

De TV-monoselector, een hulpmiddel voor motorisch gehandicapten

Citation for published version (APA):

Smolders, J. P. C., Aarts, A. J. H., Leliveld, W. H., van Uitert, A. G., Mathijssen, R. W. M., & Ossevoort, H. J. M. (1990). *De TV-monoselector, een hulpmiddel voor motorisch gehandicapten*. Technische Universiteit Eindhoven.

Document status and date:

Gepubliceerd: 01/06/1990

Document Version:

Uitgevers PDF, ook bekend als Version of Record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

SAMENVATTING

De TV-monoselector (TVM) is een programmeerbare, éénknops infrarood afstandsbediening ten behoeve van motorisch gehandicapten.

Men kan de TVM voor alle bestaande infrarood afstandsbedieningen programmeren (TV, audio, video) daar de TVM daartoe voorzien is van een infrarood ontvanger. De TVM ontvangt tijdens die programmeer cyclus de signalen van een bestaande afstandsbediening, waarna deze gecodeerd opgeslagen worden. Het programmeren is eenvoudig en kan geschieden door iedereen die de reeds bestaande standaard afstandsbediening kan bedienen. Met behulp van een menu kunnen functies geselecteerd worden die op een LED-matrix worden aangegeven. Iedere LED van de matrix heeft een eigen functie welke geselecteerd kan worden als de LED gaat branden door middel van een druk op de knop. De TVM kan de opgeslagen infrarood signalen verzenden en kan hierdoor als afstandsbediening functioneren. Bovendien blijft het mogelijk de TV ook te bedienen met de daarbij behorende afstandsbediening.

De TVM wordt gevoed door een 9 Volt (oplaadbare) batterij. Er is een testschakeling van de TVM gebouwd welke naar behoren werkt, maar nog niet met behulp van de LED-matrix bediend kan worden. Software hiervoor is nog in ontwikkeling.

TV-monoselector, een hulpmiddel voor motorisch gehandicapten

| <u>Inhoudsopgave</u> | Blz. |
|--|------|
| 1 Inleiding..... | 5 |
| 2 Vooronderzoek..... | 6 |
| 3 Infrarood signalen..... | 7 |
| 3.1 Inleiding..... | 7 |
| 3.2 Opbouw..... | 7 |
| 3.3 Registratie..... | 8 |
| 3.4 Representatie..... | 9 |
| 3.5 Opslag..... | 10 |
| 3.6 Toggle-bit..... | 12 |
| 4 Het ontwerp..... | 15 |
| 4.1 De werking van de TVM..... | 15 |
| 4.2 Bediening van de TVM..... | 16 |
| 4.3 De hardware..... | 17 |
| 4.3.1 Voeding en reset..... | 18 |
| 4.3.2 Infrarood ontvanger en zender..... | 20 |
| 4.3.3 LED-matrix en status LED's..... | 23 |
| 4.3.4 De microprocessor..... | 24 |
| 5 De programmatuur..... | 26 |
| 5.1 Inleiding..... | 26 |
| 5.2 Taken van de programmatuur..... | 26 |
| 5.3 Werking van het programma..... | 27 |
| 5.3.1 Kiezen voor programmeer- of gebruikers- gedeelte..... | 27 |
| 5.3.2 De functievorgordetabel..... | 27 |
| 5.3.3 Kiezen voor aanpassen functievorgorde- tabel..... | 28 |
| 5.3.4 Aanpassen functievorgordetabel..... | 28 |
| 5.3.5 Het selecteren van een functie..... | 28 |
| 5.3.6 Het IR-sigitaal inladen, coderen en opslaan..... | 28 |
| 5.3.7 Het IR-sigitaal genereren..... | 28 |
| 6 Conclusies..... | 29 |
| Literatuurlijst..... | 30 |

Bijlagen

| | | |
|---|---|----|
| A | Ontwikkelingsverloop TVM..... | 31 |
| B | Schematische werking van het programma. | 32 |
| C | Schema van de TVM..... | 33 |
| D | Ontkoppelingen over de voedingen..... | 34 |
| E | Lay-out van de print..... | 35 |
| F | Stuklijst..... | 36 |

HOOFDSTUK 1 INLEIDING

De vakgroep Medische Electrotechniek (EME) van de Technische Universiteit te Eindhoven houdt zich 'o.a. bezig met instrumentatie t.b.v. gehandicapten.

Een aantal jaren geleden is t.b.v. zwaar motorisch gehandicapten de monoselector ontwikkeld. Dit is een apparaat waarmee men met slechts een knop een groot aantal functies kan bedienen, zoals TV aan- en uitzetten en gordijnen sluiten en openen.

Mede naar aanleiding van verzoeken om een monoselector te ontwerpen waarmee meer functies als uitsluitend aan en uit van een bepaald type TV bediend kunnen worden, is het idee ontstaan om een universele, éénknops afstands bediening te ontwerpen. Dit is de T.V.-monoselector (TVM). Deze zou met behulp van een menu een aantal functies moeten kunnen selecteren, die op een LED-display worden aangegeven, en uitvoeren door de daarbij behorende infrarood-codes van bestaande afstandsbedieningen te genereren.

Het werk is gedaan in opdracht van de TU-Eindhoven en uitgevoerd als afstudeeropdracht in het kader van de opleiding aan de HTS-Breda.

De omschrijving van de opdracht luidt als volgt :

Verdere ontwikkeling en realisering van de hard- en software van de TVM, waarbij de nadruk ligt op :

- Bestudering van de voorafgaande ontwikkelingen betreffende de TVM.
- Het inlezen en opslaan van infrarood-signalen van bestaande afstandsbedieningen.
- Het uitzenden van deze in een geheugen opgeslagen signalen.
- Weergeven van het keuzemenu met behulp van een LED-matrix.
- De ontwikkeling van de benodigde software.

HOOFDSTUK 2 VOORONDERZOEK

Reeds zeven mensen hebben eerder aan 'de ontwikkeling van de TVM gewerkt. Dit resulteerde in eerste instantie in een werkend prototype met als nadeel dat deze via een 20-aderige kabel op een reeds bestaande afstandsbediening moest worden aangesloten (lit. 1 t/m 5). Om deze reden werd later een TVM ontworpen met een eigen infrarood zender/ontvanger die alle bestaande infrarood afstandsbedieningen zou moeten kunnen simuleren (lit. 6 en 7).

Uit voorgaand onderzoek zijn een aantal randvoorwaarden ontstaan voor de TVM :

- de TVM moet iedere bestaande infrarood afstandsbediening kunnen simuleren (audio, video etc.).
- het programmeren van de TVM moet op eenvoudige wijze kunnen geschieden door iedereen die een standaard afstandsbediening kan bedienen.
- de TVM moet voorzien zijn van een batterijvoeding. Dit omdat het apparaat dan bijvoorbeeld eenvoudig op een rolstoel gemonteerd kan worden. Deze keuze impliceert dat het stroomverbruik laag moet zijn.
- er dienen geen aanpassingen nodig te zijn van de TV en van de eigen afstandsbediening. De televisie en de daarbij behorende afstandsbediening dienen bruikbaar te blijven voor de huisgenoten van de gehandicapte.
- bij het wegvallen van de spanning moet alle in de TVM opgeslagen data behouden blijven.

R. Bloks (lit. 6) ontwierp een schakeling, die als basis diende voor ons ontwerp. Ook gaf hij de eerste aanzet voor een programma dat door M. Muris (lit. 7) verder uitgewerkt en geschreven is. Dit leverde nog geen werkende TVM op. Een aantal problemen die hij tegenkwam was:

- te weinig geheugenruimte in het EEPROM en het RAM.
- te hoog stroomverbruik.
- het inlezen en opslaan van de verschillende IR-signalen.

HOOFDSTUK 3 INFRAROOD-SIGNALLEN

3.1 Inleiding

De TVM moet de werking van een reeds bestaande infrarood afstandsbediening kunnen simuleren. Deze eis maakt een studie van de opbouw van de infrarood signalen noodzakelijk. Bovendien moet er een methode ontwikkeld worden om deze signalen eenvoudig te detecteren en vervolgens te registreren. We maken hierbij gebruik van een infrarood ontvanger om meting van de pulsbreedte en periodetijden van de signalen mogelijk te maken. Iedere periodetijd krijgt een code met daarachter het aantal maal dat deze periode achtereenvolgens voorkomt. Deze gegevens worden tijdelijk in een RAM opgeslaan en vormen samen het tijdcodenummer. Vanwege het feit dat een aantal perioden gevolgd wordt door een rust kan deze steeds terugkerende combinatie op zich een eigen code krijgen en wel een PCP-codenummer. PCP- en tijdcodenummers worden in een EEPROM opgeslaan.

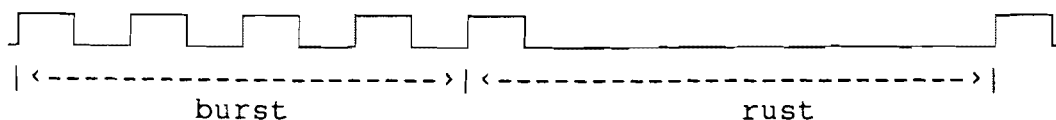
Tot slot wordt een infraroodsignaal gerepresenteerd door de opeenvolging van een aantal PCP-codes. Per signaal wordt de combinatie van PCP-codenummers in het EEPROM gezet.

