakelen, van een eenvoudige RC-vertraging voorzien, om de computer in staat te stellen een werkbare staat te verkrijgen voordat de CPU iets gaat doen. Een eenvoudige herstelknop kan tussen deze pen en de 0 volt gezet worden (zie afbeelding 14.2). Als deze schakelaar gebruikt wordt zal de CPU zich herstellen als bij het inschakelen. Het verschil is dat u de voedingskabel niet hoeft los te maken. Een HERSTEL-schakelaar is erg handig voor iemand die de BASIC-ROM vervangt door een machinecodemonitor, omdat het geheugen dan niet gewist wordt wanneer er een herstelactie wordt uitgevoerd.

21B -A7 van de adresbus.

22B -A6 van de adresbus.

23B -A5 van de adresbus.

24B -A4 van de adresbus.

25B - $\overline{\text{ROMCS}}$, de ROM-chipkeuzepen en via R33 (680 ohm) met de ULA. Als u de $\overline{\text{ROMCS}}$ met +5 volt verbindt zal het 16K BASIC-ROM uit het Spectrum-geheugen verdwijnen. Het is duidelijk dat u het moet vervangen door een ander programma in één of ander extern geheugen bij het omschakelen, anders zal de computer vastlopen!

26B $\overline{\text{BUSACK}}$, het Z80A-busbevestigingssignaal vertelt een extern apparaat dat het nu de volledige controle over de Z80A-bussen heeft. Het wordt samen met het $\overline{\text{BUSRQ}}$ -signaal gebruikt.

27B -A9 van de adresbus.

28B -A11 van de adresbus.

13. Het opsporen van fouten

De Spectrum is een gecompliceerd apparaat en er kunnen verschillende fouten in voorkomen. De meeste fouten zullen een eenvoudige oorzaak hebben zoals een losse verbinding, een slechte afstemming of een doorgebrande zekering. Het zijn deze eenvoudige fouten die we in dit hoofdstuk willen opsporen. Bij ingewikkelde fouten, moet de computer terug naar de importeur of naar een goede reparateur.

Verschijnsel: Geen beeld.

Ga in deze volgorde te werk:

- a. Verbind het videosnoer van de Spectrum met de televisie.
- b. Zet de televisie aan.
- c. Verbind de voeding van de Spectrum met het lichtnet.
- d. Verbind het voedingssnoer met de Spectrum.
- e. Druk op de 'ENTER'-toets van de Spectrum en houd hem ingedrukt. Hoort u een zwak geklik uit de zoemer komen? Zo niet ga door met 'g'.
- f. Als u tot hier bent gekomen werkt het grootste deel van uw Spectrum. Probeer de afstemming van uw TV op uw Spectrum af te regelen. Als u nog geen beeld heeft ga dan door met 'i'.
- g. Controleer of het voedingssnoer goed in de Spectrum zit.
- h. Als u over een multimeter beschikt, controleer dan of er +9 volt aanwezig is tussen de 0 volt en de +9 volt aansluiting achterop de Spectrum. Is er geen enkele spanning controleer dan grondig alle spanningsaansluitingen. Meestal is er ergens een onderbreking in de verbinding.
- i. Als uw Spectrum nog steeds niet werkt zoek dan professionele hulp.

Verschijnsel: Geen kleurenbeeld.

Ga in deze volgorde te werk:

- a. Verzeker u ervan dat u wel een kleurenbeeld zou moeten verwachten. Het programma op bladzijde 81 van uw BASIC-handboek geeft alle mogelijke kleuren op uw scherm.
- b. Controleer of de kleurregelaar van de televisie in de goede stand staat. Aan één kant van de schaal krijgt u alleen zwart/wit-weergave.
- c. Controleer of de televisie op het juiste kanaal is afgestemd. Als dat niet het geval is krijgt u een slechte kleurkwaliteit.

d. Als u tot hier gekomen bent is het waarschijnlijk dat het kleurenkristal in de Spectrum niet gelijk is afgestemd aan dat in uw televisie. U moet het daarom bijregelen. Dit is alleen mogelijk bij een Spectrum model 2.

Schroef de 5 schroeven onderaan de Spectrum los zodat het toetsenbord verwijderd kan worden. De afregeling gebeurt met VC2. Raadpleeg voor de componentenopstelling appendix D of afbeelding 11.3. Draai nu langzaam aan VC2 met een kleine schroevendraaier (met de klok mee of er tegenin). U moet een stand vinden waarbij de kleuren terugkomen op het scherm. Als na een volledige slag van VC2 de kleuren niet teruggekomen zijn moet de Spectrum door een vakman gerepareerd worden.

Verschijsel: Willekeurige tekens op het scherm.

Als de Spectrum wordt aangezet ontstaat er een scherm vol willekeurige tekens met allerlei kleuren. Er is geen 'copyright'-boodschap aanwezig. Wanneer dit gebeurt zonder dat er externe schakelingen aan de Spectrum zijn gekoppeld, moet uw Spectrum door een vakman worden nagekeken. Gewoonlijk wordt deze fout veroorzaakt door een slechte spanningsregeling of door een fout in een van de chips. De vermindering in de spanningsregeling die door externe schakelingen veroorzaakt wordt kan soms deze fout veroorzaken. Daarom moet u eerst de externe schakelingen loskoppelen en ze van een eigen voeding voorzien.

Lange draden naar externe schakelingen kunnen ook problemen veroorzaken. U kunt of de draden verkorten of, als dit onmogelijk is, de één of andere bufferschakeling opnemen. Wanneer het probleem zelfs optreedt zonder dat er randapparatuur aangesloten is, ligt de fout waarschijnlijk in de voeding. U kunt dit controleren door naar de Spectrum te luisteren, terwijl hij ingeschakeld is. Een zacht zoemend geluid kan worden gehoord als de voeding goed werkt.

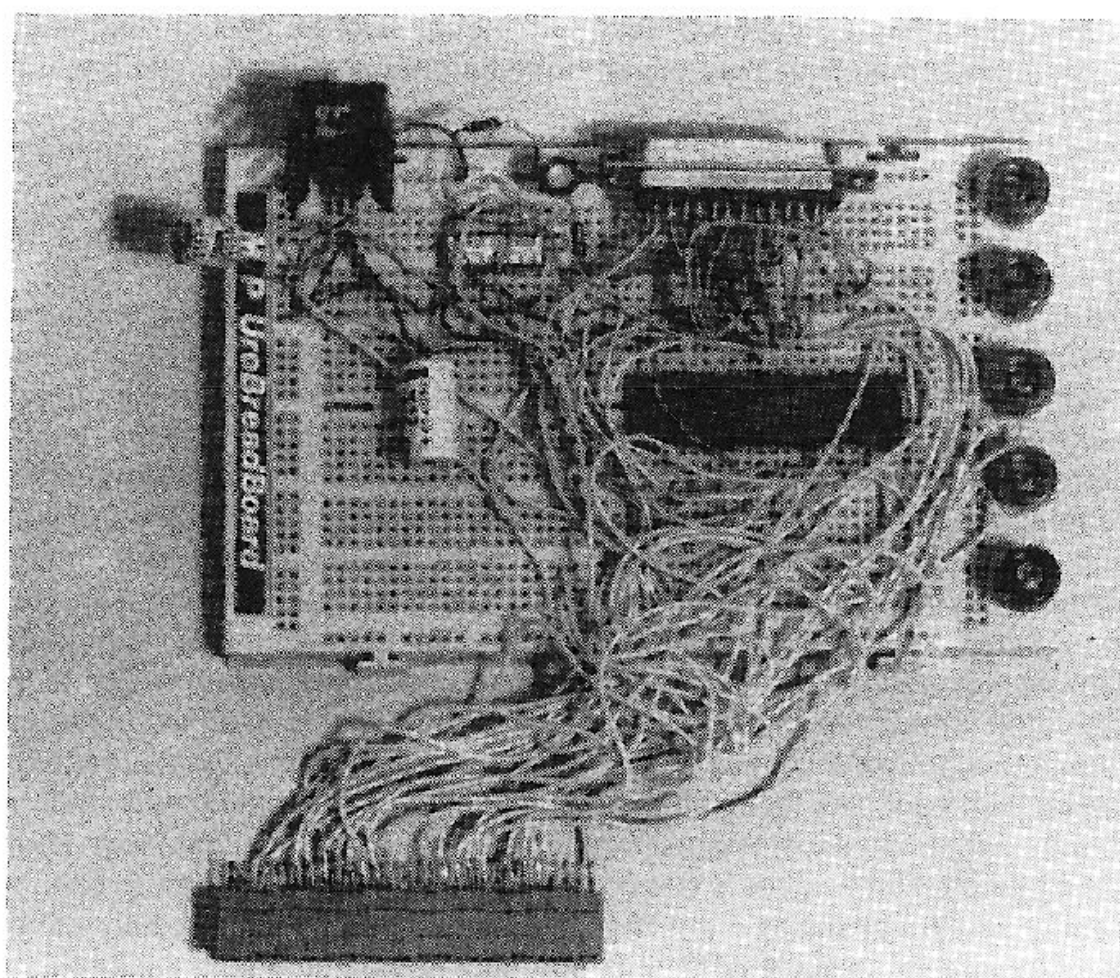
U wordt aangeraden uw Spectrum naar de importeur terug te brengen wanneer deze fout bij uw Spectrum optreedt. Als u avontuurlijk bent aangelegd en u wilt het leven van uw Spectrum in eigen hand nemen dan kunt u proberen om TR4 door een NPN-transistor te vervangen. Als dit de fout niet opheft probeer dan TR5 te vervangen door een PNP-transistor.

14. Experimenteren met de randconnector

Dit hoofdstuk geeft een heleboel praktische informatie voor hen die hun eigen schakelingen aan de Spectrum willen koppelen. U moet het in samenhang met hoofdstuk 12 lezen, dat legt elk van de randconnectorsignalen pen-voor-pen uit. Om de werking van deze signalen te demonstreren zijn er kleine schakelingen met bijbehorende testprogramma's.

14.1 Wat heeft u nodig

Als eerste heeft u een passende connector nodig om achterop uw Spectrum te



Afb. 14.1 Foto van het experimenteerbordsysteem

steken. Een connector waarbij u aan de aansluitingen verschillende draden kunt solderen is erg handig.

Het type connector is een 28-polige, tweezijdige, 0,1 inch steek randconnector met een insteekmarkering op de vijfde pool. Dit zogenaamde 'slot' dient om de connector op de juiste manier op de Spectrum te zetten. Heeft u moeilijkheden om de juiste connector te krijgen dan kunt u ook de standaard 43-polige, 0,1 inch, dubbelzijdige connector gebruiken met een slot in een van de posities van 5 tot 20. De extra contacten kunnen dan voorzichtig afgezaagd worden om de connector op de juiste lengte te brengen.

Nu dat u een goede connector heeft gaan we daar wat schakelingen aan hangen. Er zijn twee mogelijkheden:

1. Het experimenteerbord-systeem

Er zijn veel verschillende soorten experimenteerborden te koop. De meesten hebben metalen contactstrippen met ongeveer 5 contacten per strip. Deze strip-pen zijn ingegoten in een plastic basisbord. De draden naar de onderdelen zoals IC's, weerstanden, condensatoren en dergelijke kunnen door gaten in het plastic gestoken worden zodat ze contact maken met de strips daaronder. De verbindingen tussen de onderdelen worden gemaakt met stukjes geïsoleerd draad. Deze draden worden aan elke kant over ongeveer 5 mm van hun isolatie ontdaan, waarna de draad in het juiste gat van het experimenteerbord gestoken wordt.

Het maken van schakelingen op deze wijze gaat snel en eenvoudig. Als u wilt kunt u de oude schakeling vervangen door een nieuwe zonder dat het bord beschadigd wordt.

In afbeelding 14.1 is een experimenteerbord weergegeven. Dit is een opstelling voor een extra voeding en een schakeling voor 128 I/U-poorten.

2. Het verostripbord-systeem

Een verobord is een kant en klaar circuitbord met horizontale koperbanen aan één kant van het bord. Gaten zijn geboord met een steek van 0,1 inch in een vierkant patroon. Dit type bord is handig voor proefschakelingen die een blijvender karakter hebben dan die op het experimenteerbord. De aansluitingen van de onderdelen worden door de gaatjes gestoken en aan de koperbanen gesoldeerd. Om de juiste verbindingen te krijgen moeten de koperbanen vaak onderbroken worden. Hierdoor worden ongewenste verbindingen tussen de onderdelen voorkomen. U kunt gebruik maken van een speciaal stuk gereedschap, de veroppervlaktesnijder, maar het is gemakkelijker een boortje van 4,5 mm te gebruiken. Om een onderbreking te maken plaatst u het boortje in het gaatje aan de koperzijde en draait het boortje zodanig dat het koper wordt weggesneden.

Als u verobord gebruikt is het erg aan te raden voor de IC's gebruik te maken van IC-voetjes. Dan kunt u de IC's gemakkelijk verwijderen en voorkomt u oververhitting tijdens het solderen. Het grote voordeel van verobord is dat u een blijvende schakeling heeft. Het nadeel is dat wijzigingen moeilijker zijn aan te brengen en eenmaal gebruikt verobord moet meestal worden weggegooid. Een schakeling die op verobord is gemaakt ziet u in afbeelding 16.2.

Welk van de twee genoemde systemen u ook gebruikt, u moet proberen de verbindingsdraden zo kort mogelijk te houden. Als de draden aan de randconnector

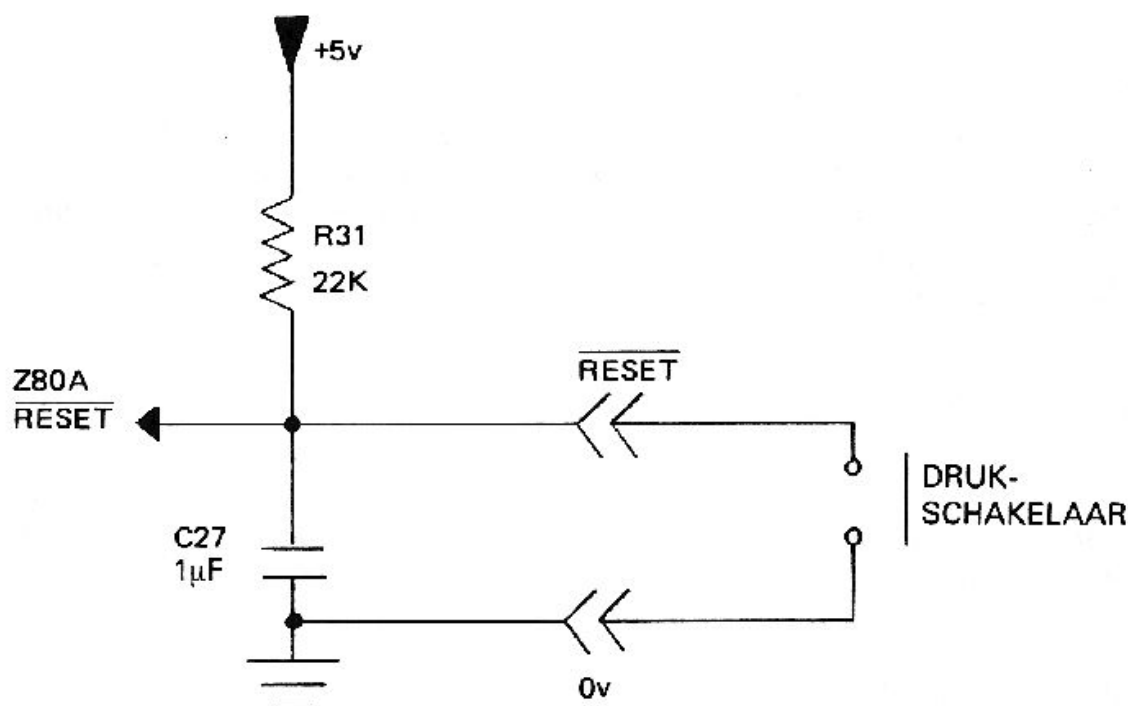
te lang zijn, kan het zijn dat de Spectrum door de extra capacitieve belasting in het geheel niet meer werkt. Lange draden kunnen er de oorzaak van zijn dat de computer, voor hij is opgewarmd, regelmatig op hol slaat.

Het bijschakelen van een paar chips aan de Spectrum, zoals de ontwerpen in dit boek, zullen geen problemen geven. De Z80A heeft zelf voldoende vermogen om de lijnen te sturen. Wanneer u veel schakelingen wilt toevoegen dan is er een vorm van buffering nodig. Als u schakelingen ontwerpt die ingewikkeld zijn dan is het, met de gegevens uit dit boek niet moeilijk om zelf een buffer te ontwerpen. Nadere aanwijzingen kunt u vinden in een van de boeken uit appendix B.