Het geheel wordt bestuurd door een microprocessor.

In dit hoofdstuk worden de opbouw, registratie, representatie en opslag van de infrarood signalen besproken.

3.2 Opbouw

Door Rudi Bloks (lit. 6) is een overzicht gemaakt van de infrarood signalen van diverse afstandsbedieningen. Bijna alle zenders werken met bifase-modulatie. Hierbij wordt een pulstrein van een aantal perioden (burst) gevolgd door een rust (zie figuur 3.1).



figuur 3.1 Golfvorm.

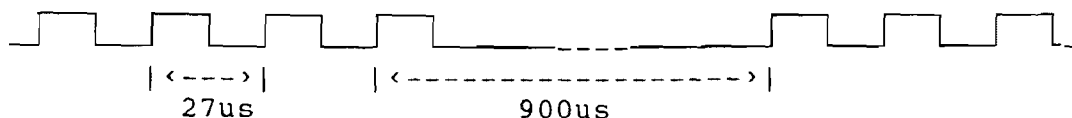
De signalen hebben de volgende eigenschappen:

- in de optredende bursts is de kleinst voorkomende pulsbreedte ongeveer 8 us.
- de afstand tussen twee opeenvolgende pulsen bedraagt minimaal 25 us.
- de pulsbreedte is constant
- een burst is opgebouwd uit een aantal perioden waarvan de laatste een lange laagtijd heeft

3.3 Registratie

Om het mogelijk te maken dat de TVM de infrarood signalen van een afstandsbediening kan simuleren moet de TVM van ieder afzonderlijk type de infrarood signalen als het ware kunnen "inlezen". Dit inlezen wordt mogelijk gemaakt door een infrarood ontvanger, welke in hoofdstuk 4 besproken wordt. Het infrarood signaal moet na ontvangst een bewerking ondergaan waarbij dit signaal geregistreerd wordt.

Deze registratie werkt als volgt. Van het signaal worden alle periodetijden gemeten en slechts één pulsbreedte omdat deze constant is. Een periodetijd is de tijd tussen de stijgende flanken van twee opeenvolgende pulsen. Op deze manier is ook een rust in het signaal te detecteren. Ter verduidelijking staat in figuur 3.2 een gedeelte van een infrarood signaal getekend met de betreffende tijdwaarden waarmee dit signaal vastgelegd wordt.



figuur 3.2 Registratie van IR signaal.

In dit voorbeeld is de periodetijd 27 us. Het meten van de periodetijden wordt gerealiseerd met een acht bits timer, die telt in stappen van 1 us. Dit houdt in dat periodetijden van maximaal 256 us. gemeten kunnen worden.

De rust periode van het signaal tussen twee elkaar opvolgende bursts is over het algemeen langer dan 256 us. De timer genereert een overflow als de tijdwaarde groter wordt als 256 us. Bij iedere overflow zal in het geheugen de tijd 00h worden geregistreerd. Dit komt dus overeen met een tijd van 256 us. aangezien de tijdwaarde 0 normaal niet voorkomt.

In figuur 3.2 is de rustperiode 900 us. lang, wat overeenkomt met drie maal een overflow ($3 \times 256 = 768$ us.) en een rest van 132 us. Deze rustperiode komt overeen met de waarde 00, 00, 00, 84 hexadecimaal.

De periodetijd van 27 us wordt voorgesteld door de waarde 1Bh. In totaal kan dit signaal dus als volgt voorgesteld worden:

1Bh, 1Bh, 1Bh, 00h, 00h, 00h, 84h, 1Bh, 1Bh, 1Bh ...

3.4 Representatie

Om een representatie te maken van de infrarood signalen wordt gebruik gemaakt van de geregistreeerde periodetijden zoals beschreven in paragraaf 3.3. Iedere periodetijd krijgt een codenummer en achter ieder codenummer komt een getal te staan dat het aantal aaneengesloten herhalingen van deze periode aangeeft.

De codering van het infrarood signaal uit figuur 3.2 levert:

```

01h    04h    , 02h    01h
|      |      |      |
code   aantal x, code   aantal x

```

Hierbij krijgt de periodetijd 1Bh het codenummer 01h en de periodetijd 00h, 00h, 00h, 84h krijgt het codenummer 02h. De op deze manier verkregen code kan men dus als volgt lezen:

```

tijdcodenummer 01h komt 4 x opeenvolgend voor
tijdcodenummer 02h komt 1 x opeenvolgend voor

```

Tot slot wordt ook aan de tijdcodenummers die bij elkaar horen en dus regelmatig voor kunnen komen, een codenummer toegekend. In het voorbeeld van figuur 3.2 wordt het signaal in eerste instantie gecodeerd met behulp van tijdcodenummers en levert zoals gezien:

```

    01h, 04h, 02h, 01h
|<----->|<----->|
  burst      rust

```

We noemen deze combinatie van een burst en een rust het Puls Code Paar (PCP) en kunnen hieraan het PCP-codenummer 01h toekennen. Op deze manier wordt een compleet infrarood commando gecodeerd in PCP-codenummers, waardoor effectieve opslag in een geheugen mogelijk wordt gemaakt. Het geheugen is in dit geval een EEPROM (zie paragraaf 4.1).

De pulsbreedte is voor iedere puls hetzelfde en hoeft daarom slechts één maal geregistreerd en gecodeerd te worden.

3.5 Opslag

Bij de codering van een infrarood signaal worden twee codenummers ingevoerd, te weten:

- tijdcodenummer
- PCP-codenummer

Voor de tijdcodenummers zijn twee bytes nodig:

- higher byte = periodetijd
- lower byte = aantal opeenvolgende perioden

Deze worden opgeslagen in een tijdcodetabel zoals te zien is in figuur 3.3.

In deze tabel komt het codenummer overeen met het volgnummer. De plaats van het higher en lower byte is in principe vrij te kiezen, evenals het beginadres van de tabel.

| tijdcodenummer | byte | adres |
|----------------|------|-------|
| 01 | high | 8000 |
| | low | |
| 02 | high | 8002 |
| | low | |
| 03 | high | 8004 |
| | low | |

figuur 3.3 Tijdcodetabel.

Doordat ieder codenummer opgebouwd is uit twee bytes zijn voor iedere tijdcode twee adressen nodig. Wanneer nu het tijdcodenummer en het beginadres van de tijdcodetabel bekend zijn dan kan het adres van een tijdcode berekend worden. Het PCP-codenummer wordt opgeslagen in een PCP-tabel, zoals te zien is in figuur 3.4.

Een PCP bestaat uit vier bytes:

- 1 tijdcodenummer #1
- 2 herhaling #1
- 3 tijdcodenummer #2
- 4 herhaling #2

Bij het voorbeeld van figuur 3.2 stellen de bytes 1 en 2 een burst voor en de bytes 3 en 4 een rust.

De PCP-tabel ziet er als volgt uit:

| PCP-codenummer | byte | adres |
|----------------|--|-------|
| 01 | tijdcode #1 herhaling #1 tijdcode #2 herhaling #2 | 8500 |
| 02 | tijdcode #1 herhaling #1 tijdcode #2 herhaling #2 | 8504 |

etc.

figuur 3.4 PCP-tabel.

Omdat het PCP-codenummer overeenkomt met het volgnummer kan ook hier het adres van een bepaalde PCP-code berekend worden, mits het begin adres van de PCP-tabel bekend is.

Uiteindelijk moet een volledig infrarood commando opgeslagen worden. Elk commando is als het ware een opeenvolging van een aantal PCP-codes. Deze worden opgeslagen in een infrarood-tabel zoals te zien is in het voorbeeld van figuur 3.5.

| infrarood code | byte | adres |
|----------------|---|------------------|
| | pulsbreedte | |
| 01 | lengte + 1 PCP #1 PCP #2 PCP #n | 9000 |
| 02 | lengte + 1 PCP #1 PCP #2 PCP #m | 9000 + n + 1 |
| 03 | lengte + 1 PCP #1 PCP #2 | 9000 + n + m + 2 |

etc.

figuur 3.5 Infrarood-tabel.

Wanneer een commando gegeven wordt kan de bijbehorende infrarood-code in de infrarood-tabel opgezocht worden. De PCP-codenummers worden op hun beurt weer opgezocht in de PCP-tabel en tot slot worden de eigenlijke periodetijden uit de tijdcodetabel gehaald.

Voor het uitlezen van de PCP-tabel is het noodzakelijk te weten hoeveel PCP-codenummers bij één infrarood commando behoren, zodat het juiste aantal PCP-codes uitgelezen worden. Het aantal PCP-codes per commando is niet constant. Om te voorkomen dat er fouten optreden bij het uitlezen van deze tabel wordt bij ieder commando de lengte vermeld.

Bovendien moet de pulsbreedte geregistreerd worden. Er wordt maar met één pulsbreedte gewerkt waardoor het voldoende is de pulsbreedte één keer op te slaan.

Samenvattend kunnen we zeggen dat de opslag van een volledig infrarood commando plaats vindt in de volgende drie tabellen:

- 1 Tijdcodetabel : bevat de periodetijd van een puls.
- 2 PCP-tabel : bevat informatie van een burst + rust, wat wordt voorgesteld door een Puls Code Paar.
- 3 Infrarood-tabel: bevat het aantal PCP-codes per commando. Bovendien staat hierin de pulsduur en het aantal adressen dat een commando in de tabel inneemt.

3.6 Toggle-bit

Enkele afstandsbedieningen, bijvoorbeeld Philips, maken gebruik van een zogenaamd "toggle-bit". Dit is een bit dat van waarde verandert bij ieder nieuw commando. Op deze manier wordt een scheiding verkregen tussen elk volgend commando, wat bijvoorbeeld nodig is als men teletekst gebruikt. Bij het selecteren van een teletekst pagina moet men meerdere toetsen indrukken. Wanneer bijvoorbeeld pagina 444 wordt geselecteerd, dan kan de TV ontvanger het drie maal indrukken van een vier zien als drie maal hetzelfde commando. Om ervoor te zorgen dat de TV ontvanger iedere vier als een afzonderlijk commando ziet wordt een toggle-bit toegepast.

Er wordt hier gesproken van een bit omdat een burst gevolgd door een rust opgevat kan worden als een logische "0" en een rust gevolgd door een burst als logische "1".

In figuur 3.6 zijn beide mogelijkheden getekend.

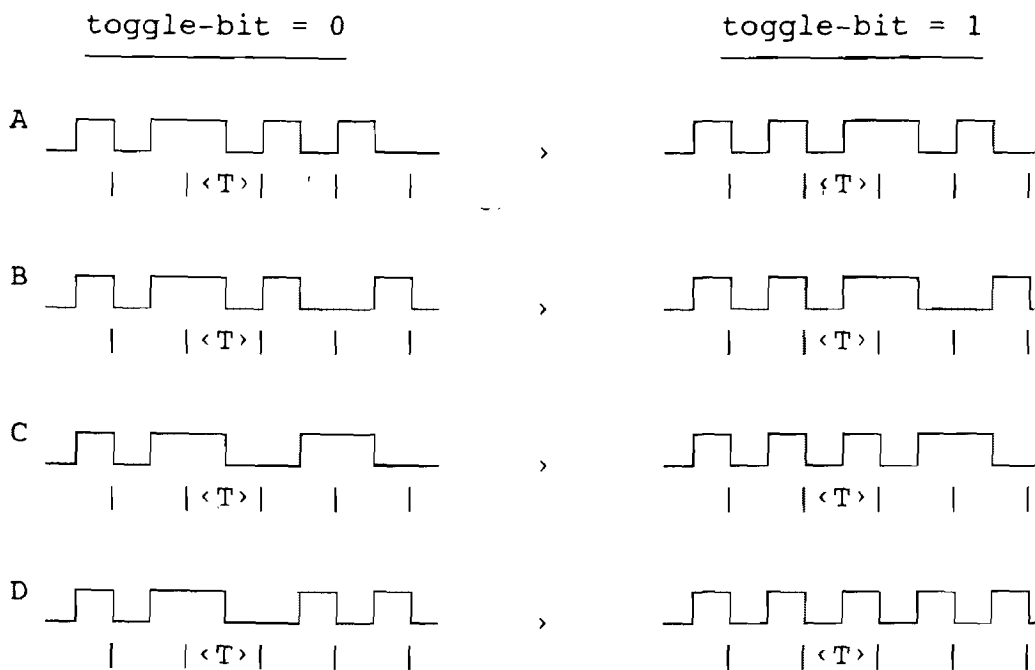


figuur 3.6 Golfvorm van logisch "0" en logisch "1"

In figuur 3.7 zijn de mogelijke signaalvormen ten gevolge van een toggle-bit aangegeven. Eén puls in de figuur stelt een burst gevolgd door een rust voor. Ter verduidelijking een tekening van de manier waarop een logisch "0" in figuur 3.7 wordt weergegeven:



De signalen waarbij het toggle-bit "0" respectievelijk "1" is staan in figuur 3.7 naast elkaar. Ieder bit wordt gemarkeerd door twee verticale lijnen onder het signaal. Hierbij dient opgemerkt te worden dat bij de registratie van de signalen niet de bits maar periodetijden tussen twee opgaande flanken gemeten worden.



figuur 3.7 Signalen bij toggle-bits.

Een commando begint met twee startbits die altijd "1" zijn, gevolgd door het toggle-bit. Na dit toggle-bit komen de resterende bits van het commando. De gebieden aangeduid met <T> geven aan waar de golfvormen voor de twee waarden van het toggle-bit van elkaar verschillen.

In de codering worden de toggle-bits zonder meer meegenomen. Het blijft echter noodzakelijk het verschil in een bepaald commando als gevolg van het toggle-bit te registreren. Dit wordt softwarematig ondervangen door het signaal van één commando meerdere malen in te lezen, te registreren in het RAM en vervolgens onderling te vergelijken.

Uiteindelijk worden twee signalen voor één commando in het EEPROM opgeslagen. Door deze signalen beurtelings uit te zenden blijft de functie van het toggle-bit gehandhaafd.

Wanneer een afstandsbediening niet met een toggle-bit werkt dan is het niet nodig om twee signalen op te slaan. Om de TVM universeel te houden wordt dit echter wel gedaan, zodat men de TVM voor iedere willekeurige afstandsbediening kan programmeren en men dus niet hoeft te weten of de afstandsbediening nu wel of niet met een toggle-bit werkt.

In hoofdstuk 4 wordt verder ingegaan op het ontvangen, opslaan en verzenden van de infrarood signalen.

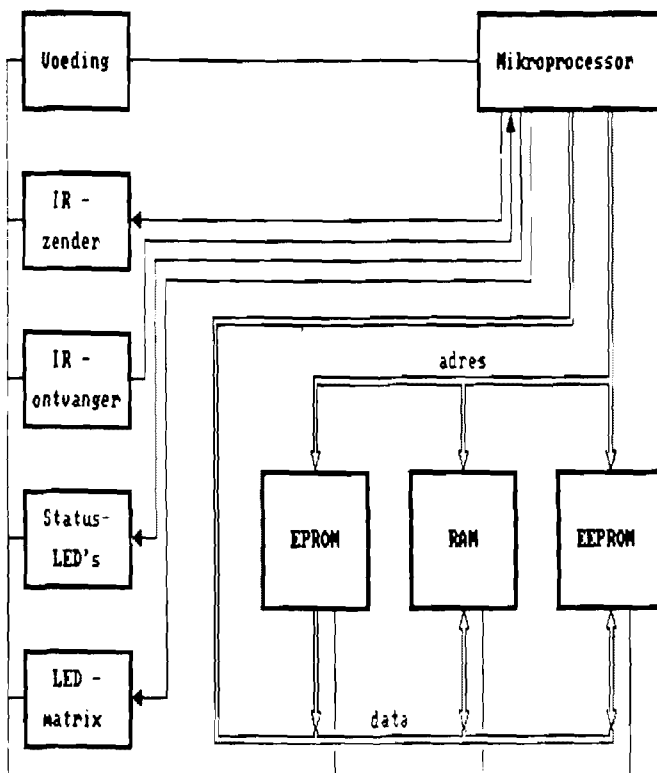
HOOFDSTUK 4 HET ONTWERP

4.1 De werking van de TVM

De TVM moet elke infrarood afstandsbediening kunnen simuleren. Hiervoor is het noodzakelijk dat de TVM de infrarood signalen van elke afstandsbediening kan genereren.

De meeste afstandsbedieningen werken met een zender-IC waarin de infrarood signalen zijn opgeslagen. Het is natuurlijk mogelijk deze informatie te kopiëren en in een geheugen te plaatsen en wel zodanig dat dit voor de TVM bruikbaar is. Een nadeel hierbij is dat, wil de TVM universeel te gebruiken zijn, van iedere reeds bestaande afstandsbediening de infrarood informatie in de TVM aanwezig moet zijn. Dit kost een grote hoeveelheid geheugenruimte en programmeerwerk.

De TVM moet dus per afstandsbediening afzonderlijk te programmeren zijn. Hiervoor wordt de TVM uitgerust met een infrarood ontvanger, zoals te zien is in het blokschema van figuur 4.1.



figuur 4.1 Blokschema van de TVM.

Door nu met de bestaande afstandsbediening het gewenste commando uit te zenden kan de TVM dit signaal met behulp van de ontvanger registreren, waarna codering en opslag plaats vindt, zoals beschreven in hoofdstuk 3.

In eerste instantie worden de signalen van één bepaald commando opgeslagen in een RAM, waarna ze onderling vergeleken kunnen worden. Er worden twee verschillende signalen per commando opgeslagen in een EEPROM. Het is noodzakelijk dat dit een EEPROM is omdat, wanneer de TVM voor een andere afstandsbediening geprogrammeerd moet worden, de reeds aanwezige informatie in het geheugen wisbaar moet zijn en deze informatie bij het wegvallen van de voedingsspanning niet verloren mag gaan.

Om de infrarood commando's weer uit te kunnen zenden is een infrarood-zender in de schakeling opgenomen.

De 80C31 microprocessor van Intel heeft twee interne timers van elk 16 bit die de klokfrequentie delen door twaalf. Deze klokfrequentie bedraagt 12 MHz zodat de timers een frequentie van 1 MHz hebben. Er wordt dus getimed na 1 us. wat voldoende is om een minimale pulsbreedte van 8 us. te meten.