Nu zijn we toegekomen aan enkele eenvoudige schakelingen die u kunt bouwen. Verschillende bestaan alleen uit een simpele schakelaar en een weerstand.

14.2 Herstelschakelaar

De schakeling staat in afbeelding 14.2. U sluit een drukknop aan tussen de 0 volt en de **RESET** van de randconnector (zie afb. 12.1). Als u op de drukknop drukt wordt de RESET-lijn (herstellijn) naar 0 volt getrokken waardoor de Z80A hersteld wordt. Wanneer u de knop loslaat wordt C27 via R31 opgeladen tot de spanning het logische 1-niveau bereikt. De Z80A begint dan het machinecode-programma, dat op adres 0 (meestal in de Sinclair-ROM) staat, uit te voeren. Deze drukknop heeft hetzelfde effect als het inschakelen van de Spectrum en dus zullen alle programma's die in het geheugen staan verloren zijn.



Afb. 14.2 Schema van de herstelschakelaar

14.3 HALT-experiment

De $\overline{\text{HALT}}$ -lijn is een uitvoer van de Z80A-chip. Wanneer hij actief is (uitvoer = logisch 0) geef hij aan dat de CPU een software HALT-instructie uitvoert en een onderbreking verwacht van een extern apparaat. Wanneer de CPU gestopt is voert hij NOP's (No OPeration = zonder werking) uit. Dit zorgt ervoor dat het geheugenopfrissysteem blijft werken onafhankelijk van de tijdsduur van de HALT. De schakeling voor dit experiment staat in afbeelding 14.3. Een kleine LED (lichtgevende diode) is voldoende. Probeer het onderstaand programma eens.

```
10 CLEAR 32499
20 POKE 32500,118: REM HALT
30 POKE 32501,201: REM RET
40 LET a =USR 32500
50 GO TO 40
```

Regel 10 verhindert BASIC het geheugen boven adres 32499 te gebruiken. Regel 20 en 30 maken een machinecodeprogramma om een HALT uit te voeren en weer naar BASIC terug te keren (nadat de CPU een onderbreking van de ULA heeft ontvangen). Regel 40 springt naar het machinecodeprogramma. U zult de LED zien oplichten als u het programma laat lopen. In feite wordt hij ingeschakeld als het programma voor de eerste keer bij regel 40 komt. Daarna zendt de ULA een onderbreking naar de CPU die de HALT-lijn uitschakelt. Het programma gaat terug naar regel 40 en zet de HALT weer aan, tot de volgende onderbreking van de ULA wordt ontvangen. U kunt de LED niet aan en uit zien gaan want dit gebeurt met 50Hz (de ULA onderbreekt de CPU 50 maal per seconde).

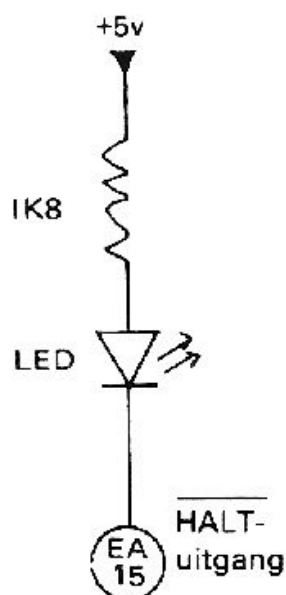
14.4 NMI-test

De schakeling voor deze test staat in afbeelding 14.4. Hiervoor is een drukknop nodig tussen de $\overline{\text{NMI}}$ -invoer op de randconnector en de 0 volt lijn. Als u op de knop drukt om de NMI (niet te maskeren onderbreking) op te wekken dan zal, ongeacht wat de CPU deed, begonnen worden met de onderbrekingsafhandelings-routine op adres 66 hex (102 decimaal). In feite zal dat het BASIC opnieuw initialiseren en dat heeft hetzelfde effect als op de herstelknop te drukken.

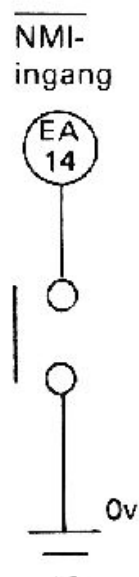
14.5 INT- en IORQGE-proeven

Bij deze proeven gebruiken we de schakelingen van afbeelding 14.5 en 14.6. Deze verbinden de juiste lijnen van de randconnector met de +5 volt. De 100 ohm weerstand dient om de stroom vanuit de voeding te beperken en kan eventueel worden weggelaten.

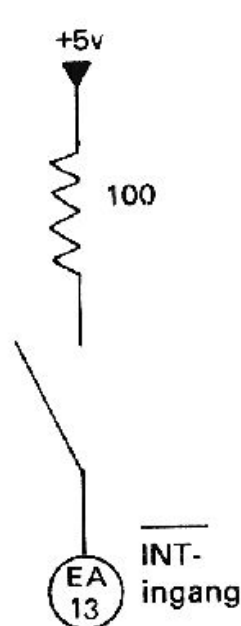
Toets het nu volgende korte programma in. Zorg ervoor dat beide schakelaars voor de $\overline{\text{INT}}$ en $\overline{\text{IORQGE}}$ open staan tijdens het intoetsen.



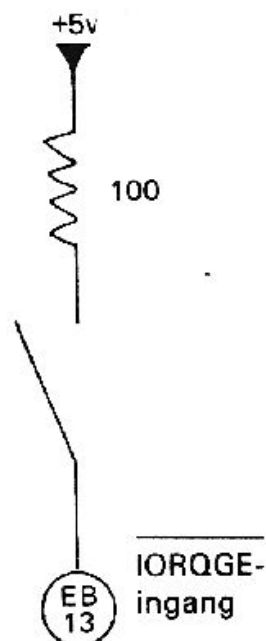
Afb. 14.3 $\overline{\text{HALT}}$ -test



Afb. 14.4 $\overline{\text{NMI}}$ -test



Afb. 14.5 $\overline{\text{INT}}$ -test



Afb. 14.6 $\overline{\text{IORQGE}}$ -test

```
10 CLS
20 PRINT AT 0,0;PEEK 23672+256*PEEK 23673
30 GO TO 20
```

Als u dit programma laat lopen zult u bovenaan uw scherm een oplopend getal zien. Dit geeft aan hoeveel keer de CPU onderbroken is sinds hij is ingeschakeld. Sluit nu de INT-schakelaar. De telling stopt. Probeer uw BASIC-programma te BREAKen. Er gebeurt niets! Er is echter niets verloren. Wanneer u de INT-schakelaar weer afzet zal de teller weer doorgaan en kunt u het toetsenbord weer gebruiken.

Waarom gebeurde dit? De werking van de teller en van het toetsenbord berust op onderbrekingen. Vijftig maal per seconde maakt de ULA de \overline{INT} -lijn actief. Hierdoor springt de CPU naar een klein stukje machinecode in het BASIC-ROM. Deze routine zorgt ervoor dat het toetsenbord wordt afgetast of er een toets is ingedrukt en verhoogt een teller van 3 geheugenbytes met één. Door de \overline{INT} met de +5 volt te verbinden zorgt u ervoor dat het onderbreeksignaal vanuit de ULA de CPU bereikt. De CPU tast het toetsenbord niet meer af en verhoogt zijn teller niet meer totdat hij een onderbreeksignaal ontvangt. Daarom werkt geen van de toetsen en stopt de teller.

Tik nu het volgende programma in.

```
10 FOR t=1 TO 30: BEEP 0.01,t= NEXT t: GO TO 10
```

Wanneer u dit programma laat lopen zult u een signaal met opklimmende toonhoogte horen. Probeer de \overline{INT} -schakelaar. Hij heeft geen effect op de toon. \overline{INT} beïnvloedt de uitvoer van de CPU naar de ULA niet. Hij beïnvloedt alleen de invoer van de ULA naar de CPU. Schakel de \overline{INT} weer uit. Sluit nu de \overline{IORQGE} -schakelaar. De toon stopt plotseling. Ook het toetsenbord wordt weer onbruikbaar, maar om een volledig andere reden. Kunt u bedenken waarom?

Alle I/U welke door de ULA wordt gedaan berust erop dat de ULA in staat is te ontdekken wanneer de CPU gegevens naar de ULA wil sturen of van de ULA wil ontvangen. De ULA gebruikt de adreslijnen, \overline{RD} , \overline{WR} en \overline{IORQGE} om te ontdekken of de CPU verbinding zoekt. Door nu de \overline{IORQGE} -lijn met de +5 volt te verbinden, verhindert u dat de CPU het \overline{IORQ} -signaal van de ULA ontvangt. Daarom kunnen er geen gegevens tussen de CPU en de ULA worden uitgewisseld. De 3-byte AAN-tijdsklok gaat door omdat de ULA nog steeds 50 maal per seconde onderbrekingen voor de CPU opwekt.

14.6 Het toevoegen van 128 I/U-poorten

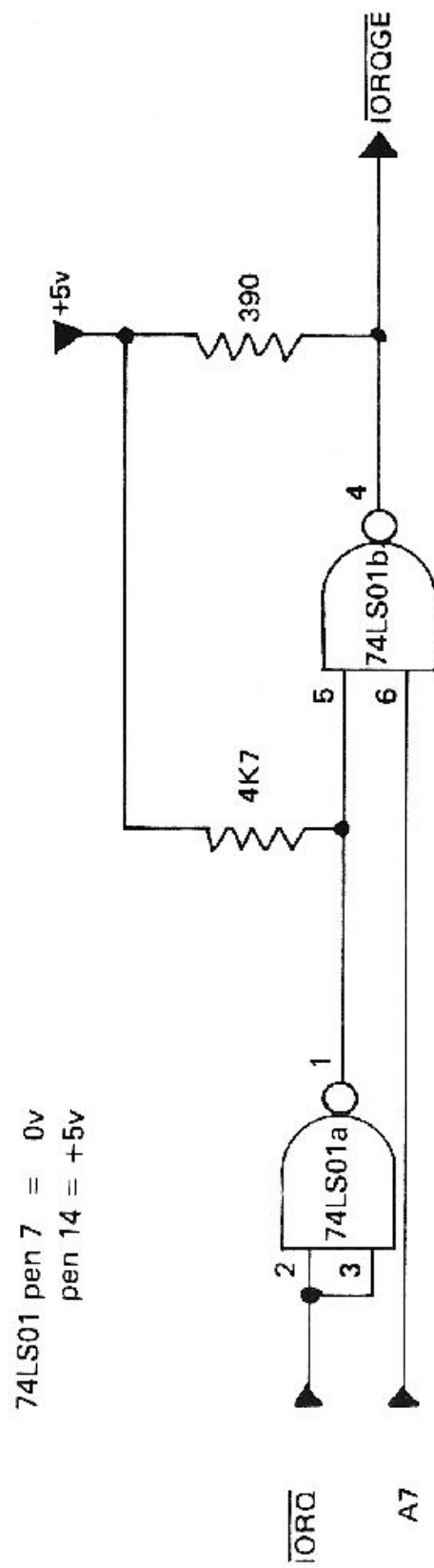
In de Spectrum wordt een zeer eenvoudige manier gebruikt om de adressen van de I/U-poort te decoderen. A0 wordt gebruikt om de ULA te selecteren, A1 voor de ZX-printer en A3, A4, A5 voor de microdrives en de RS232-uitgang.

Slechts één van deze adreslijnen mag logisch 0 zijn tijdens I/U-bewerkingen. Als dat het geval is, dan zijn er verschillende apparaten die de databus tegelijk willen gebruiken. Zelfgebouwde schakelingen kunnen daarom alleen de adreslijnen A5, A6 en A7 gebruiken voor de selectie. Deze adreslijnen zouden allemaal gebruikt worden om één PIO-chip te adresseren (zoals in hoofdstuk 15).

Een methode om A0 t/m A4 te gebruiken zou handig zijn. Dit is wat deze schakeling kan doen, maar er zijn enkele problemen.

Bekijken we de schakeling in afbeelding 14.7. De 74LS01-chip heeft een 'open collector'-uitgang die de stroom kan wegnemen maar niet op peil kan houden. Als A7 hoog is wordt \overline{IORQ} via \overline{IORQGE} gevoed zoals gewoonlijk. Echter wanneer A7 laag is, dan blijft \overline{IORQGE} hoog onafhankelijk van \overline{IORQ} .

In het geval van de ULA zal hij alleen als een I/U-apparaat geselecteerd worden wanneer A7=1 en A0=0. Als A7=0 zal hij in dit geval niet geselecteerd worden wat wel het geval zou zijn zonder deze extra schakeling. Als u de ZX-printer



Afb. 14.7 Schema van de 128 in/uitvoerpoorten

en andere randapparatuur wilt gebruiken met het $\overline{\text{IORQ}}$ in plaats van het $\overline{\text{IORQGE}}$ -signaal dan moet de schakeling van afbeelding 19h veranderd worden. Het binnenkomende $\overline{\text{IORQ}}$ -signaal moet van de randconnector komen en het uitgaande $\overline{\text{IORQ}}$ -signaal moet dan verbonden worden met het inkomende $\overline{\text{IORQ}}$ -signaal van alle randapparatuur in plaats van het oorspronkelijke Z80A- $\overline{\text{IORQ}}$ -signaal.

Om te laten zien dat de schakeling werkt met de ULA probeer dit korte programma eens.

```
100 LET a= IN 63486
110 LET b= IN (63486-128)
120 CLS: PRINT a,b
130 PAUSE 5
140 GO TO 100
```

Dit programma leest de toetsen 1 t/m 5 van het toetsenbord. Wanneer u één van deze toetsen indrukt zult u zien dat de 'a'-waarde verschilt van 255 maar dat de 'b'-waarde onveranderd blijft. Wanneer u hetzelfde programma probeert zonder dat de extra schakeling is aangesloten, dan zullen zowel de 'a'- als de 'b'-waarde veranderen als u op één van de toetsen 1 t/m 5 drukt. 'b' wordt niet van de ULA gelezen als de schakeling ingeschakeld is omdat A7 dan 0 is, waardoor de I/U van de ULA wordt ontkoppeld.

OPMERKING: Deze schakeling kan niet gebruikt worden bij Spectrums model 2 die een transistor over de Z80-chip hebben.

15. Het toevoegen van een Z80A-PIO-chip

Deze parallel-in-en uitvoer (PIO)-chip stelt de Z80A-CPU in staat met de buitenwereld te communiceren. In dit hoofdstuk wordt een korte uitleg gegeven hoe de PIO werkt, hoe u hem moet programmeren en hoe hij met de randconnector moet worden verbonden. Het is niet de bedoeling een volledige en gedetailleerde beschrijving van de PIO-chip te geven. Deze informatie kan in boeken over Z80-interfacing gevonden worden. Hoofdstuk 18 bevat details over een 8-kanaals analoog naar digitaal omzetter als een bruikbaar voorbeeld van een handig apparaat dat door een PIO kan worden bestuurd. Een van de vele toepassingen is het uitlezen van de X- en Y-coördinaten van een stuurknuppel.

De Z80A-PIO-chip is speciaal voor gebruik met de Z80A-CPU ontworpen. Als u de pen aansluitingen in afbeelding 15.1 bekijkt, zult u veel bekende signalen tegenkomen. Er is de databus voor volledige tweerichtingsverkeer met de CPU, de klok, $\overline{M1}$ -, \overline{IORQ} -, \overline{RD} - en \overline{INT} -signalen zijn direct verbonden met de overeenkomstige pennen op de randconnector. Er zijn ook twee pennen voor de 'onderbreking inschakelen invoer'-en de 'onderbreking inschakelen uitvoer'-signalen. Deze zijn alleen van belang wanneer er onderbrekingen gebruikt worden. Onderbrekingen worden vooral gebruikt om de CPU te dwingen een speciaal stuk machinecode af te handelen.