4.2 Bediening van de TVM

Voor de selectie van functies wordt bij de TVM gebruik gemaakt van een 12x8 LED-matrix. Elk LED duidt een geprogrammeerd infrarood commando aan (bijvoorbeeld geluid harder).

De commando's worden in de vorm van een menu aangeboden. Eén LED geeft een groep commando's (LED's) aan die volgens hun functie bij elkaar horen (bijvoorbeeld zenderkeuze commando's). Wanneer dit zenderkeuze-LED gaat branden en men geeft een druk op de knop dan gaan achtereenvolgens de LED's branden die op de zenders 1 t/m 9 duiden. Selectie van een zender (LED) gebeurt door middel van een druk op de knop.

Wil men echter geen zenderkeuze maar bijvoorbeeld het geluid harder of zachter dan moet men in het menu niet kiezen voor het zenderkeuze-LED maar voor het LED dat op geluid duidt.

Veel gebruikte functies komen het eerst aan de orde in het menu. Weinig gebruikte functies kan men eventueel onder één noemer vatten en als laatste in het menu aanbieden.

Indeling van dit menu zou kunnen zijn:

- zenderkeuze televisie
- geluid televisie
- teletekst
- videorecorder
- audio
- overige televisie (bijv. contrast, helderheid etc.)

Bovendien bevat de TVM vier zogenaamde status LED's. Deze geven de toestand van de TVM aan:

- LW = wacht status. Tijdens de wachttijd dient een verificatie van de gekozen functie te worden gegeven. Op deze manier hoeft men bijvoorbeeld niet het gehele menu opnieuw te doorlopen wanneer men per abuis een verkeerde functie heeft geselecteerd (lit. 6).
- LP = programmeer status. De TVM stelt nu de IR-ontvanger in werking en maakt zich gereed om het te ontvangen IR-sigitaal te coderen en op te slaan.
- LO = ontvang status. Deze wordt alleen gebruikt wanneer men IR-signalen wil inlezen (programmeer status). Deze LED geeft aan dat een IR-commando van de bestaande afstandsbediening ingelezen kan worden.
- LE = error status. Wanneer er met het programma van de TVM iets mis gaat dan gaat de error LED branden.

4.3 De hardware

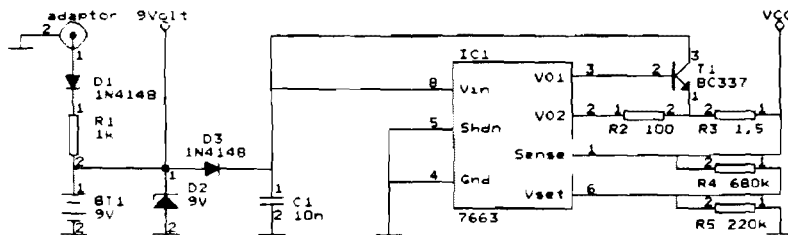
Aan de hand van het blokschema van figuur 4.1 is de electro-nische schakeling ontworpen. Belangrijke aandachtspunten hierbij waren:

- voeding door middel van een 9 Volt (oplaadbare) batterij.
- 9 Volt voeding van de ontvanger schakelen i.v.m. minimalisatie van het energieverbruik.
- 5 Volt gestabiliseerde voeding, d.m.v. spanningsstabilisator die op de 9 Volt voeding van de batterij is aangesloten
- LED-matrix die geschikt is voor een grote hoeveelheid functies (audio, video).
- reset schakeling.

In dit hoofdstuk zal de schakeling per onderdeel besproken worden.

4.3.1 Voeding en reset

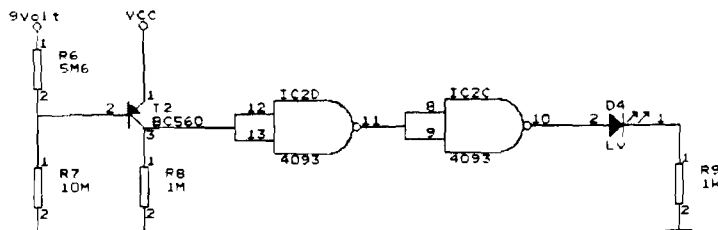
De schakeling wordt gevoed door een 9 Volt (oplaadbare) batterij. Deze 9 Volt wordt ongestabiliseerd gebruikt voor de infrarood-zender en -ontvanger. Voor alle gebruikte IC's, uitgezonderd de opamp in de infrarood-ontvanger, is een gestabiliseerde 5 Volt aanwezig. De opamp wordt gevoed met 9 Volt om een zo groot mogelijke versterking te krijgen. Het schema van de voeding ziet er als volgt uit:



figuur 4.2 Voeding.

Spanningsstabilisator ICL7663 zorgt voor een gestabiliseerde 5 Volt voeding.

Om aan te geven dat de batterijspanning te laag wordt is een spanningsindicator in de TVM opgenomen, zoals te zien is in figuur 4.3.



figuur 4.3 Spanningsindicator.

Hierbij wordt de gestabiliseerde 5 Volt geschakeld met behulp van een PNP-transistor (T2). Deze wordt aangestuurd door een spanningsdeler die 7 Volt op de basis zet. Om het uitgangsniveau van de transistor te bufferen wordt gebruik gemaakt van twee NAND's met Schmitt trigger ingangen. Er is hier gekozen voor twee NAND's omdat deze aanwezig waren in IC 2 en nog niet gebruikt werden.

De LED zal pas gaan branden als de basisspanning van T2 daalt, waardoor het uitgangsniveau van de transistor stijgt tot ongeveer 3.5 Volt (omslagspanning van de Schmitt trigger).

Om energie te besparen wordt de microprocessor, wanneer de TVM een bepaalde tijd niet gebruikt wordt, in de power down mode gezet. Voordat de microprocessor in power down gaat worden softwarematig alle poorten van de microprocessor laag (= 0 Volt) gemaakt, met uitzondering van P1.2 (= res0 in figuur 4.4). Deze poort wordt hoog (= 5 Volt) omdat deze gebruikt wordt in de reset schakeling.

De reset schakeling zorgt ervoor dat, wanneer de TVM vanuit de power down mode weer in gebruik wordt genomen, de oscillator opgestart wordt. Bovendien wordt op hetzelfde moment de Chip Enable van de EPROM laag, zodat hieruit het programma gelezen kan worden. Deze Chip Enable wordt hoog wanneer de microprocessor weer in power down gaat, aangezien P1.2 (res0) hoog wordt. Op deze manier wordt het stroomverbruik van de EPROM beperkt.

Om nu de microprocessor te resetten als deze in de power down mode is, wordt de schakeling van figuur 4.4 toegepast. Voor de verklaring van de werking van deze schakeling gaan we uit van de power down mode van de microprocessor.

Uitgang P1.2 (res0) is hoog waardoor over C4 5 Volt staat.

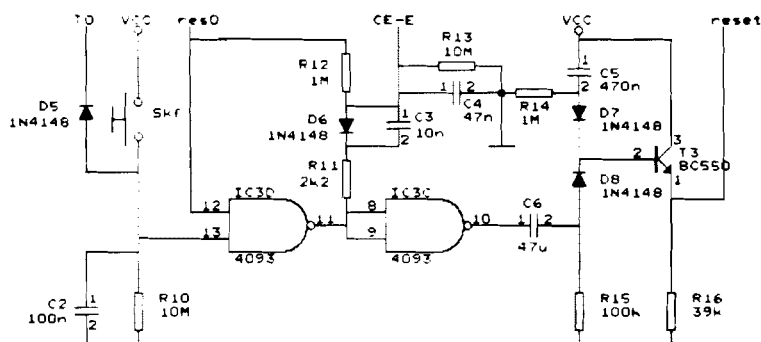
Hierdoor is de EPROM gedisablend zodat deze een geringe stroom verbruikt. De ingangen 12 en 13 van NAND D zijn nu respectievelijk hoog en laag, waardoor de uitgang hoog is.

Geeft men nu een druk op de knop (Skf) dan verandert de uitgang van NAND D naar laag, zodat C4 ontladen wordt. De EPROM wordt nu enabled, waardoor het programma opgestart kan worden.

Gelijktijdig wordt de uitgang van NAND C hoog zodat een puls op de basis van T3 komt te staan. Deze emittervolger komt even in geleiding zodat een puls op de emitter ontstaat, die als reset gebruikt wordt voor de microprocessor. De microprocessor maakt nu P1.2 (res0) meteen laag zodat C4 zich niet weer kan opladen waarmee de EPROM gedisablend zou worden.

Bovendien wordt de uitgang van NAND D nu constant hoog gehouden zodat een druk op de knop niet resulteert in een reset voor de microprocessor. Als nu de microprocessor weer in power down gaat wordt P1.2 (res0) hoog zodat we weer bij de beginsituatie zijn aangekomen.

T3 is nodig voor het geval dat de batterij wordt verwisseld. De uitgangen van de microprocessor zijn dan niet gedefinieerd waardoor P1.2 (res0) hoog kan zijn. Een druk op de knop geeft dan geen verandering op de uitgang van NAND D, zodat de microprocessor nooit meer een reset puls kan ontvangen. Door nu de collector van T3 met C5 aan de basis te leggen ontstaat bij het inpluggen van de batterij een puls op de basis. De microprocessor krijgt nu een reset puls. Bij het verwisselen van de batterij is C4 langzaam ontladen waardoor de EPROM enabled is.