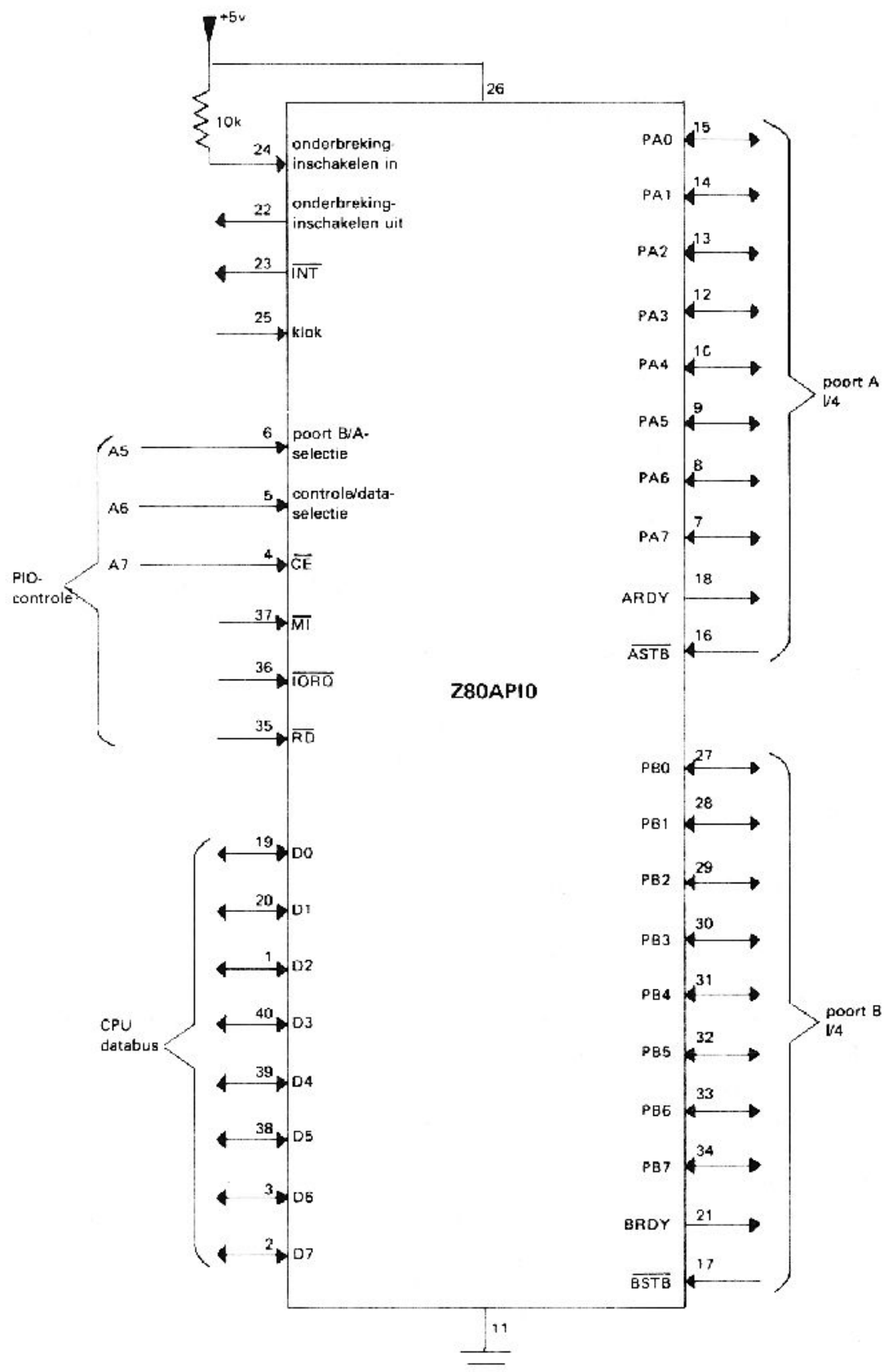
Bijvoorbeeld, de Spectrum wordt gebruikt om een BASIC-programma te draaien en tegelijkertijd de centrale verwarming te sturen. De CPU wil geen tijd verkwisten met het controleren van de kamertemperatuur (door een signaal van de kamerthermostaat) want dit vertraagt het BASIC-programma. Daarom gebruiken we onderbrekingen.

Wanneer de temperatuur te laag of te hoog wordt zal het thermostaat-circuit de CPU onderbreken, die dan gedwongen wordt het centrale verwarmingsprogramma uit te voeren.

Wanneer dit gedaan is kan teruggekeerd worden naar het BASIC en de verwarming vergeten worden tot de volgende onderbreking.

Wanneer u de andere aansluitingen bekijkt dan ziet u dat er twee I/U-poorten zijn. Deze worden poort A en poort B genoemd. Elke poort heeft 8 I/U-lijnen (PA0 t/m PA7 en PB0 t/m PB7) en twee 'handshake' (aansluitbevestiging)-lijnen (\overline{ARDY} , \overline{ASTB} en \overline{BRDY} , \overline{BSTB}). In dit boek zullen alleen de acht I/U-lijnen gebruikt worden. De andere signalen worden gebruikt bij BYTE-data-overdracht tussen in/uitvoerapparaten (meestal van verschillende computers).

Om het programmeren van de PIO-chip te demonstreren volgt er een praktisch voorbeeld met een uitleg. De bedoeling is om PA0 tot PA1 als ingangen en PA2 tot PA7 als uitgangen te definiëren. U zult dan zien hoe deze door de CPU ge-



Afb. 15.1 Penaansluitingen van de Z80A-parallel in/uitvoerchip

bruikt kunnen worden om de schakelaars en de lampen te bedienen. Dit voorbeeld gebruikt poort A, hoewel poort B op precies dezelfde manier gebruikt kan worden door controlewoorden en datawoorden naar poort B te zenden. De schakeling die in afbeelding 20b staat moet op hetzelfde bord als de PIO-chip worden gebouwd. Het 74LS05-IC wordt gebruikt om lichtuitzendende dioden (LED's) te schakelen want de uitvoer van de PIO-chip kan daarvoor niet genoeg stroom leveren. Elk type LED kan in deze schakeling gebruikt worden.

Als u een 16K Spectrum heeft dan is de +5 volt voeding voldoende om alle schakelingen uit dit boek te voeden. Bij een 48K Spectrum kan de +5 volt voeding onvoldoende zijn om de stroom én voor het extra 32K geheugen én de bijkomende schakelingen te leveren. Is dat bij u het geval dan is een extra +5 volt voeding noodzakelijk. Een bruikbare schakeling staat in hoofdstuk 3 beschreven.

Voordat de PIO-chip kan werken zoals u wilt, moet u hem vertellen wat van hem verlangd wordt. Dit wordt gedaan door er een controlewoord naar toe te sturen. De vorm van een controlewoord is:

D7	D6	D5	D4	D3	D2	D1	D0
M1	M0	x	x	1	1	1	1
status-woord		bepaalt status die wordt ingevoerd					

x= niet gebruikt door de PIO, kan dus 0 of 1 zijn.

Het statuswoord kan elk getal van 0 tot en met 3 zijn.

status 0 = uitvoerbyte met handshaking

status 1 = invoerbyte met handshaking

status 2 = in/uitvoerbyte met handshaking

status 3 = controlestatus

Omdat we de controlestatus willen gebruiken moet het statuswoord op 3 gezet worden. Het controlewoord is daarom: 1 1 1 1 1 1 1 = 255 decimaal

Dit kan naar de controlepoort van poort A gezonden worden. In BASIC kunt u: OUT CA,255 gebruiken. CA = controlepoort-adres van poort A en hangt af van hoe de PIO-chip met de adresbus verbonden is (zie verderop).

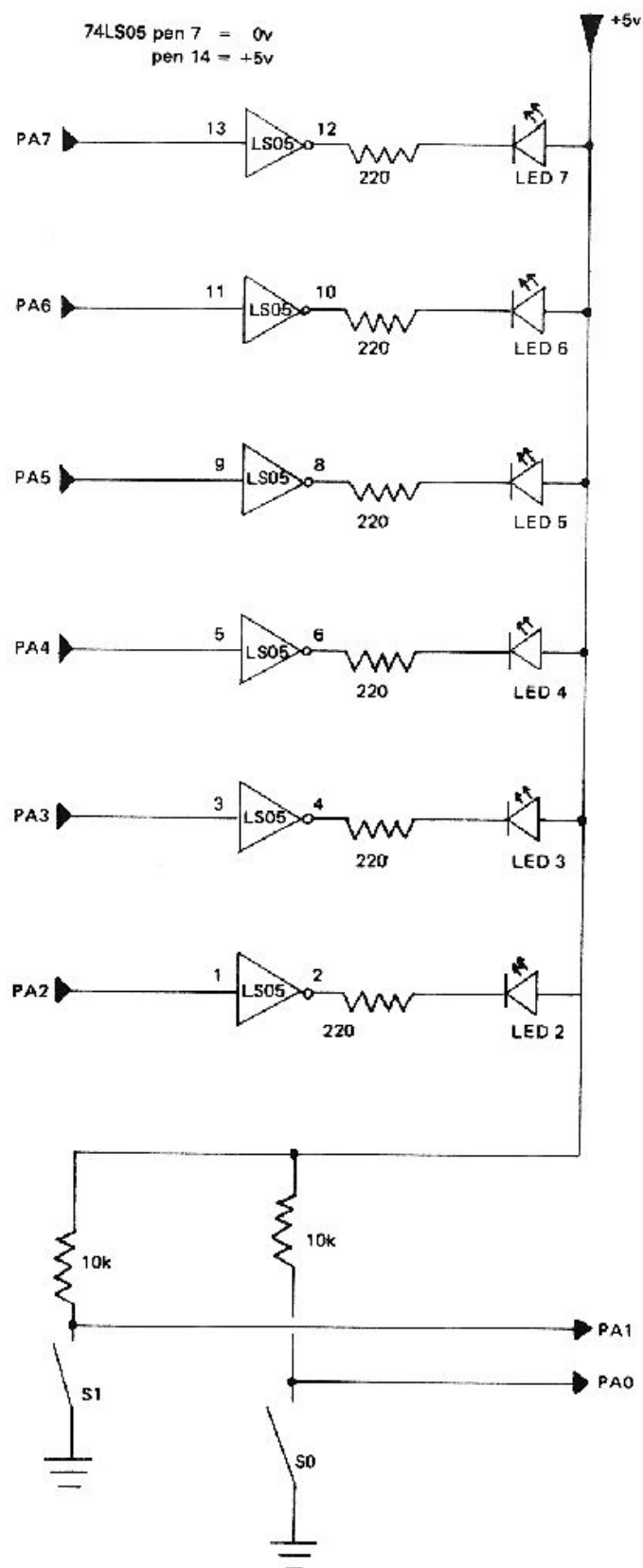
De PIO-chip moet nu weten welke lijnen van poort A voor invoer en welke voor uitvoer gebruikt gaan worden. Daarom wordt er een ander controlewoord naar poort A gezonden waarin de juiste databits op 0 gezet worden voor uitvoer en op 1 gezet worden voor invoer. Dus:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	1	1

wanneer u dit naar CA stuurt worden D0 t/m D1 als invoer en D2 t/m D7 als uitvoer gedefinieerd. In BASIC kunt u OUT CA,3 gebruiken.

Voor u één van de BASIC IN- en OUT-opdrachten die hierboven zijn beschreven kunt geven, moet u eerst CA en DA definiëren plus CB en DB wanneer u poort B wilt gebruiken.

De Spectrum zelf gebruikt A7, A6 en A5 niet voor I/U-apparaten, dus worden die gebruikt om de PIO-chip te sturen. A7 wordt, als hij is aangesloten zoals in afbeelding 15.1 staat, gebruikt om de PIO-chip in te schakelen wanneer hij 0 wordt. A6



Afb. 15.2 Schema van de LED's en schakelaarsproef

bepaalt of er een controlewoord (A6=0) of een datawoord (A6=1) naar de PIO wordt gezonden. A5 selecteert of poort A (A5=0) of poort B (A5=1). A4 tot en met A0 moeten logisch 1 blijven, zodat de gewone Spectrum-I/U-functies niet verbroken worden.

Geselecteerd	Binair	Decimaal	Token
Poort A data	0 0 0 1 1 1 1 1	31	DA
Poort A controle	0 1 0 1 1 1 1 1	93	CA
Poort B data	0 0 1 1 1 1 1 1	63	DB
Poort B controle	0 1 1 1 1 1 1 1	127	CB

15.1 Proeven met LED's en schakelaars

Begin de BASIC-programma's altijd met:

```
10 REM Poort A initialisatieprogramma
20 LET DA = 31
30 LET CA = 93
40 OUT CA,255
50 OUT CA,3
```

Hierdoor wordt poort A op de juiste manier voorbereid voor uw programma.

Invoer

Het is mogelijk om elk van de binaire getallen 00,01,10 en 11 overeenkomend met 0 t/m 3 decimaal met schakelaars in te stellen. Probeer de volgende programmalus om te zien hoe het weergegeven getal verandert met de verschillende standen van de schakelaars.

```
100 LET x = IN DA
110 PRINT 'De invoerschakelaar staat op';x
120 GO TO 100
```

Uitvoer

De LED's stellen u in staat om een binair getal tussen 0 en 63 weer te geven. Het volgende programma doet dit voor u:

```
100 INPUT 'Welk getal?';A
110 LET A = A*4
120 OUT DA,A
130 GO TO 100
```

Nu u gezien heeft hoe de in- en uitvoer met de PIO gebeurt, kunt u een eenvoudig programma schrijven dat automatisch binair omhoog telt. Door een schakelaar om te zetten kan het programma overgaan op binair omlaag tellen.

16. Het toevoegen van uw eigen toetsenbord

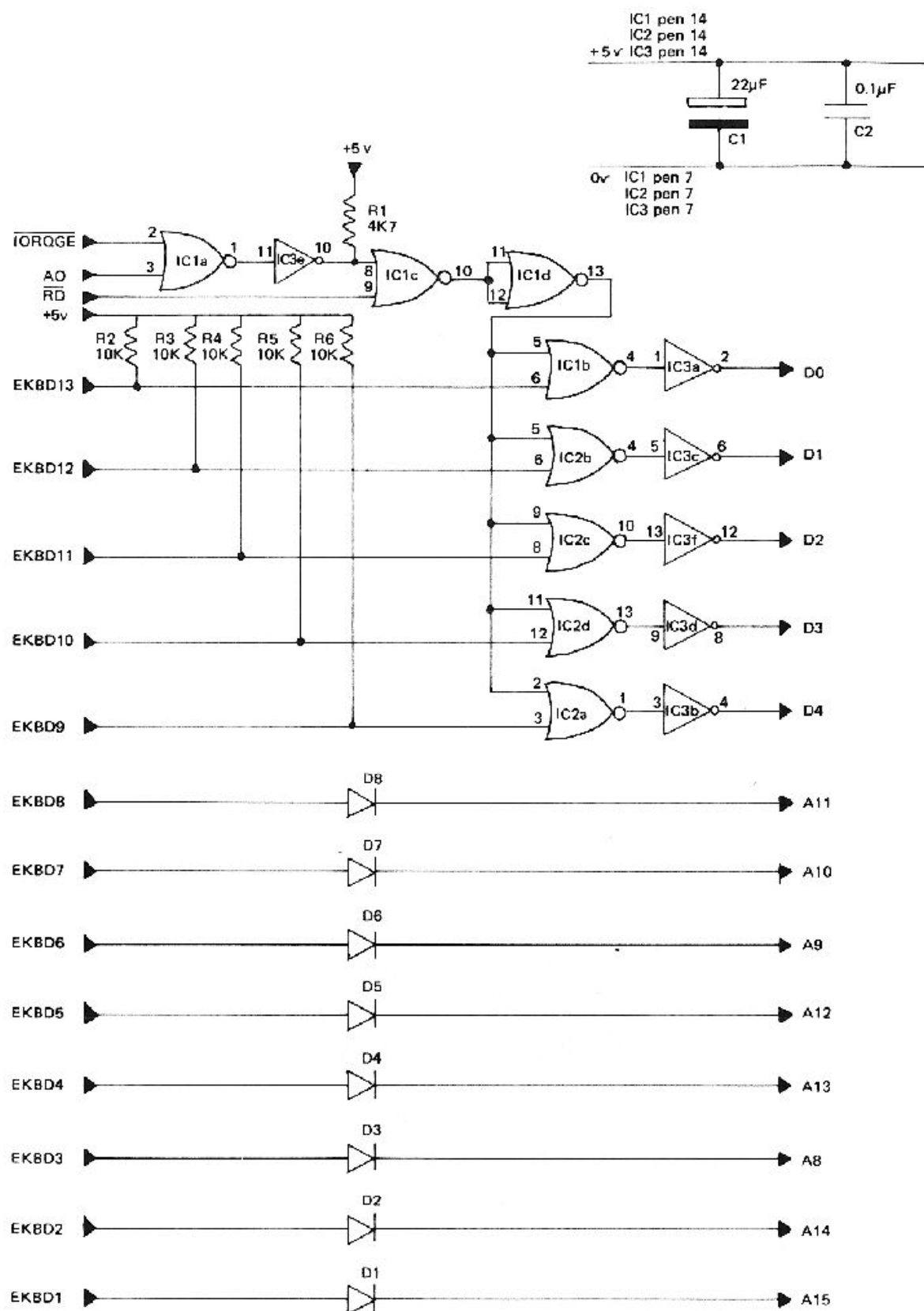
16.1 Inleiding

Dit hoofdstuk legt tot in de details uit hoe u een toetsenbord-interface kunt bouwen dat direct op de randconnector wordt aangesloten. Het werkt parallel met het originele Spectrum-toetsenbord. U kunt dus twee toetsenborden tegelijkertijd gebruiken. Hoofdstuk 17 legt uit hoe u dit toetsenbord-interface kunt gebruiken voor stuurknuppels die u met de Spectrum kunt gebruiken. De mogelijkheden zijn enorm groot. U kunt een numeriek toetsenbord aansluiten of een hexadecimaal toetsenbord, voor het programmeren in machinecode, of een volledig groot toetsenbord.

16.2 Beschrijving van de schakeling

De schakeling waarover gesproken wordt staat in afbeelding 16.1. De toetsenbordaansluitingen EKBD1 t/m EKBD13 in deze schakeling komen overeen met de aansluitingen KDB1 t/m KDB13 in afbeelding 7.2. De signalen \overline{IORQGE} , A0 en \overline{RD} zijn gecombineerd zodat de gemeenschappelijke ingang op de NOR-poorten IC1b en IC2a t/m IC2d, alleen laag is als al deze signalen tegelijk laag zijn. Het extra toetsenbord-interface wordt daarom geselecteerd als de ULA (poort 254) geadresseerd wordt voor uitlezing. Wanneer geen van de toetsen is ingedrukt worden de ingangen van de NOR-poorten hoog gehouden door de weerstanden R2 t/m R6. De uitgangen van de NOR-poorten zijn laag en worden door IC3 geïnverteerd zodat er een hoog signaal op de databuslijnen D0 t/m D4 komt. Omdat de zes buffers van IC3 een 'open collector'-uitgang hebben, kan de ULA gemakkelijk D0 t/m D4 logisch 0 maken wanneer een van de Spectrum-toetsen wordt ingedrukt.

Stelt u zich voor dat EKBD10 verbonden is met EKBD6. Wanneer u naar afbeelding 7.2 kijkt ziet u dat dit betekent dat de 'F'-toets is ingedrukt. Als er nu uitgelezen wordt met A9 laag dan wordt de invoer van IC2d omlaag getrokken. De uitgang van IC2d wordt hoog en D3 wordt door IC3d omlaag getrokken. De CPU leest dan D0 t/m D4 direct van de databus en weet dat de 'F'-toets is ingedrukt.



Afb. 16.1 Schema voor een extra toetsenbord

Onderdelenlijst

Weerstanden

R1 4K7

R2 t/m R6 10K

allen 5% 1/4 watt

Condensatoren

C1 22 μ F 6 volt elektrolytisch

C2 0,1 μ F keramische schijf

Halfgeleiders

IC1 74LS02

IC2 74LS02

IC3 74LS05

D1 t/m D8 IN4148 diodes

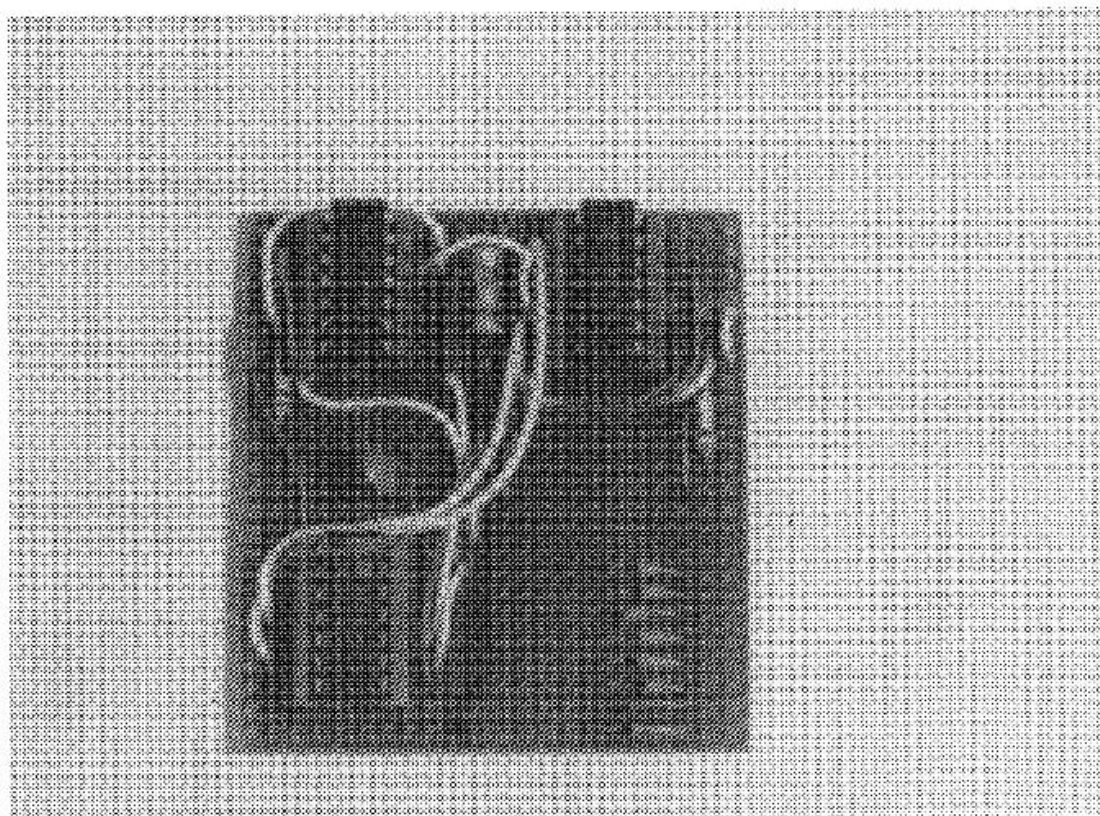
Diversen

3 stuks 14-pens DIL IC-voeten

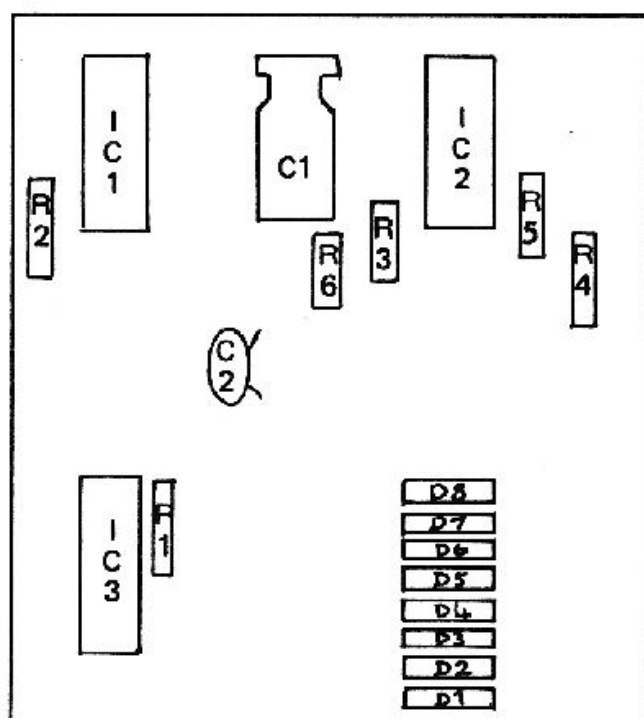
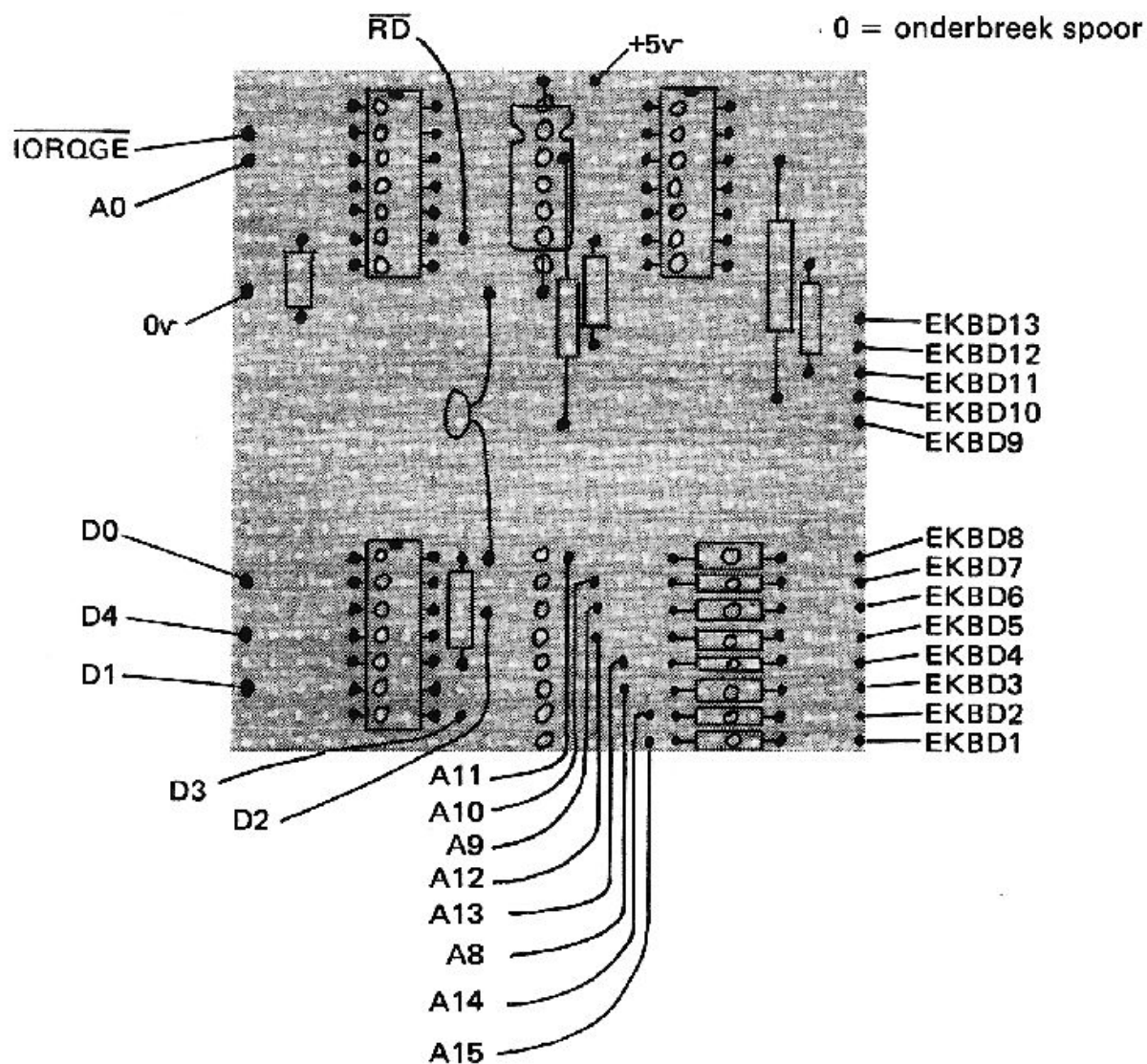
28-polige Spectrum randconnector

60*70 mm (of groter) verobord 0,1" steek

draad en soldeer



Afb. 16.2 Foto van het toetsenbord-interface



Afb. 16.3 Extra toetsenbord/stuurknuppel, opstelling van de onderdelen

16.3 Bouwaanwijzingen

Deze schakeling kan op verobord of op een experimenteerbord gebouwd worden. In afbeelding 16.2 staat de opstelling van de onderdelen aangegeven voor het verobordontwerp. De IC-voeten moeten voor de 3 IC's gebruikt worden, waardoor deze niet door oververhitting bij het solderen beschadigd worden. Denk erom dat u alle koperbanen onderbreekt zoals is aangegeven in afbeelding 16.3. Een boortje van 4,5 mm is hiervoor erg handig (zie bouwtips in hoofdstuk 14). De volledige schakeling met alle onderdelen en de verbindingsdraden ziet u in afbeelding 16.1. De punt van de pijlen op de diodes D1 t/m D8 in het schema worden meestal door een band op het goede einde van het onderdeel aangegeven.

Nu zult u het één of andere toetsenbord willen aansluiten. Het volgende hoofdstuk bespreekt hoe stuurknuppels moeten worden aangesloten. Er zijn vele soorten toetsenborden te verkrijgen die aangesloten kunnen worden. Het grootste probleem is dat alle toetsenborden op een verschillende manier bedraad zijn. Daarom is het niet mogelijk precieze informatie te geven hoe de verschillende toetsenborden moeten worden omgebouwd. Uiteindelijk moet het toetsenbord bedraad zijn overeenkomstig het Spectrum-toetsenbord zoals in afbeelding 7.2. Bijvoorbeeld, de 'I'-toets moet tussen EKBD4 en EKBD11 zitten. De '7'-toets tussen EKBD5 en EKBD10 enzovoorts. Als u het herbedraden niet leuk vindt kunt u ook losse toetsschakelaars kopen. Het is tamelijk eenvoudig deze op een groot stuk verobord in een 5*8 patroon te monteren.

17. Stuurknuppels aangepast aan de Sinclair

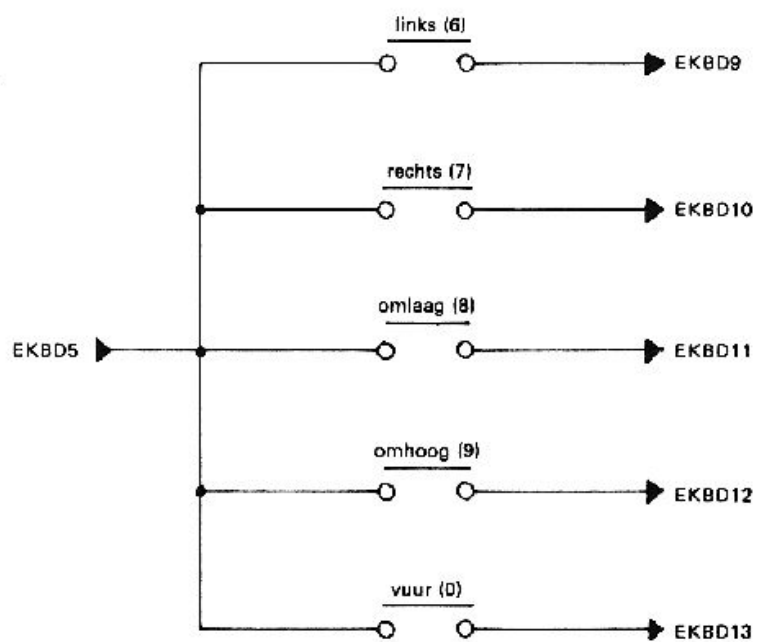
Met een stuurknuppel (joystick) wordt in dit verhaal een apparaat bedoeld dat in de hand gehouden wordt en waarmee bewegingsopdrachten aan programma's worden gegeven. De Sinclair-stuurknuppels hebben de functies LINKS, RECHTS, OP, NEER en VUUR. Het deel dat in de hand gehouden wordt kan uit een handgreep bestaan die één van de vier schakelaars bedient als hij vooruit, achteruit of zijdelings bewogen wordt. Een andere mogelijkheid met compact opgestelde drukknoppen ziet u in afbeelding 17.3. De vier buitenste knoppen verzorgen de beweging en de middelste knop is de vuurknop.