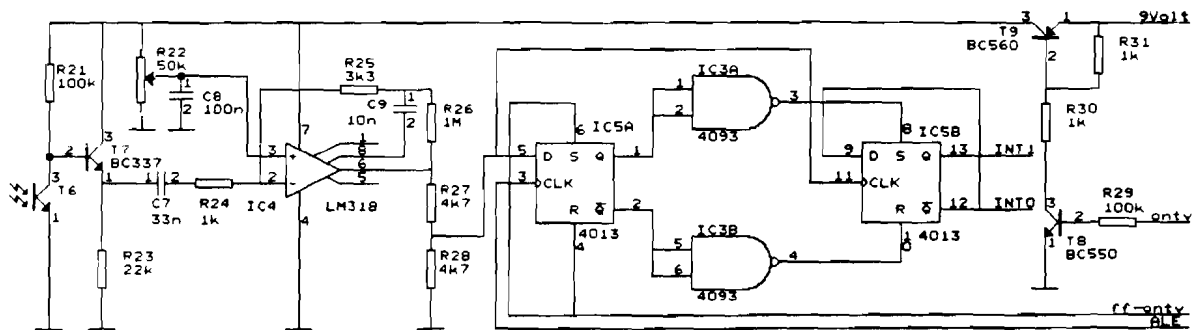
figuur 4.4 Reset schakeling.

4.3.2 Infrarood ontvanger en zender

Zowel de zender als de ontvanger worden gevoed met 9 Volt uit de batterij. Wanneer de ontvanger niet gebruikt wordt dan is T7 in figuur 4.5 constant in geleiding aangezien de fototransistor T6 geen infrarood signaal ontvangt en dus niet geleidt. Door R21, R22 en R23 loopt nu onnodig stroom.

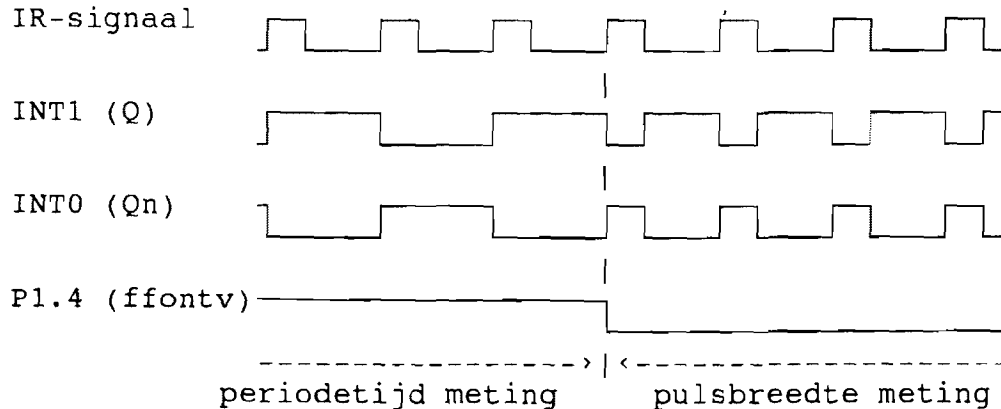
Om dit energieverlies tegen te gaan wordt de 9 Volt van de ontvanger door middel van PNP-transistor T9 geschakeld. Wanneer nu de programmeer schakelaar Skp bij de microprocessor (zie figuur 4.9) in de programmeerstand wordt gezet dan maakt de microprocessor P1.5 (ontv) hoog, zodat T8 in geleiding komt en de basisspanning van T9 van 9 Volt naar 4.5 Volt daalt. T9 komt nu in geleiding zodat de ontvanger van 9 Volt voeding voorzien wordt.

De schakeling is nu gereed om infrarood golven te ontvangen. Bij de ontvanger is een OPAMP toegepast om een hoge versterking te krijgen. Het zwakke infrarood signaal aan de fototransistor wordt hier omgezet in een blokgolf. Met de potmeter van 50 kOhm moet de spanning op de + ingang zodanig afgeregeld worden dat de versterker goed werkt voor voedingsspanningen tussen 7 en 9 Volt.



figuur 4.5 Infrarood ontvanger.

Achter de infrarood versterker bevindt zich een schakeling met twee flipflops die de omschakeling verzorgt tussen het meten van periodetijd en pulsbreedte. Wanneer uitgang P1.4 (ffontv) hoog gemaakt wordt krijgt flipflop A zowel een set als een reset signaal, waardoor beide uitgangen hoog worden. Aangezien deze geïnverteerd worden zijn de SET en RESET ingang van flipflop B beide laag. Bovendien is Qn (=geïnverteerde uitgang van de flipflop) bij flipflop B teruggekoppeld naar de D ingang en werkt de flipflop op deze manier als tweedeler. Dit is te zien in figuur 4.6 waar een infrarood signaal getekend is met daaronder het signaal op de uitgang van flipflop B. De Q uitgang van flipflop B is aangesloten op de INT1 ingang van de microprocessor en Qn op INT0. P1.4 (ffontv) zorgt voor de omschakeling van periodetijd meting (P1.4 is hoog) naar pulsbreedte meting (P1.4 is laag).



figuur 4.6 Meting van periodetijd en pulsbreedte.

Om de werking van deze periodetijd en pulsbreedte meting te begrijpen is het nodig te weten dat de timers van de microprocessor alleen kunnen werken als de gelijknamige INT-ingang hoog is. Met andere woorden, timer 0 telt alleen als INTO hoog is. Als de interruptingang laag wordt stopt de timer automatisch en wordt er bovendien een interrupt gegenereerd waardoor de processor de timerwaarde kan ophalen.

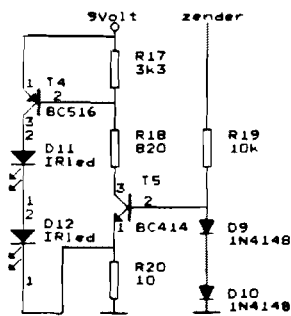
Om het mogelijk te maken dat opeenvolgende periodetijden gemeten worden wordt gebruik gemaakt van beide interrupt-ingangen INTO en INT1. Deze zijn elkaars inverse aangezien zij op de Q en Qn uitgang van flipflop B zijn aangesloten. Wanneer INTO hoog is dan is INT1 laag, ofwel timer 0 loopt en timer 1 staat stil.

Zoals gezegd werkt flipflop B als tweedeler indien P1.4 (ffontv) hoog is. Voor elke periode is dus afwisselend één interrupt ingang hoog en loopt dus de desbetreffende timer. Stel INTO is hoog en timer 0 loopt. Wanneer nu INTO laag wordt dan stopt timer 0, maar wordt tevens INT1 hoog en start dus timer 1. Gedurende de tijd dat INT1 hoog is moet de waarde van timer 0 opgehaald worden en in het RAM worden opgeslaan. Wanneer INT1 laag wordt stopt timer 1 en wordt INTO hoog waardoor timer 0 weer start. De waarde van timer 1 wordt opgehaald en opgeslaan in het RAM gedurende de tijd dat INTO hoog is. Op deze manier worden de opeenvolgende periodetijden in het RAM geregistreerd. Wanneer een bepaalde hoeveelheid informatie in het RAM is opgeslaan, wat getest kan worden met behulp van software, dan moet ook nog de pulsbreedte gemeten worden.

Als P1.4 (ffontv) laag wordt dan zal flipflop A het infrarood signaal binnen een halve microseconde volgen omdat deze flipflop met 2 MHz wordt geklokt door de ALE uitgang van de microprocessor (1/6 x de oscillator frequentie). Doordat de uitgangen van flipflop A geïnverteerd worden zullen de SET en RESET ingang van flipflop B afwisselend geactiveerd worden.

Flipflop B zal de stand van flipflop A meteen overnemen. De uitgang van flipflop B is echter de geïnverteerde van de uitgang van flipflop A omdat de RESET ingang van flipflop B het infrarood signaal volgt (clock en D ingang don't care). De Q uitgang van flipflop B (INT1) volgt de SET ingang, wat te zien is in figuur 4.6 waar INT1 de inverse is van het IR-signaal nadat P1.4 (ffontv) laag geworden is. De pulsbreedte kan nu gemeten worden met behulp van INTO aangezien deze ingang het infrarood signaal volgt. Hierna kan een representatie van het infrarood signaal gemaakt worden zoals beschreven in hoofdstuk 3.

Het schema van de infrarood zender (figuur 4.7) is rechtstreeks overgenomen van een bestaande afstandsbediening (lit. 6). De zender wordt aangestuurd door P1.3 (zender) van de processor.



figuur 4.7 Infrarood zender.