De Sinclair-stuurknuppels komt wat de software betreft overeen met de bovenste rij toetsen van het toetsenbord, zowel in BASIC als in machinecodeprogramma's. Stuurknuppel 1 komt aan invoerpoort 61438 en stuurknuppel 2 aan invoerpoort 63486. De functies van de stuurknuppels komen als volgt overeen met de cijfertoetsen:

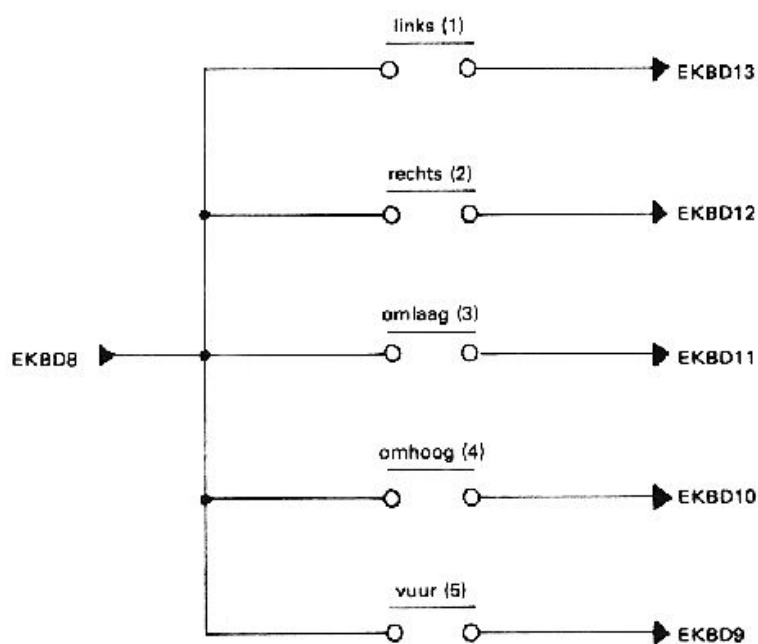
Toets	functie	
1	LINKS	
2	RECHTS	
3	OMLAAG	STUURKNUPPEL 2
4	OMHOOG	
5	VUUR	
6	LINKS	
7	RECHTS	
8	OMLAAG	STUURKNUPPEL 1
9	OMHOOG	
0	VUUR	

Om deze schakeling met de Spectrum te verbinden is het toetsenbord-interface uit hoofdstuk 16 nodig. Alleen de diodes D8 en D5 worden gebruikt. De andere diodes kunnen worden weggelaten.

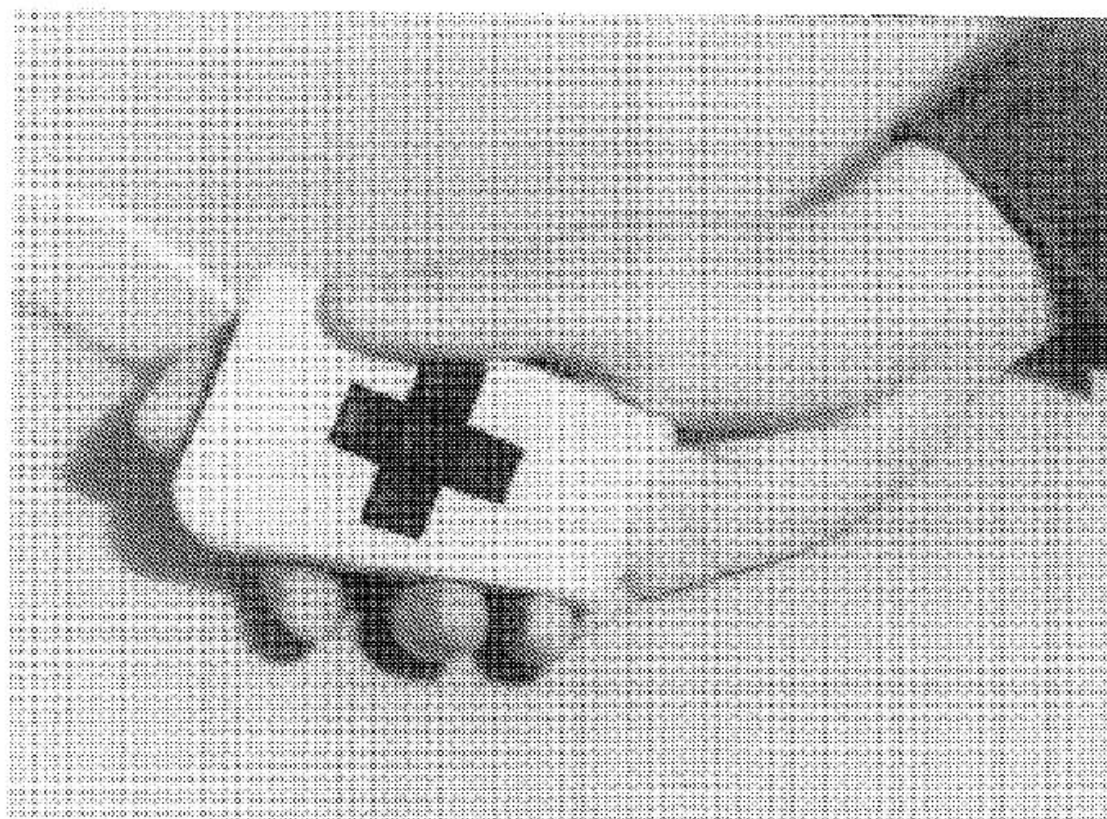
Het apparaat dat in afbeelding 17.3 staat gebruikt 5 drukknoppen voor de controle. De inwendige opstelling van de knoppen die op een stukje verobord zijn gesoldeerd ziet u in afbeelding 17.4. De precieze opstelling en de soort drukknoppen wordt aan u overgelaten. Inwendig moeten de schakelingen bedraad worden zoals in afbeelding 17.1 staat voor stuurknuppel 1 en zoals in afbeelding 17.2 voor stuurknuppel 2. De verbindingen met het toetsenbord-interface staan duidelijk in deze schema's aangegeven. Wanneer u een compleet gebouwde stuur-



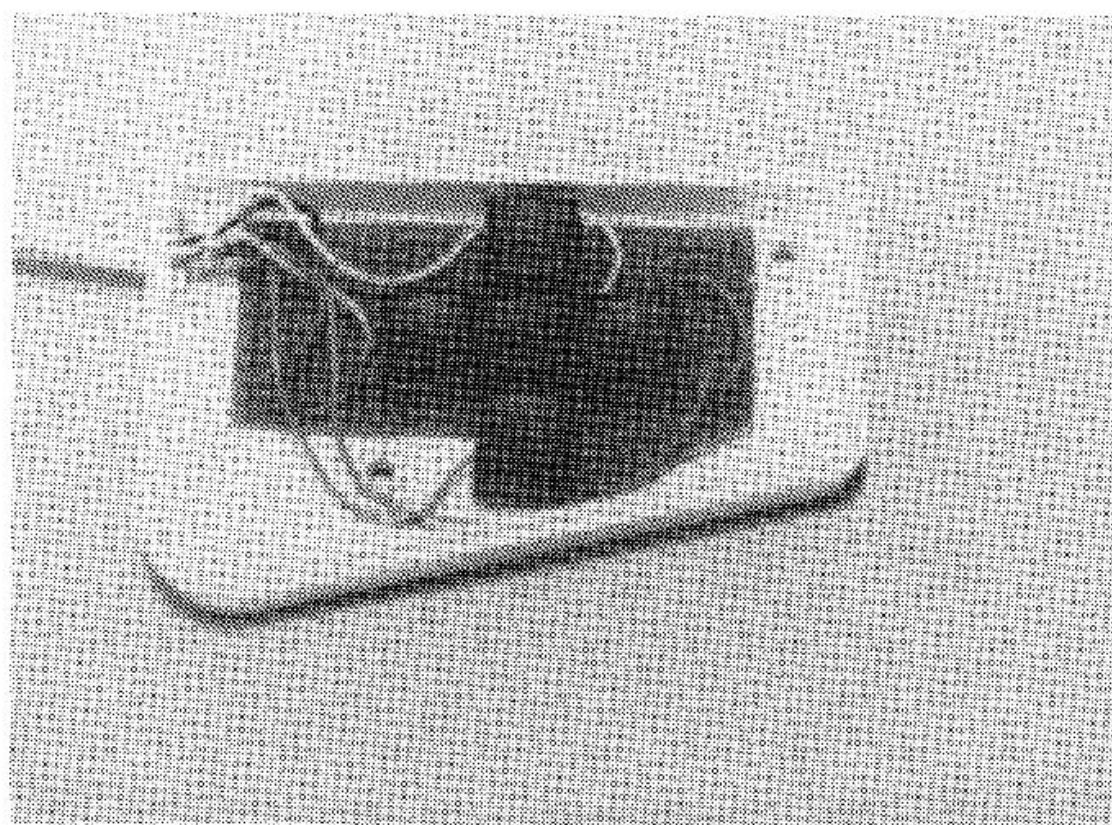
Afb. 17.1 Schema van de stuurknuppel nr 1 verbindingen



Afb. 17.2 Schema van de stuurknuppel nr 2 verbindingen



Afb. 17.3 Foto van het uitwendige van de stuurknuppel



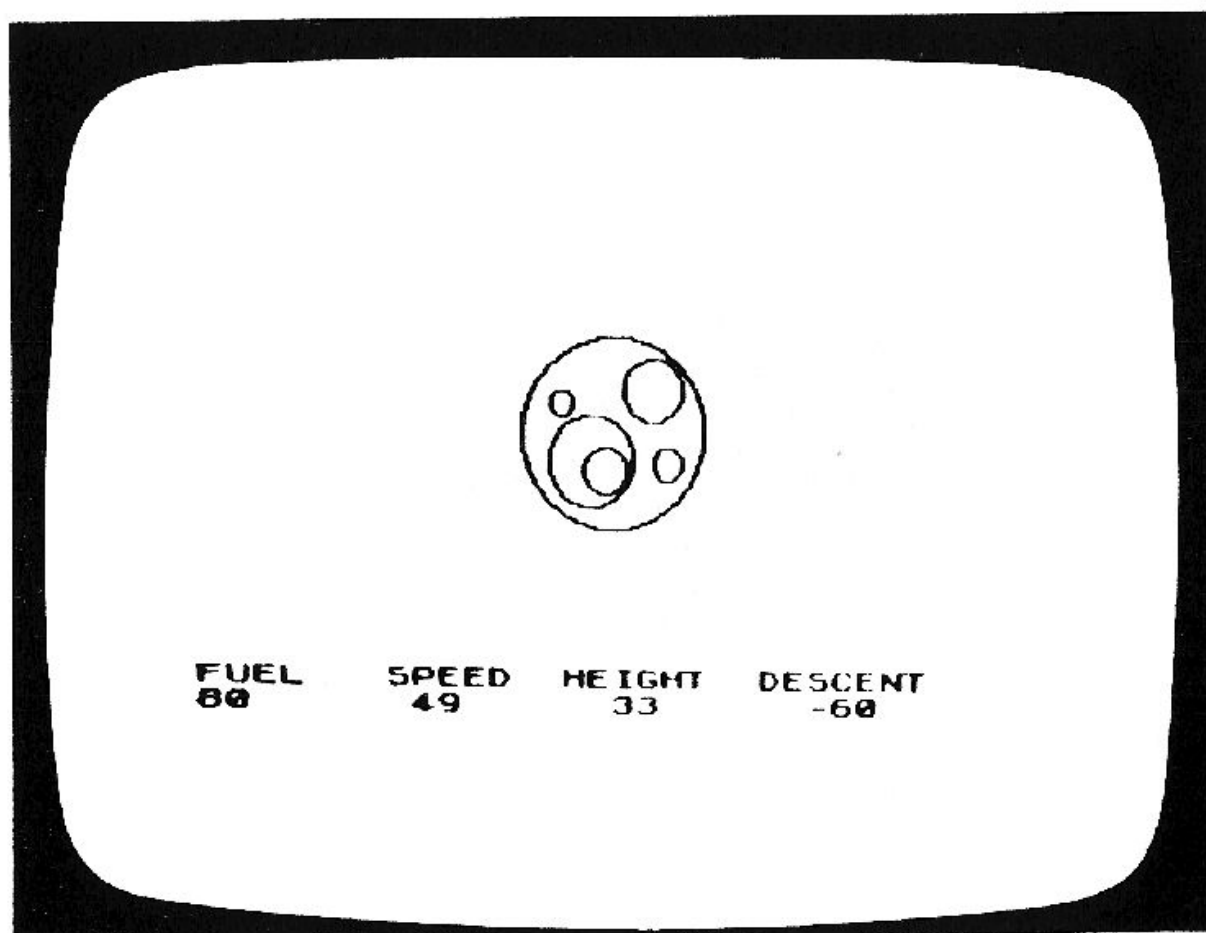
Afb. 17.4 Foto van het inwendige van de stuurknuppel

knuppel kunt bemachtigen die schakelt zoals in afbeelding 17.1 en 17.2 is aangegeven dan kan die met het interface verbonden worden in plaats van de zelfgebouwde schakeling.

Wat u nu nog nodig heeft is wat software om de stuurknuppels bij te gebruiken. Hierna volgt een programma dat 'Apollo' heet en waarmee u met de stuurknuppels kunt experimenteren. Stuurknuppel 2 of de toetsen 1 t/m 5 worden gebruikt.

17.1 Beschrijving 'Apollo'-programma

Het doel van het programma is om een maanlander met succes op het maanoppervlak neer te zetten. Aan het begin van het spel bent u in een cirkelvormige baan. De stuurknuppel bedient uw stuurraketjes. Deze geven een beweging van de maan af (OMHOOG), naar de maan toe (OMLAAG), toename van de tangentiële snelheid (RECHTS) en afname van de tangentiële snelheid (LINKS). U gebruikt alleen maar brandstof wanneer u een stuurraket laat werken. Een voortdurend bijgewerkte stand van de overgebleven brandstof, de snelheid, de hoogte boven het maanoppervlak en de daalsnelheid verschijnen op het scherm. De bedoeling is om op het oppervlak te landen met een kleine daalsnelheid en een kleine voorwaartse snelheid. Wanneer één van beide te groot is verongelukt u.



Afb. 17.5 Schermfoto van het 'Apollo'-spel

Wanneer de brandstof op is werken de stuurraketjes niet meer en bent u overgeleverd aan de werking van de zwaartekracht! De VUURknop moet alleen in uiterste nood gebruikt worden. Hiermee vuurt u de superstuwketten af en schiet u weg vanaf de maan. Gebruik deze mogelijkheid met zorg. De superstuw-raket gebruikt zeer veel brandstof en is zo krachtig dat u in enkele seconden voor eeuwig uit uw baan bent geblazen... De normale zwaartekrachtwetten gelden niet. Verminder uw baansnelheid en u zakt in een lagere baan waardoor uw snelheid weer toeneemt. Door met volle kracht naar het maanoppervlak te sturen komt u dichterbij, maar tegelijkertijd neemt uw baansnelheid snel toe waardoor u door de centrifugaalkracht van het oppervlak wordt weggedrukt in de ruimte. Goede landing.

Programma Apollo

```

100 REM apollo maan lander
110 REM copyright : A.C.DICKENS
120 REM december 1982
130 LET r1=30: LET s=0
140 REM Teken de maan
150 CLS : CIRCLE 127,87,r1
160 CIRCLE 110,96,4
170 CIRCLE 140,100,10
180 CIRCLE 125,75,7
190 CIRCLE 145,77,5
200 CIRCLE 120,78,14
210 PRINT AT 19,0;"Brand      Snel   Hoogte   Daal"
215 PRINT AT 20,0;"stof      heid           snelh."
220 LET x=127
230 LET y=157
240 LET r=70
250 LET v=1
260 LET kt=70
270 LET p=.095
280 LET w=1/70
290 LET am=70
300 LET p1=p/3
310 LET u=0
320 LET k=0
330 LET fu=100
340 LET fl=.2
350 REM stuorkuppel invoer
360 IF fu=0 THEN LET e=31: GO TO 380
370 LET e=IN 63486-224
380 LET f=INT (e/16): REM Vuur-knop
390 LET e=e-f*16
400 LET f1=INT (e/8): REM Links-knop
410 LET e=e-f1*8
420 LET f3=INT (e/4): REM Neer-knop
430 LET e=e-f3*4
440 LET f2=INT (e/2): REM Rechts-knop
450 LET e=e-f2*2

```

```

460 LET f4=INT e: REM Omhoog-knop
470 LET f1=(1-f1)*p
480 LET f2=(1-f2)*p1
490 LET f3=(1-f3)*p
500 LET f4=(1-f4)*p1
510 IF f=0 THEN LET fu=fu-10: LET f1=5*p
520 IF f1+f2+f3+f4>0 THEN LET fu=fu-f1
530 REM Bereken volgende plaats
540 LET f=(kt/r-v*v)/r
550 LET am=am+(f2-f4)*r
560 LET u=u+(f1-f3-f)
570 LET r=r+u
580 LET v=am/r
590 LET j=v/r
600 LET k=k+j
610 REM Verwijder vorige maanlander
620 PLOT INVERSE 1;x,y
630 PLOT INVERSE 1;x,y+1
640 PLOT INVERSE 1;x-1,y
650 LET x=127+r*SIN k
660 LET y=87+r*COS k
670 IF x<225 AND x>=1 AND y>=0 AND y<=174 THEN GO TO 710
680 LET x=0
690 LET y=0
700 GO TO 750
710 REM Teken nieuwe maanlander
720 PLOT x,y
730 PLOT x-1,y
740 PLOT x,y+1
750 IF fu<0 THEN GO SUB 950
760 LET h=r-r1
770 PRINT AT 21,0;INT (fu*10)/10;" ";
780 PRINT AT 21,8;INT (v*100);" ";
790 PRINT AT 21,14;INT h;" ";
800 PRINT AT 21,23;INT (u*100);" ";
810 IF h<=0 THEN GO TO 830
820 GO TO 350
830 REM Op het maanoppervlak
840 IF ABS v>.04 THEN GO TO 900
850 IF ABS u>.15 THEN GO TO 900
860 CLS
870 PRINT AT 20,0;"Veilige landing - goed gedaan!"
880 FOR x=0 TO 300: NEXT x
890 GO TO 10
900 REM Botsing met de maan
910 CLS
920 PRINT ; FLASH 1;"Te SNEL>>>U stort neer!!!"
930 PAUSE 200
940 GO TO 100
950 REM Zonder brandstof
960 PRINT AT 20,0; FLASH 1;"LEEG";
970 LET fu=0
980 RETURN

```

18. Acht-kanaals analoog-naar-digitaal omzetter

Inleiding

Door de vorige hoofdstukken zult u begrepen hebben dat de computer met digitale hoeveelheden werkt. Hierdoor heeft u maar twee mogelijkheden voor elke bit informatie. Het kan 0 of 1 zijn. In werkelijkheid zullen de hoeveelheden die we willen meten een breed scala van waarden omvatten.

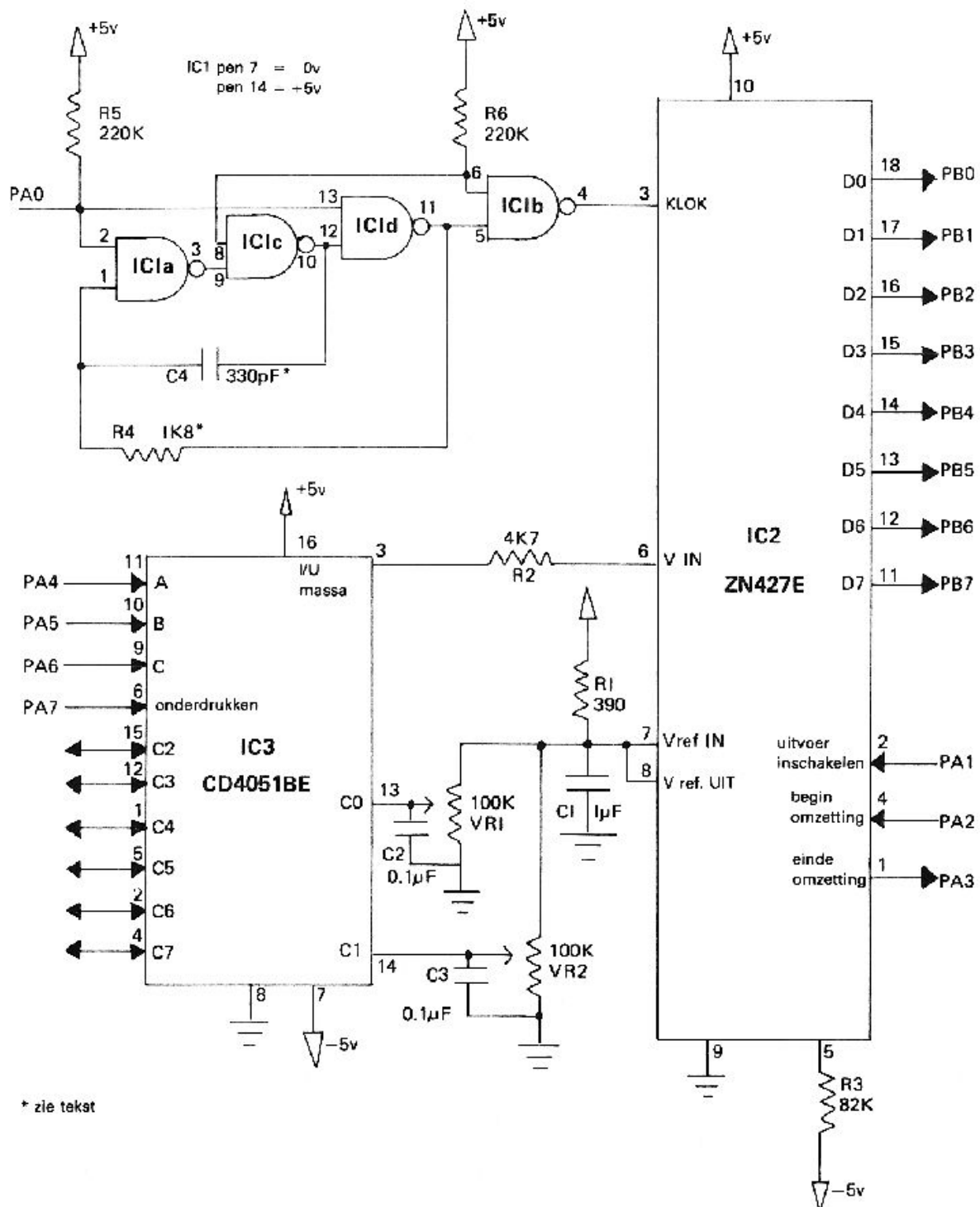
Enkele voorbeelden zijn: bloeddruk, temperatuur, lichtintensiteit of geluidsvolume. Deze parameters kunnen meestal als een wisselend voltage worden gemeten door een speciale omzetter. Een analoog naar digitaal omzetter (ADC) zal als invoer een wisselend voltage accepteren (bijvoorbeeld 0 tot 2,22 volt) en een digitale uitvoer geven die door een computer gelezen wordt. In deze schakeling wordt een 8-bit ADC gebruikt, waardoor zijn uitvoer kan variëren van 0 tot 255 in stappen van 1. Het circuit kan daarom met een nauwkeurigheid van 0,01 volt meten.

18.1 Bouwbeschrijving

De 8-kanaals ADC is direct met poort A en poort B van de PIO-chip, beschreven in hoofdstuk 15, verbonden. De schrijver gebruikt een 25-polige D-connector om de apparaten los te kunnen koppelen van de PIO. De gebruikte eenheid is afgebeeld in afbeelding 18.3. U ziet de D-connector aan de zijkant uitsteken. De opstelling van de onderdelen op 0,1 inch verobord ziet u in afbeelding 18.2.

Onderbreek de koperbanen zoals aangegeven voordat u gaat solderen. Het is verstandig om alle IC's in voeten te zetten en ze niet direct op het bord te solderen. Hierdoor kunt u defecte chips sneller vervangen en voorkomt u beschadiging door oververhitting tijdens het solderen. R4 en C4 hebben in afbeelding 18.1 en 18.2 een sterretje.

Dit zijn de tijdsbepalende onderdelen van het klokcircuit. De foto van de schakeling in afbeelding 18.4 toont R4 als een instelpotentiometer in serie met een vaste weerstand. Hiermee kunt u de oscillator op zijn maximum van 600 kHz zetten, hoewel het meestal niet nodig is dat hij zo snel loopt omdat BASIC zo langzaam is. Tenzij u een oscilloscoop of frequentieteller ter beschikking heeft is het moeilijk om te bepalen dat de frequentie niet boven de 600 kHz komt. Daarom is het verstandig om de instelpotentiometer R4 te laten vervallen en een vaste weerstand van 1K8 ohm daarvoor in de plaats te zetten. De verbindingen tussen de onderdelen moeten volgens het schema in afbeelding 18.1 gemaakt worden.



Afb. 18.1 Schema van de 8-kanaals analoog-naar-digitaal omzetter

Onderdelenlijst voor de 8-kanaals ADC

Weerstanden

R1 390 ohm

R2 4K7 ohm

R3 82K ohm

R4* 1K8 ohm (of 330 ohm in serie met een 2K2 instel, zie tekst)

R5 220K ohm

R6 220K ohm

alle weerstanden zijn 5% en 1/4 watt

Condensatoren

C1 1 μ F

C2 0.1 μ F

C3 0.1 μ F

C4 330 pf

C5 22 μ F 6V

IC's

IC1 CD4011BE

IC2 ZN427E

IC3 CD4051BE

Diversen

VR1, VR2 100K ohm de X- en Y-as van de stuurknuppel

0,1 inch verobord

IC-voetjes

doosje

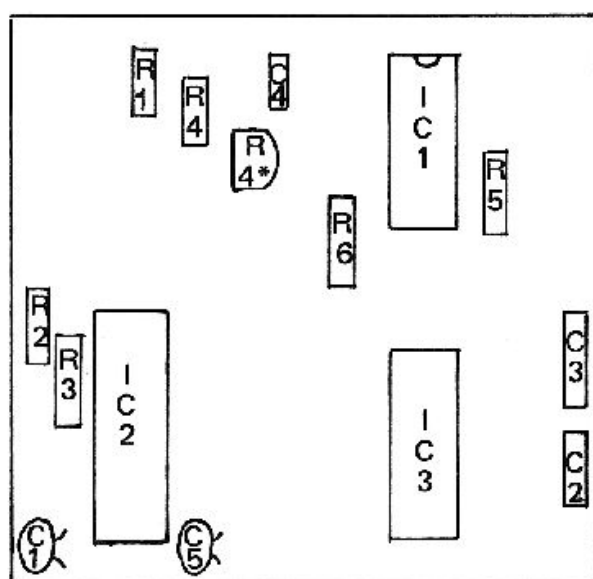
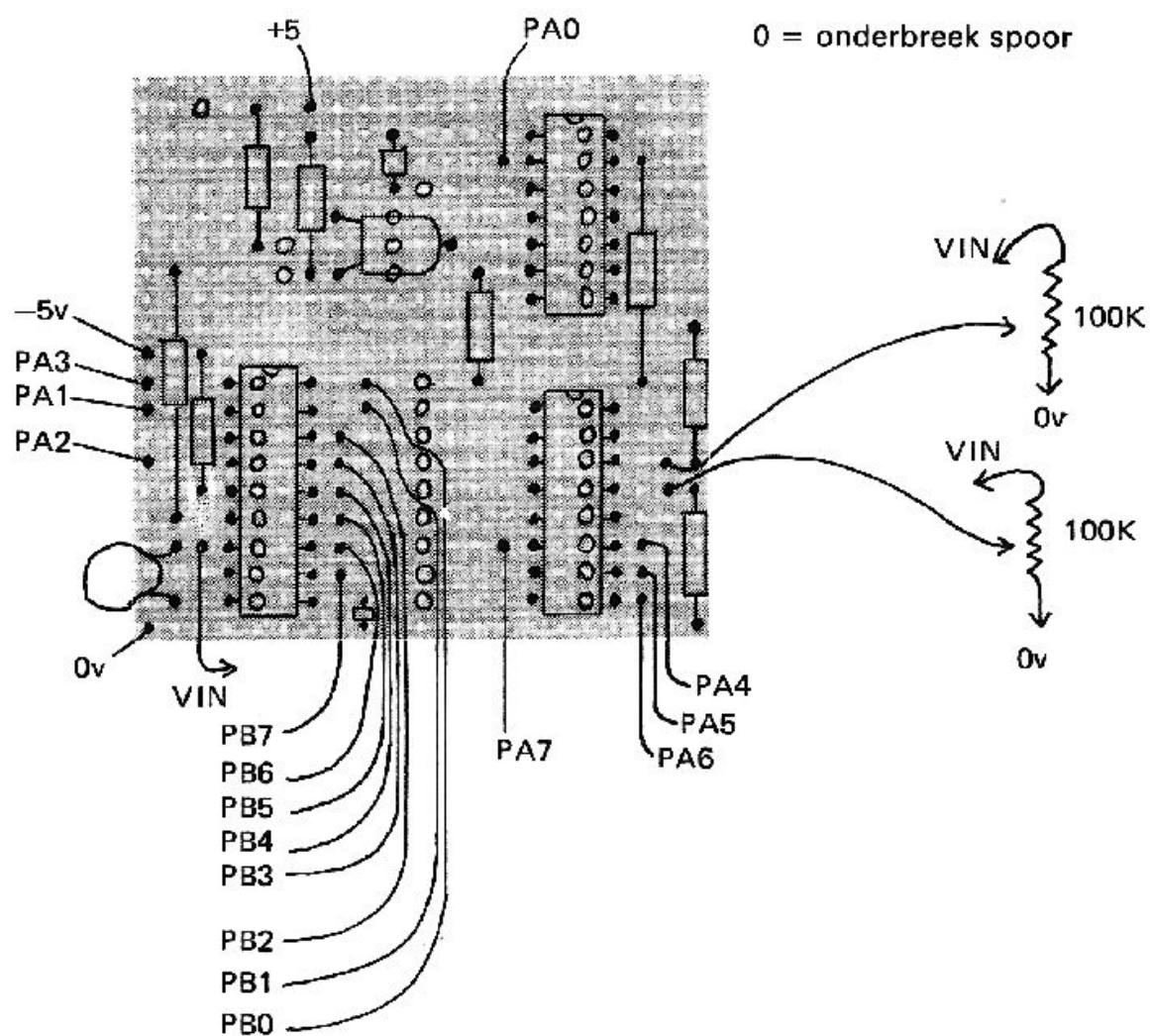
connector voor de PIO

Opmerking als u moeite heeft met het vinden van een twee-assige stuurknuppel dan kunt u eens gaan kijken bij één van de TANDY-winkels.

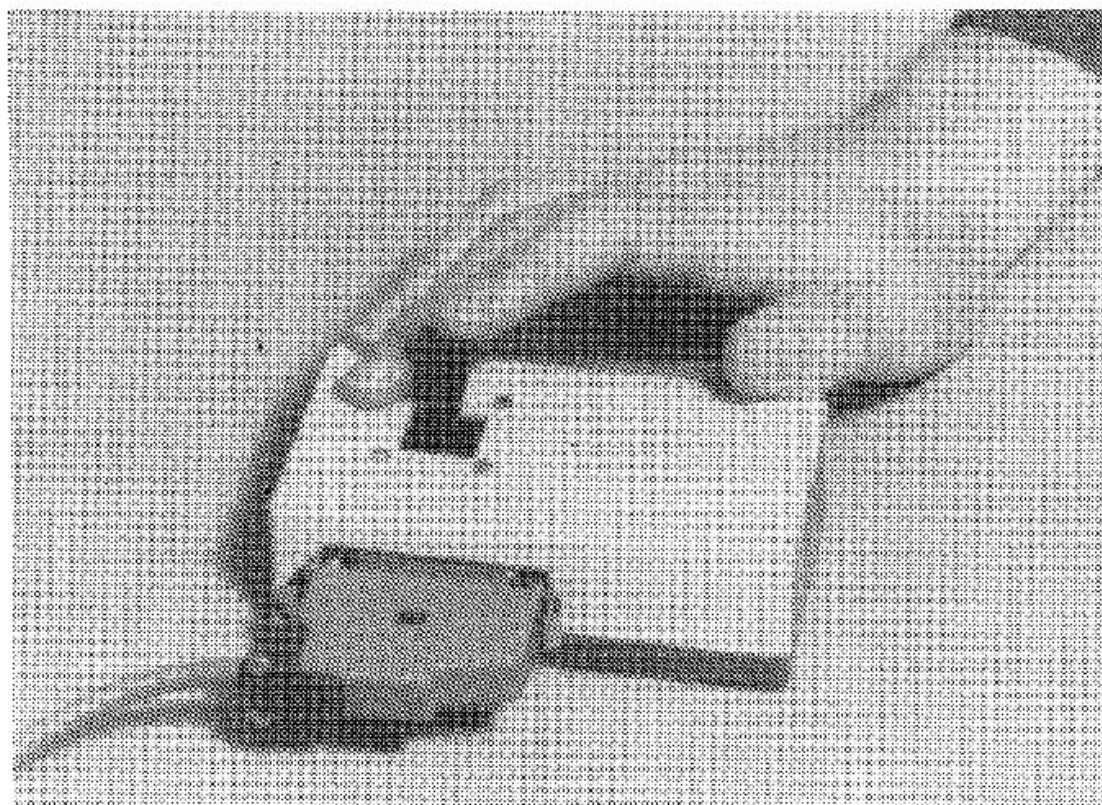
18.2 Hoe werkt het

Het hart van de schakeling is de analoog-naar-digitaal-omzetterchip (IC2). Hij accepteert een spanning op de V-in-invoer en zet het om in een 8-bits digitaal signaal op de uitgang. Om dit te kunnen doen heeft hij verschillende signalen van de PIO nodig en een klok. De klok is gelijk aan de Z80A-klok in uw Spectrum, maar werkt op een lagere frequentie. Bij elke klokpuls vertaalt de chip een bit. De frequentie van de klok is ongeveer 500 kHz en wordt aan- en uitgezet door PA0. Elke analoog-naar-digitaal-omzetting duurt daarom ongeveer 20 microseconden. De omzetting wordt begonnen door PA2 die de pen 'begin omzetting' activeert en pen PA0 die de klok inschakelt.