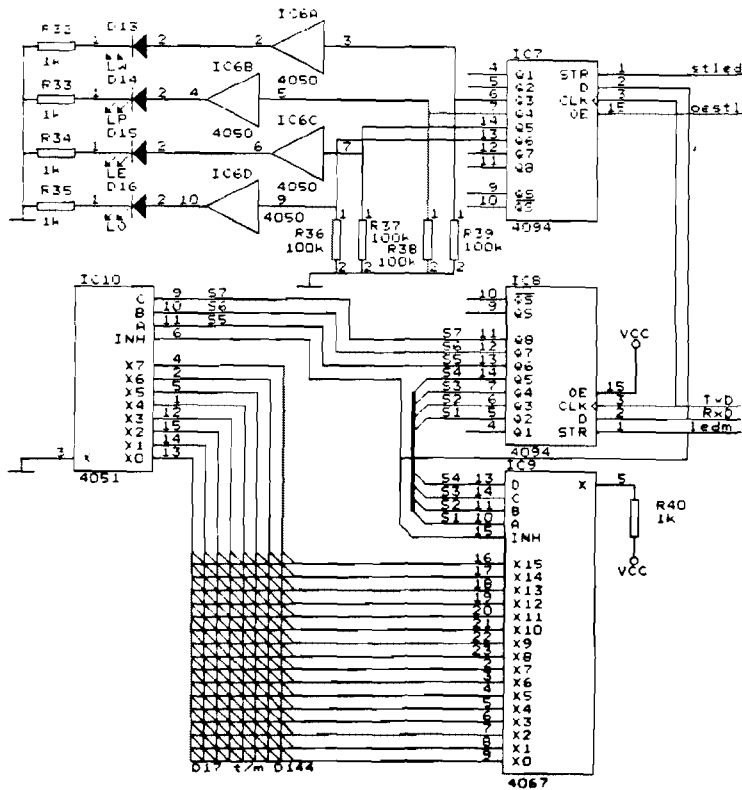
4.3.3 LED-matrix en status-LED's

Zowel de LED-matrix als de status LED's worden aangestuurd door schuifregisters die aangesloten zijn op de seriële poort van de microprocessor (lijnen RxD en TxD in figuur 4.8).

Als er een byte wordt geschreven in het interne zenderregister wordt dit automatisch serieel naar buiten gestuurd (data op RxD en klok op TxD). Wanneer nu poort P1.1 (ledm) van de processor hoog gemaakt wordt, dan wordt de strobe ingang van de LED matrix aangestuurd (IC8). Het serieel ingelezen byte komt nu op de uitgangen 0-0 tot en met 0-7 van het schuifregister (IC8) te staan. Op deze manier wordt de multiplexer (IC9 en IC10) van de LED matrix aangestuurd.

Wordt echter P1.7 (stled) van de microprocessor hoog gemaakt dan wordt de strobe ingang van de status LED's aangestuurd (IC7). Om dit register verder niet te belasten wordt een buffer op de uitgang geplaatst.

Omdat de LED's alleen mogen branden wanneer de TVM gebruikt wordt moeten, voordat de microprocessor in power down mode gaat, alle uitgangen van beide schuifregisters laag gemaakt worden. Hiervoor wordt een byte met enkel nullen in beide registers geschreven.



figuur 4.8 LED-matrix en status LED's.

Bovendien wordt de Output Enable van het schuifregister voor de status LED's geschakeld door P1.6 (oest1). Wanneer deze poort hoog wordt dan worden de uitgangen van dit schuifregister in de 'high impedance off state' gezet.

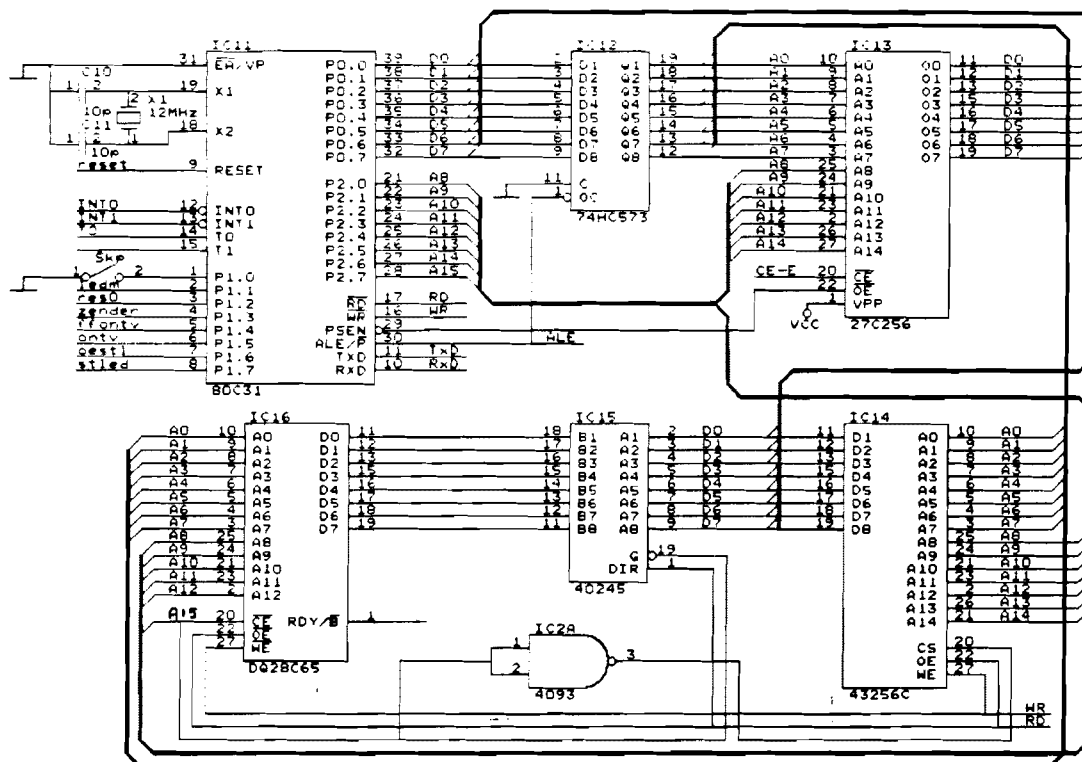
4.3.4 De microprocessor

In figuur 4.9 is het schema van de microprocessor met bijbehorende geheugens opgenomen.

Beschrijving van de functie van de diverse onderdelen:

- IC11 is de 80C31 microprocessor. Deze wordt geklokt met een frequentie van 12 MHz.
- IC12 is een latch. Pin 11 ligt aan aarde, om ervoor te zorgen dat de latch enabled is. ALE (pin 1) ligt aan de microprocessor. Als ALE laag is, komt de data die aan de ingang stond ook aan de uitgang te staan.
- IC13 is een EPROM, welke gebruikt wordt om het programma in op te slaan.
- IC14 is een RAM. Hierin worden alle tijdelijke gegevens opgeslagen
- IC15 is een buffer. Deze buffert de datalijnen naar het EEPROM.

- IC16 is een EEPROM. Hierin worden de gecodeerde IR-signalen opgeslagen.
- de NAND uit IC2 werkt als inverter en is aangesloten tussen de Chip Enable van het EEPROM en de Chip Select van het RAM. Op deze manier is ofwel het EEPROM ofwel het RAM actief.



figuur 4.9 Microprocessor en geheugen.

HOOFDSTUK 5 DE PROGRAMMATUUR

5.1 Inleiding

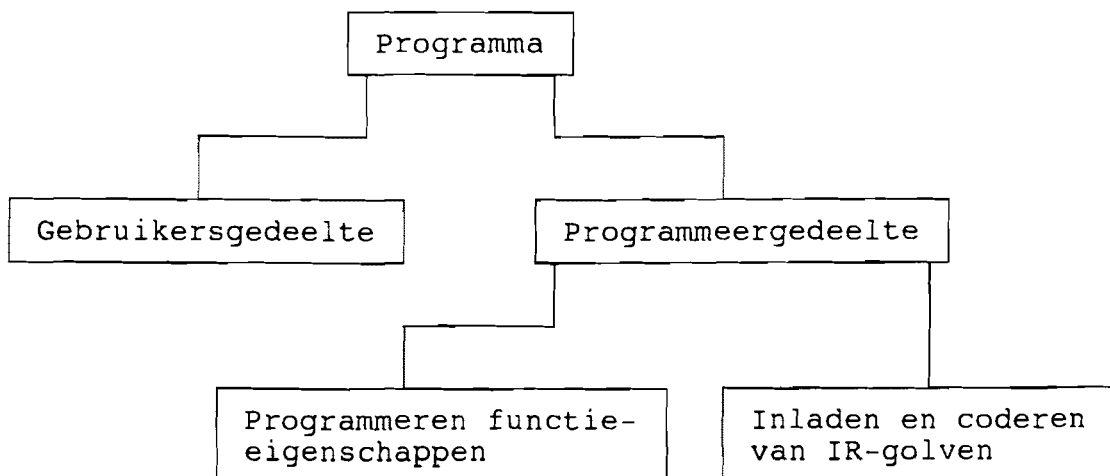
Door R. Bloks was reeds een eerste aanzet gegeven tot de ontwikkeling van de programmatuur. Een probleem hiervan was echter dat het programma niet modulair opgezet was. Dit betekende dat het zeer moeilijk was om afgebakende stukken aan te geven. Kleine veranderingen in dit programma waren dan ook vrijwel niet mogelijk. Ook het lokaliseren van fouten in dit programma was vrijwel onbegonnen werk.

De programmatuur van R. Bloks (lit. 6) is daarom zeer ingrijpend aangepast door M. Muris (lit. 7). Deze heeft het programma modulair opgezet. Eventuele veranderingen hebben nu alleen invloed op de werking van concrete duidelijke blokken (meestal subroutines). Een fout opsporen is zo veel eenvoudiger. Subroutines zijn niet alleen ingevoerd voor programmagedeelten die meerdere malen uitgevoerd dienen te worden, maar ook voor programmagedeelten die een duidelijk vastomlijnde functie hebben.