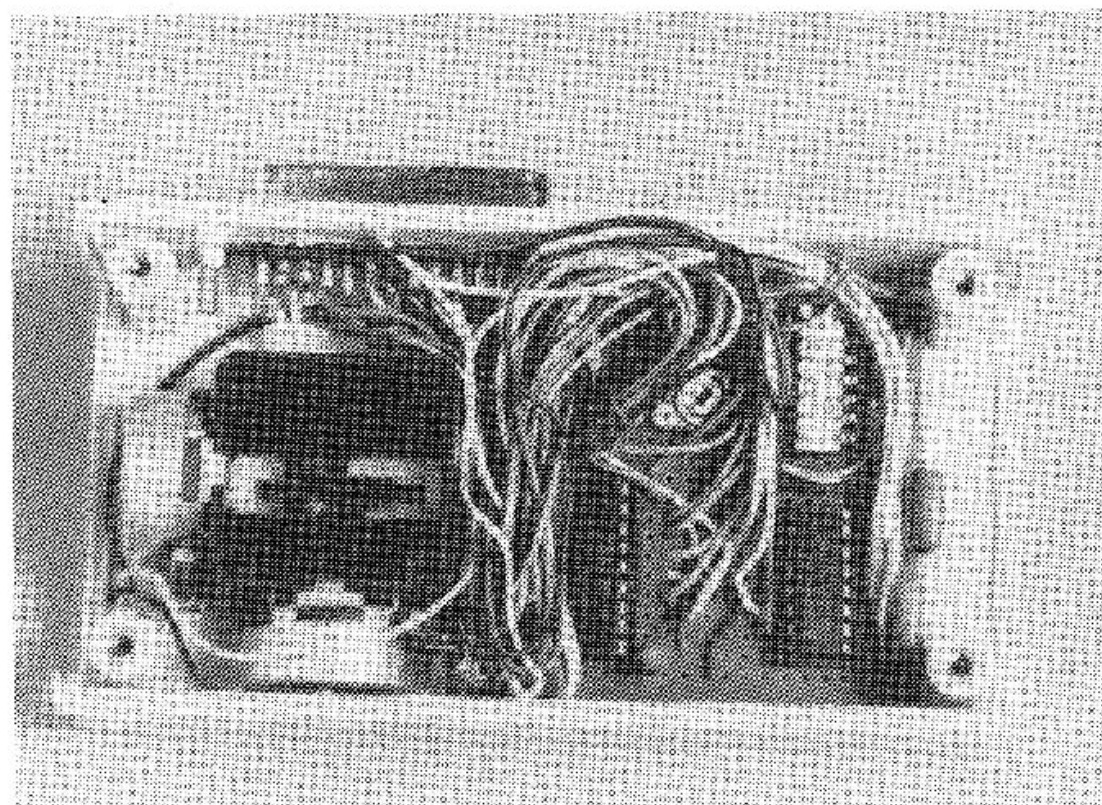
IC1 te zamen met de bijbehorende weerstanden en condensatoren vormen de klokschakeling. Als de omzetting klaar is, zendt IC2 een 'einde omzetting'-signaal naar PA3. Dit geeft aan dat het uitvoerdatabyte kan worden gelezen. PA1 bedient



Afb. 18.2 8-Kanaals analoog-naar-digitaal omzetter, opstelling van de onderdelen



Afb. 18.3 Foto van de 8-kanaals ADC met een stuurknuppel



Afb. 18.4 Foto van het inwendige van de 8-kanaals ADC

de uitvoerinschakelpen. Poort B van de PIO wordt dan gebruikt om het 8-bit data-byte van de ADC te lezen.

In het BASIC-voorbeeldprogramma, werken PA0, PA1, en PA2 tegelijkertijd. Het 'einde omzetting'-signaal is gegeven voordat het BASIC zijn coördinaten-LEES-routine heeft afgewerkt. Als er gebruik gemaakt was van een machinecoderoutine, dan was het noodzakelijk het 'einde omzetting'-signaal te gebruiken. Dit komt omdat machinecode veel sneller werkt dan BASIC. De ADC zou nog niet klaar kunnen zijn met de omzetting en dat moet gecontroleerd worden.

De ADC-chip is nogal duur, dus als we analoge spanningen van meerdere bronnen willen kunnen meten, is het handiger daarvoor één ADC-chip te gebruiken. Hiervoor is IC3 handig. Dit is een 1-naar-8-lijns analoog-multiplexer-chip. Een adres tussen 0 en 7 kan via PA4, PA5 en PA6 worden aangeboden. Hierdoor wordt één van de 8 ingangen via de multiplexer verbonden met de gemeenschappelijke uitgang. Als u bijvoorbeeld adres 0 kiest, zal de spanning op C0 bij de gemeenschappelijke I/U pen aankomen. De chip is vergelijkbaar met een 1-polige 8-standenschakelaar.

Het BASIC-tekenprogramma werkt met de X-en Y-assen van de stuurknuppel, die respectievelijk met adres 0 en 1 verbonden zijn. Regel 30 t/m 90 initialiseren de PIO-poort waardoor poort B ingang is en poort A uitgang, behalve PA3. Bekijken we regel 140 t/m 170 waar de X-coördinaat wordt gelezen. Eerst worden de uitgangen van poort A op nul gezet. Hierdoor stopt de klok, kiest de multiplexer kanaal 0 en wordt de ADC-chip losgekoppeld. Dan wordt de klok gestart in regel 160, tegelijk met het omzetsignaal aan de ADC-chip. Tegen de tijd dat regel 170 de X-coördinaat in poort B leest is de omzetting voltooid.

Het tekenprogramma in BASIC is erg eenvoudig te gebruiken. Er zijn twee mogelijkheden, 'D' en 'S', die worden gekozen door het indrukken van de overeenkomstige toets op het Spectrum-toetsenbord. Bij de 'S'-mogelijkheid kan een kleine blinkende stip over het scherm worden bewogen met de stuurknuppel zonder dat deze een spoor achter laat. Bij de 'D'-mogelijkheid laat de stip een spoor achter. U kunt op deze wijze tekeningen maken. Door op de 'S' te drukken stopt u het tekenen totdat u weer op de 'D' drukt. Om een nieuwe tekening te beginnen moet u op de 'N' drukken.

18.3 Tekenprogramma

```
10 REM Teken programma voor een stuurknuppel
20 REM
30 REM initialiseer de PIO poorten
40 LET da=31: LET ca=95
50 LET db=63: LET cb=127
60 OUT ca,BIN 11111111
70 OUT ca,BIN 00001000
80 OUT cb,BIN 11111111
90 OUT cb,BIN 11111111
100 LET a$="s"
110 LET x1=0: LET y1=0
120 REM initialiseer het scherm voor het tekenen
```

```

130 BORDER 7: PAPER 7: INK 0: CLS
140 REM lees de X coördinaat
150 OUT da,BIN 00000000
160 OUT da,BIN 00000111
170 LET x=IN db
180 REM lees de Y coördinaat
190 OUT da,BIN 00000000
200 OUT da,BIN 00010111
210 LET y=IN db*175/255
220 REM t=tekenen s=stop tekenen n=nieuwe tekening
230 IF INKEY$="t" THEN LET a$="t"
240 IF INKEY$="s" THEN LET a$="s"
250 IF INKEY$="n" THEN GO TO 100
260 IF a$="t" THEN GO TO 350
270 REM flitsend tekenpunt
280 PLOT OVER 1;x,y
290 REM algemene vertraging
300 PAUSE 10
310 PLOT OVER 1;x,y
320 LET x1=x
330 LET y1=y
340 GO TO 140
350 REM tekenen
360 PLOT x1,y1
370 DRAW x-x1,y-y1
380 GO TO 320

```


Appendix A

Verklarende woordenlijst

Adresbus – 16 bij elkaar horende verbindingen die de CPU in staat stellen de plaats te kiezen waar hij een bepaalde bewerking wil uitvoeren.

Analoog naar digitaal omzetter (ADC) – een schakeling welke een analoge spanning omzet in een digitaal getal dat door een computer gelezen kan worden.

Asynchroon – wanneer twee apparaten onafhankelijk van elkaar werken, dan zegt men dat ze asynchroon werken.

Bidirectioneel – in staat zijn zowel informatie te ontvangen als informatie te kunnen verzenden.

Binair – een getallenstelsel gebaseerd op het grondtal twee in plaats van op grondtal tien zoals in het decimale systeem.

Chip – genoemd naar het kleine plakje silicium waarop alle schakelingen van de computer geëtst zijn. Deze kleine siliciumschakeling is vaak in een zwart blokje plastic verpakt met twee rijen metalen pennen om het aan te sluiten.

CPU – Centrale verwerkingseenheid. De Z80A in de Spectrum. De CPU bestaat uit een besturingseenheid en een rekeneenheid.

Databus – 8 bij elkaar horende verbindingen waarover alle gegevens, die de verschillende onderdelen van de Spectrum nodig hebben, verzonden worden.

Geheugen – Het onderdeel waarin alle gegevens over BASIC en programma's worden opgeslagen.

Geheugenbit – Kleinste eenheid in het geheugen. Elk bit kan 0 of 1 zijn.

Geheugenbyte – 8 geheugenbits. De CPU leest en schrijft gegevens met 8 bits = 1 byte tegelijk via de databus.

Hex – afkorting van hexadecimaal. Dit getalstelsel is gebaseerd op grondtal 16 in plaats van op grondtal 10 zoals bij het decimale stelsel. De tekens '0 1 2 3 4 5 6 7 8 9 A B C D E F' worden gebruikt om 0 t/m 15 weer te geven. Het getal 1F hex komt overeen met $1 \cdot 16 + 15 = 31$ decimaal. Hexadecimaal is handig omdat 0 t/m 255 is 0 t/m FF hex, dus zijn er maar twee tekens nodig om alle mogelijke getallen, die over de databus kunnen, weer te geven.

Hoog – soms gebruikt om het niveau logisch 1 aan te geven.

Koelplaat – ontworpen om warmte van kleine onderdelen af te voeren. Het stuk aluminium in de Spectrum voert de warmte af van de +5 volt regelaar om hem koel te houden.

Kristal – een klein kwarts-kristal dat nauwkeurig op maat gesneden is, zodat het meetrikt op een vaste frequentie. Het wordt gebruikt om de frequentie waarop de klok loopt te stabiliseren.

Laag – soms gebruikt om het niveau logisch 0 aan te geven.

Laagactief – signalen die laagactief zijn, worden geacht werkzaam te zijn wanneer ze op een logisch 0-niveau zijn.

Lichtuitzendende diode (LED) – een onderdeel dat de stroom maar in één richting doorlaat. Licht wordt uitgezonden als het onderdeel geleidt.

Machinecode – een binair programma dat direct door de CPU begrepen kan worden.

1 MHz – een frequentie van één miljoen trillingen per seconde.

Omzetter – een apparaat dat een natuurkundige hoeveelheid zoals snelheid, luchtdruk of temperatuur omzet in een elektrisch signaal dat door een computer verwerkt kan worden.

Onderbreking – een signaal, gegeven door een randapparaat, om de CPU te onderbreken in zijn taak en hem wat anders te laten doen.

Opfrissen – een bewerking die regelmatig moet worden uitgevoerd op bepaalde soorten geheugens om er voor te zorgen dat ze gegevens vast blijven houden.

Op hol slaan – wat gebeurt als de CPU een programma uitvoert dat onzin bevat. In het algemeen bij het op hol slaan raakt de computer zodanig in de war dat alleen uitzetten en opnieuw beginnen uitkomst biedt.

RAM – willekeurig toegankelijk geheugen waarin geschreven of waaruit gelezen kan worden door de CPU op elk adres. Er worden twee soorten veel gebruikt, statische die niet opgefrist hoeven te worden en dynamische (zoals in de Spectrum) die dit wel moeten.

Randapparaat – een apparaat verbonden met de CPU. Het toetsenbord en de cassette recorder zijn beide randapparaten.

ROM – alleen te lezen geheugen, zoals de naam aangeeft een geheugen waaruit alleen gelezen kan worden en niet in kan worden geschreven. Het BASIC-systeem van de Spectrum is opgeborgen in dit soort geheugen.

Status – een invoer of uitvoer is normaal in één van de twee toestanden 0 of 1 (zie ook tristate).

Tristate – Engels voor driestanden, soms kunnen verschillende uitgangen van verschillende chips met elkaar verbonden worden. Om de datalijnen niet met elkaar in conflict te laten komen, de ene uitgang kan logisch 1 zijn terwijl de andere logisch 0 is waardoor er sluiting via de chips optreedt, worden alle uitgangen

op één na in de tristate-stand gebracht (elektrisch gezien: niet aanwezig). Een tristate-uitgang kan logisch 0 of 1 zijn of 'niet aanwezig'.

ULA – niet gebonden logisch netwerk. Een in massa geproduceerd onderdeel dat bepaalde functies verricht die anders door losse onderdelen gedaan zouden worden.

Zenerdiode – Wordt gebruikt om de spanning over de zenerdiode te stabiliseren op een door de diodekarakteristiek gedefinieerd niveau.

Appendix B

Referenties

1. Sinclair ZX Spectrum BASIC programming manual, Sinclair Research, 1982.
2. Sinclair ZX Spectrum Introduction booklet, Sinclair Research, 1982.
3. Oxford Illustrated Dictionary, Oxford Univ. Press, 1962.
4. Mostek Z80 en PIO databooks.
5. Mostek 1980 memory databook and designers guide.
6. Texas Instruments, TTL Databook, 1980.
7. Ferranti Semiconductors, Quick Reference Guide, 1981.
8. Zilog Microcomputer Components Databook, 1980.
9. National Semiconductor, Linear Databook, 1980.
10. National Semiconductor, CMOS Databook, 1981.
11. Ferranti Electronics Ltd., Data Converter Handbook, 1980.
12. J.C. Nichols, E.A. Nichols en P.R. Rony, Z80 Microprocessor Programming & Interfacing Book 2, 1981.