Bij het ontwikkelen van de programmatuur is gebruik gemaakt van een assembler die, geschreven in TURBO-PASCAL draait op een IBM-PC. Op dit moment werkt het programma nog niet naar behoren, maar de opzet zal niet meer veranderen. In dit hoofdstuk zal de opzet van het programma beschreven worden.

5.2 Taken van de programmatuur

In figuur 5.1) wordt de opzet van het programma schematisch weergegeven.



De programmatuur van de TVM kan in twee stukken verdeeld worden, namelijk het gebruikersgedeelte en het programmeergedeelte. Het programmeergedeelte valt uiteen in het programmeren van de functie-eigenschappen (functievolgorde, functies onderverdelen in groepen) en het inladen en coderen van IR-signalen.

M. Muris heeft in zijn stage het gedeelte van de programmatuur ontwikkeld dat de IR-signalen inlaadt en codeert. Tevens heeft hij een subroutine ontwikkeld waarmee een groepsindicatie gegeven kan worden. De subroutine waarmee de IR-signalen weer gegenereerd kunnen worden is eveneens door hem ontwikkeld.

Onze opdracht omvat het wijzigen van het programma dat geschreven is door M. Muris. Dit is nodig omdat de schakeling veranderd is en omdat sommige principes eenvoudiger uitgevoerd kunnen worden (b.v. het coderen van de gemeten pulsduur en periodetijden).

5.3 Werking van het programma

Bijlage B) geeft schematisch de werking van het programma weer. In deze paragraaf worden de verschillende functieblokken beschreven.

5.3.1 Kiezen voor programmeer- of gebruikersgedeelte

In dit gedeelte van het programma kan de keuze gemaakt worden tussen het programmeren van de TVM en het gebruiken van eerder opgeslagen functies. Dit gebeurt door het omzetten van een schakelaar. Deze schakelaar kan men in een programmeerstand zetten. Het programma test of deze schakelaar in de programmeerstand staat of niet. Is dit wel het geval dan wordt in het programma naar het programmeergedeelte gesprongen. Is dit niet het geval dan gaat het programma naar het gebruikersgedeelte.

5.3.2 De functievorgordetabel

De functievorgordetabel is een tabel waarin staat vermeld welke functies bij welke groep horen. In totaal zijn er 8 groepen en 120 functieplaatsen beschikbaar. De bovenste rij LED's op de LED-matrix vertegenwoordigen de groepen. De overige 120 vertegenwoordigen de functies. Deze tabel staat op een vaste plaats in het EEPROM. Voor we beginnen met het selecteren van een functie moet deze tabel gecopiëerd worden van het EEPROM naar het RAM, zodat we sneller in deze tabel kunnen zoeken.

5.3.3 Kiezen voor aanpassen functievolgordetabel

De gebruiker moet de mogelijkheid hebben het aantal functies wat onder een groep valt zelf te bepalen. Dit kan kenbaar gemaakt worden door het indrukken van de functiekноп (Skf) zodra de bovenste rij LED's brandt. Men moet dan wel in het programmeergedeelte van het programma zitten. Als men deze keuze maakt kan men zelf bepalen welke functies bij welke groep horen.

5.3.4 Aanpassen functievolgordetabel

In dit blok kan men het aantal LED's bepalen welke onder worden gebracht in een groep. Men kan elke willekeurige LED onderbrengen bij een willekeurige groep, zodat men bijvoorbeeld de plaatsen van de functies hetzelfde kan maken als bij de originele afstandsbediening.

5.3.5 Het selecteren van een functie

In dit blok kan een bepaalde functie geselecteerd worden. Dit geschiedt m.b.v. een menu. Eerst selecteert men de groep waaronder de functie valt om daarna een functie uit deze groep te kunnen selecteren. Als men een groep heeft geselecteerd dan gaan alle functieLED's behorende bij deze groep branden, zodat men weet welke functies bij deze groep horen. Na bevestiging kan men dan een functie selecteren. Alle keuzes die men maakt dienen bevestigd te worden omdat men ook verkeerde keuzes kan maken.

5.3.6 Het IR-sigitaal inladen, coderen en opslaan

Nadat een functieLED geselecteerd is moet onder de functieLED een functie geplaatst worden. Hiervoor wordt het IR-sigitaal van de bestaande afstandsbediening ingelezen. Dit inladen gebeurt volgens het principe beschreven in hoofdstuk 3. De pulsduur en de periodetijden worden opgeslagen in het RAM, daarna wordt er een repeterend patroon uit gehaald, de code wordt bepaald en opgeslagen in het EEPROM.

5.3.7 Het IR-sigitaal genereren

In dit blok wordt een IR-sigitaal gegenereerd. Eerst wordt de bijbehorende code (PCP's) opgehaald uit het EEPROM en omgerekend naar pulsduur en periodetijden. Aan de hand van de uitgerekenede pulsduur en periodetijden wordt het sigitaal gegenereerd en uitgezonden voor b.v. 2 seconden. Zodra dit gebeurt is gaat het programma weer naar het begin. Men kan dan dus weer opnieuw beginnen.

HOOFDSTUK 6 CONCLUSIES

Dit afstudeerwerk heeft geleid tot de volgende resultaten:

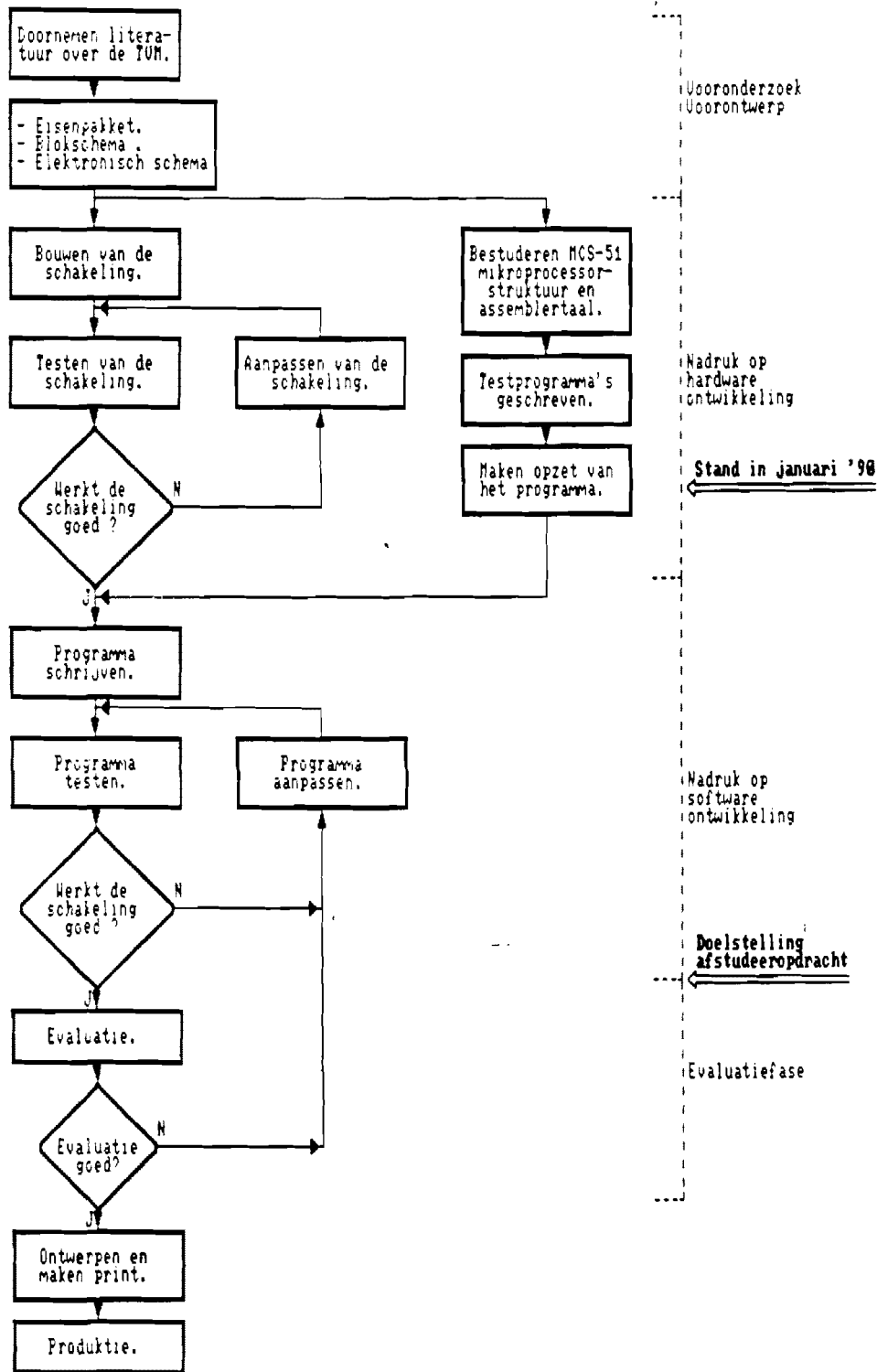
- de hardware is ontwikkeld, met speciale aandacht voor de volgende zaken;
 - voeding
 - spanningsindicator
 - energieverbruik
 - LED-matrix
 - reset schakeling
 - IC's in CMOS
- een testschakeling van de TVM is opgebouwd en werkt naar behoren.
- er is een programma ontwikkeld waarmee de TVM infrarood commando's kan inlezen, coderen en opslaan.
- er is een oplossing gevonden voor het probleem van het toggle-bit.
- er is een begin gemaakt met de ontwikkeling van de software die nodig is om de gehele schakeling te besturen.