Appendix C

Onderdelenlijst van de Spectrum (model 2)

Weerstand

R1	470	R34	15R
R2	470R	R35	10K
R3	470R	R36	680R
R4	470R	R37	1K
R5	470R	R38	3K3
R6	470R	R39	3K3
R7	470R	R40	1K
R8	470R	R41	1K5
R9	8K2	R42	1K
R10	8K2	R43	3K
R11	8K2	R44	5K1
R12	8K2	R45	1K
R13	8K2	R46	1K
R14	8K2	R47	220R
R15	8K2	R48	4K7
R16	8K2	R49	18K
R17	330R	R50	4K7
R18	330R	R51	2K2
R19	330R	R52	2K2
R20	330R	R53	390R
R21	330R	R54	100K
R22	330R	R55	56R
R23	330R	R56	220R
R24	3K3	R57	330R
R25	180R	R58	1K
R26	680R	R59	1K8
R27	680R	R60	100R
R28	10K	R61	15R
R29	1K5	R62	15R
R30	1K	R63	220R
R31	220K	R64	15R
R32	100R		
R33	680R		

Condensatoren

C1 47nF
C2 47nF
C3 47nF
C4 47nF
C5 47nF
C6 47nF
C7 47nF
C8 47nF
C9 t/m C24 niet aanw.
C25 22µF 16V
C26 47nF
C27 1µF 63V
C28 22µF 16V
C29 47nF
C30 47nF
C31 100nF
C32 100nF

C33 47nF
C34 22µF 16V
C35 10nF
C36 47nF
C37 33pF
C38 33pF
C39 10nF
C40 47nF
C41 47nF
C42 47nF
C43 100nF
C44 100µF 16V
C45 100µF 16V
C46 1µF 50v
C47 22µF 16V
C48 47nF
C49 47nF

Transistoren

TR1 ZTX 313
TR2 ZTX 313
TR3 ZTX 313

TR4 ZTX 651
TR5 ZTX 213

Diversen

7805 1 amp. 5 volt regelaar
X1 14 MHz kristal
X2 4,4336 MHz kleurenkristal
VR1 2K2 horizontale afregeling
VR2 2K2 horizontale afregeling

Diodes

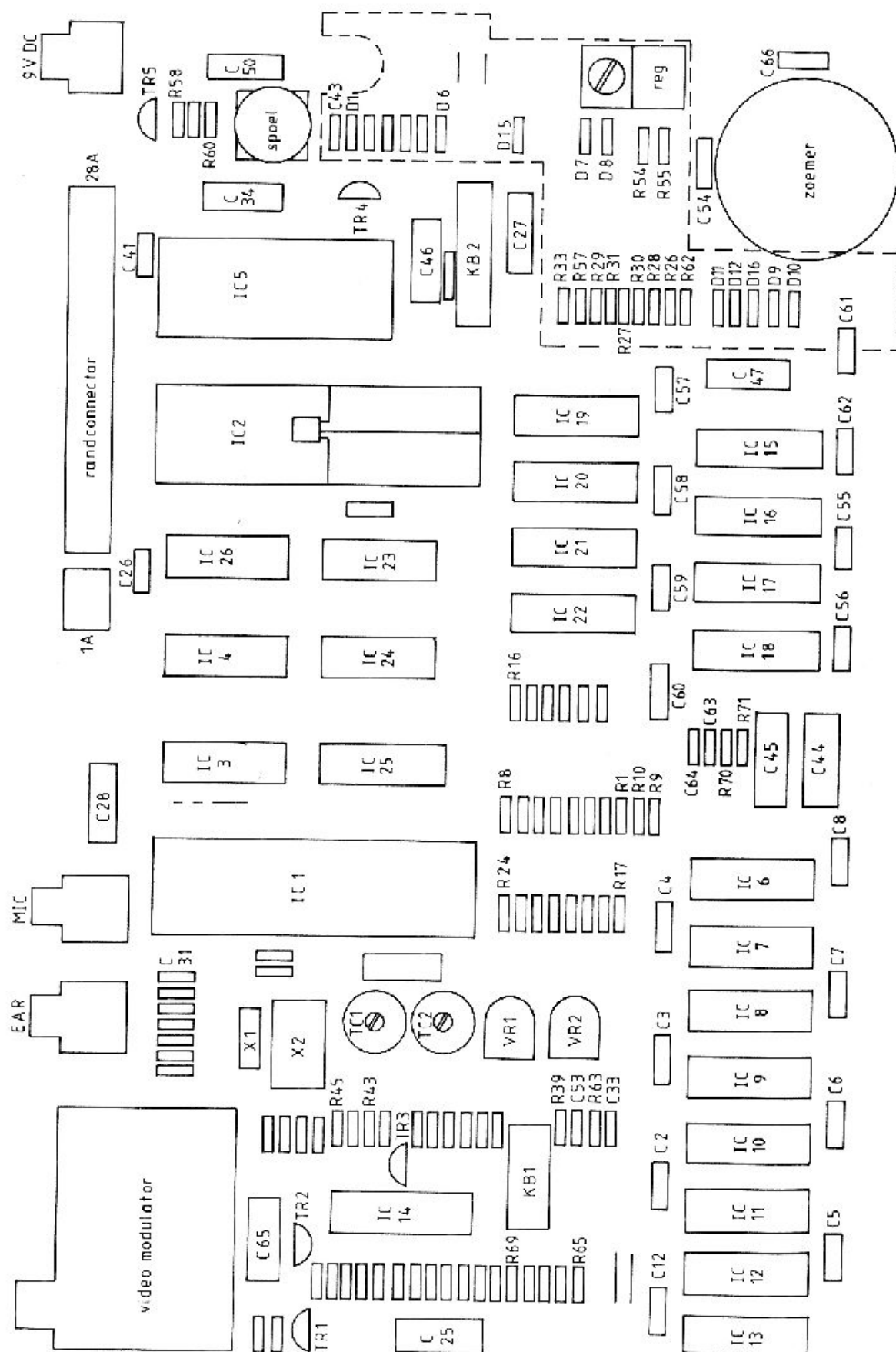
D1 t/m D14 IN4148
D15 vermogensdiode
D16 zenerdiode

IC's

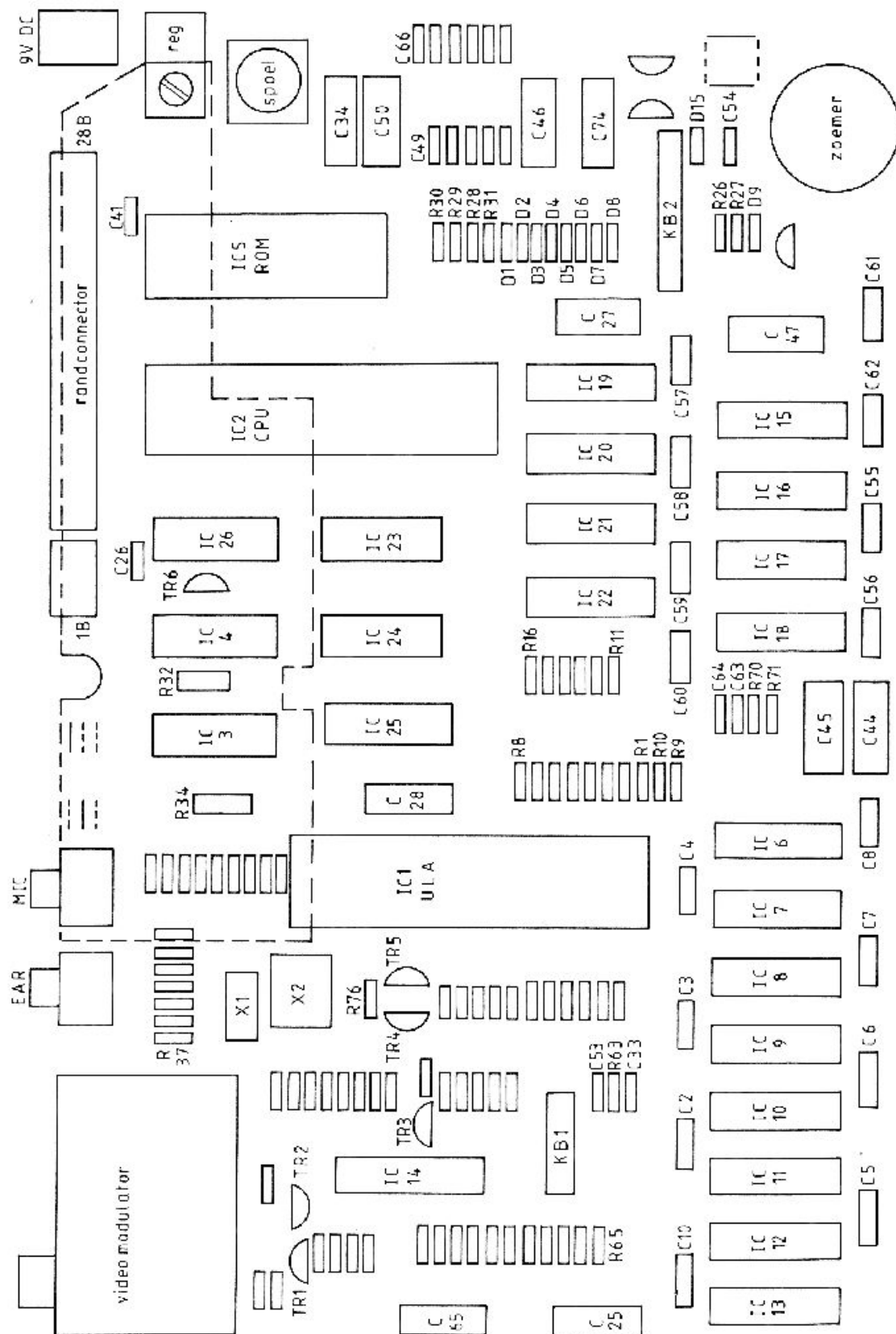
IC1 Sinclair ULA
IC2 Z80A CPU
IC3 74LS157
IC4 74LS157
IC5 16K ROM
IC6 LM1889N
IC7 t/m IC14 16K*1 dynamische RAM
IC15 74LS00

Appendix D

Onderdelenopstelling Model 2

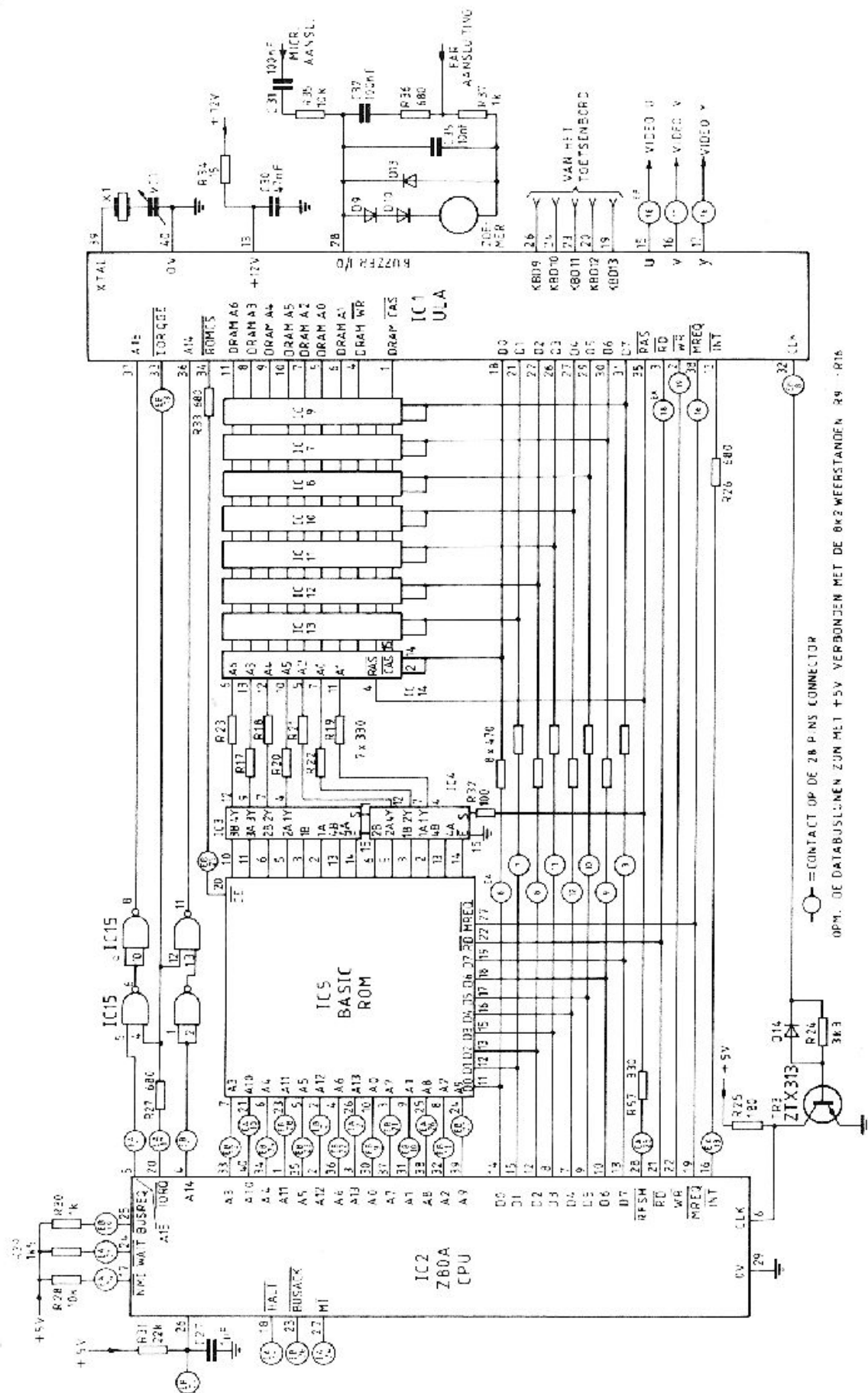


Onderdelenopstelling Model 3



Appendix E

Volledig schema



Index

A

aanwijzer 43
ADC 87
adresbus 14, 26
analoog-multiplexer-chips . . . 92
analoog-naar-digitaal omzetter . 87

B

beeldterugslagtijd 43
besturingssysteem 14
binair. 11

C

carry-bit 11
CAS-signalen 29, 46
centrale-verwarmingsprogramma 71
centrale verwerkingseenheid . . 11
controlebus 14
CPU 11, 24

D

databus 11, 26
D-connector 87
DRAM 29

E

EPROM. 32
experimenteerbord 64

F

fouten 61

G

geheugenmatrix 29
grijsschaal 52

H

helderheidssignaal 50
herstel-schakelaar 65

I

interface 9
I-register 43

J

joystick. 81
juiste adressen 29

K

kleurenkristal 62
kleurverschilsignalen 50

L

LED's	73
lichtuitzende dioden	73
LM1889N	50

M

model 2	9
model 3	9
moederklok	44
multiplexer	29
multiplexing	29

O

ontkoppelcondensator	22
open-collectoruitgang	68
operating system	32
opfrissing	29

P

PIO	71
power supply	16

R

RAM	14
randconnector	56
randkleur	45
RAS-signalen	29, 46

7805-regelaar	16
RGB-video	50
ROM	14, 32

S

statische RAM	29
stroombegrenzings	19
stroomterugkoppeling	19
stuurknuppel	81

T

toetsenbord-interface	76
---------------------------------	----

U

ULA	14, 38
Uncommitted Logic Array	38

V

verostripbord	64
videomodulator	50
videosynchronisatiesignaal	50
voeding	16

Z

Z80A	24
Z80A-PIO-chip	71

De ZX Spectrum wordt geleverd met een introductieboek en een BASIC-programmeerhandleiding. Deze boeken geven nauwelijks informatie over de hardware-aspecten van de computer.

In dit boek wordt de Spectrum van binnen bekeken. Voor de beginner die nog geheel onbekend is met de hardware, geeft dit een goed beeld op welke wijze de computer de informatie verwerkt. Voor de doorgewinterde elektronicus die er op uit is om zijn Spectrum voor allerlei toepassingen in te zetten, geeft dit boek eveneens de gewenste informatie.

Beginnend met de principes achter de Spectrum-bewerkingen geeft het boek een volledige beschrijving van de werking van de verschillende componenten en de onderlinge samenhang ervan. Nadat alle signalen aan de connector zijn bekeken, laten enkele experimenten de gebruiksmogelijkheden hiervan zien.

Het resterende deel van het boek wordt in beslag genomen door een aantal praktische schakelingen voor zelfbouw.

De beschrijvingen en de afbeeldingen maken het ook voor degene zonder ervaring mogelijk deze schakelingen te bouwen.

Ten slotte wordt in de appendices een volledig prinsipschema met print-lay out en onderdelenlijst van de Spectrum gegeven.

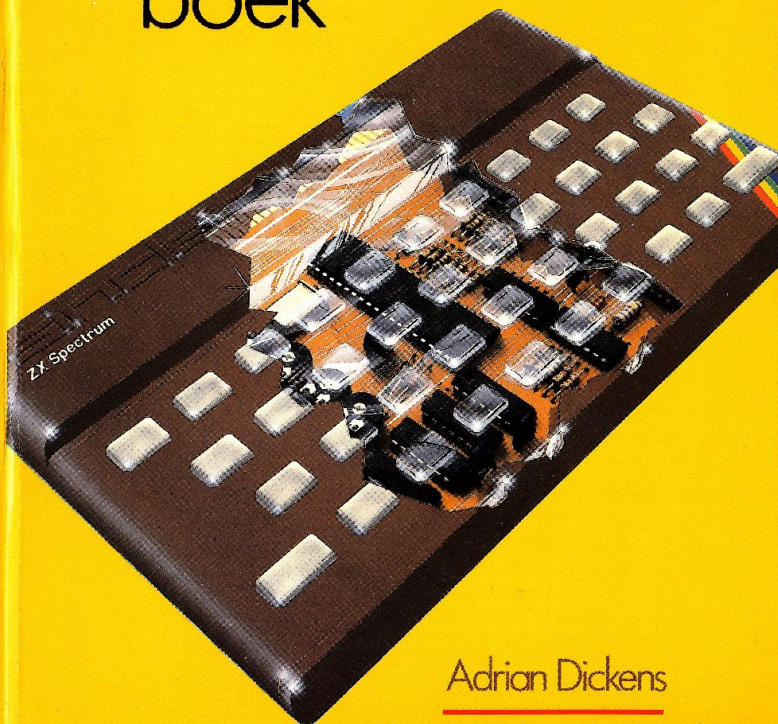
ISBN 90 201 1762 9

Adrian Dickens

ZX SPECTRUM HARDWARE-BOEK



ZX Spectrum HARDWARE boek



Adrian Dickens
Kluwer technische boeken