Het is op dit moment nog niet mogelijk om de TVM te programmeren voor een reeds bestaande afstandsbediening. Software die hiervoor nodig is moet nog verder uitgewerkt en getest worden.

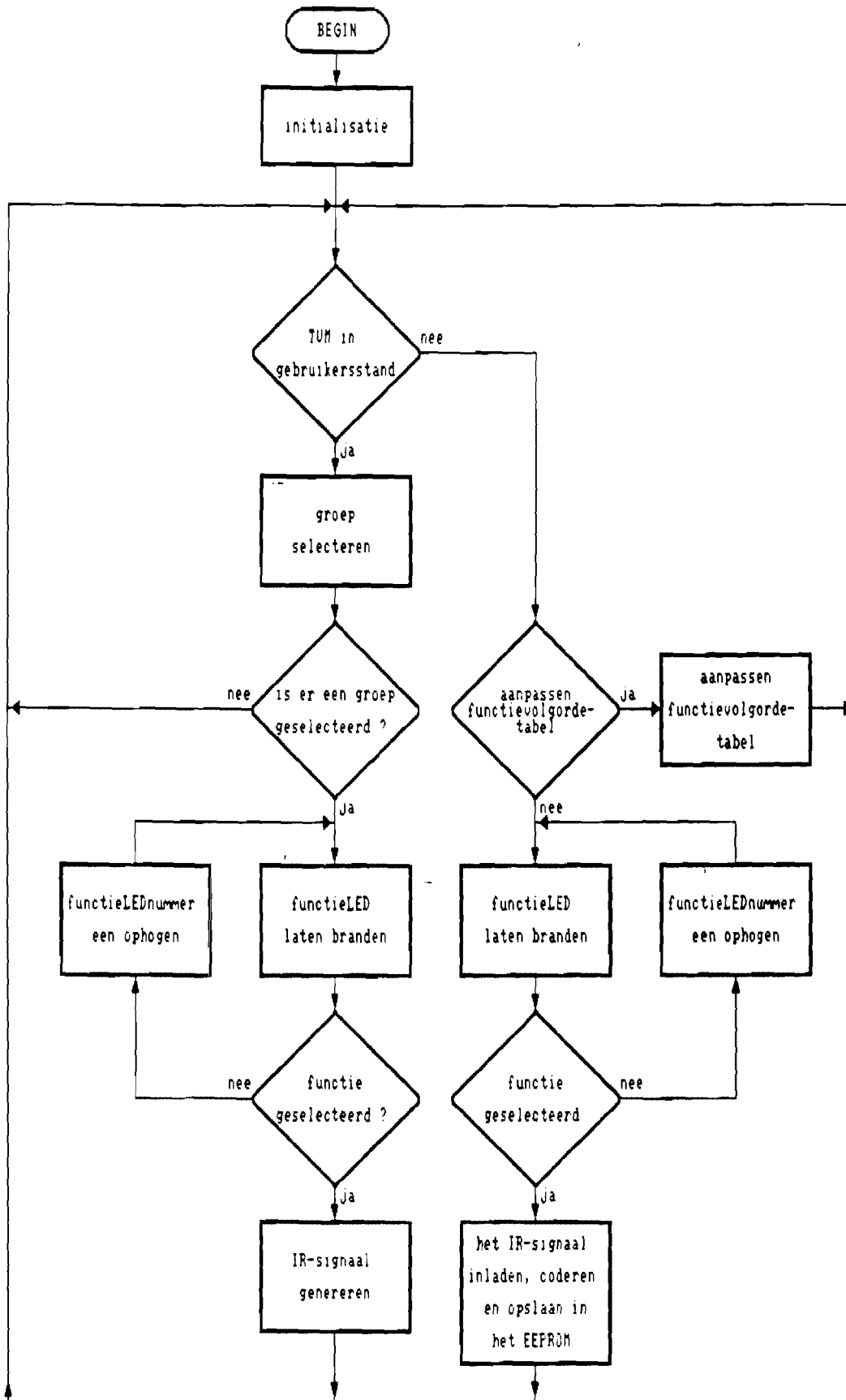
Literatuurlijst

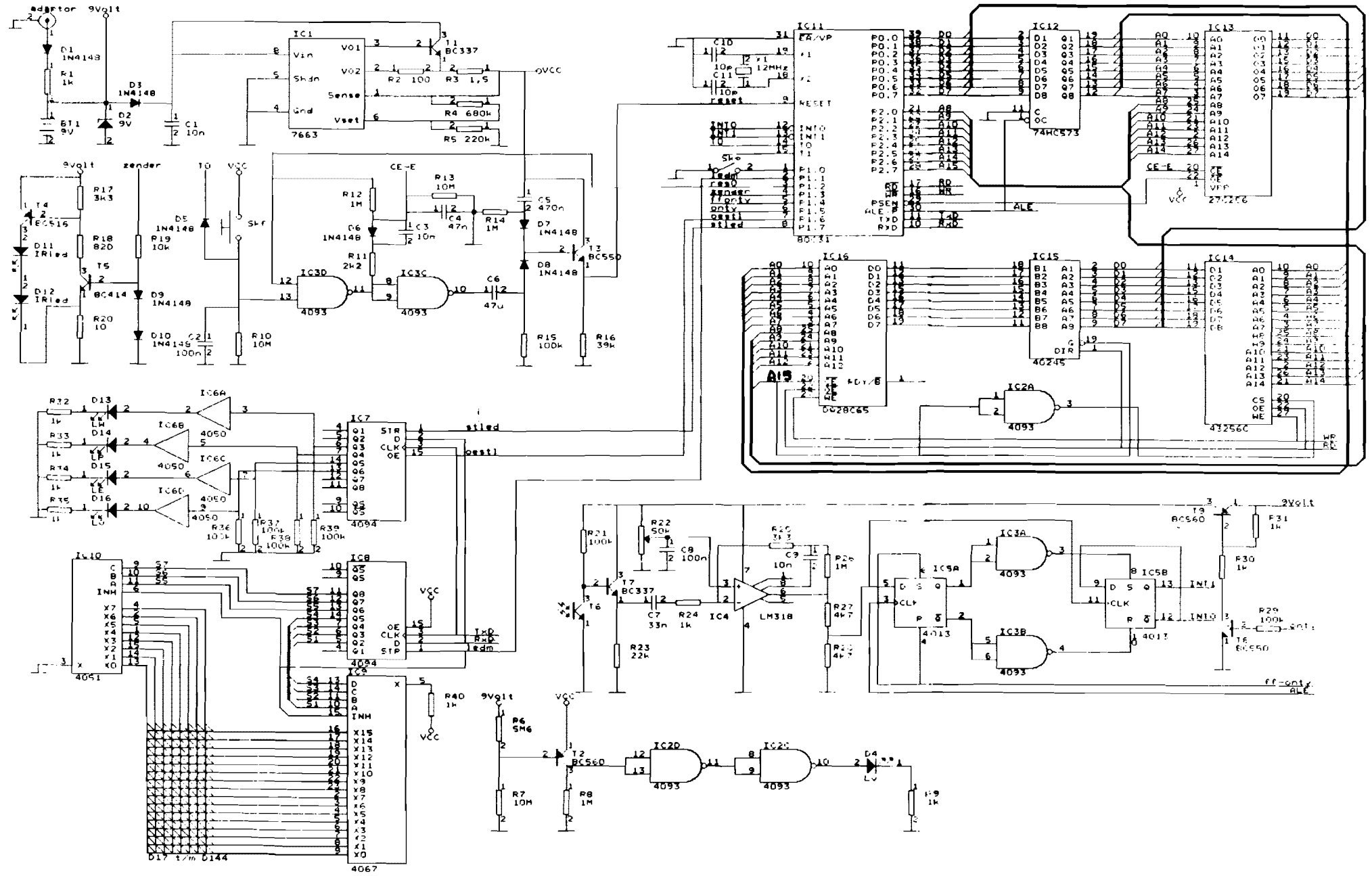
- [1] Bosch, H.; De TV-monoselector - universele TV-afstandsbediening met een maakcontact voor gehandicapten. Stageverslag TUE vakgroep EME, Eindhoven 1982.
- [2] Hieltjes, P.; De televisie-monoselector - een hulpmiddel voor zwaar lichamelijk gehandicapten. Stageverslag TUE vakgroep EME, Eindhoven 1983.
- [3] Muijtjens, P.; De TV-monoselector - een hulpmiddel voor gehandicapten. Stageverslag TUE vakgroep EME, Eindhoven 1983.
- [4] Vandamme, J.; Een industrieel ontwerp voor de TV-monoselector - een hulpmiddel voor gehandicapten. Stageverslag TUE vakgroep EME, Eindhoven 1984.
- [5] Sprenkels, H.A.A.; De TV-monoselector - een universele TV-afstandsbediening voor gehandicapten - uitgevoerd met CMOS IC's. Stageverslag TUE vakgroep EME, Eindhoven 1985.
- [6] Bloks, R.H.J.; Een universele TV-monoselector met ingebouwde infrarood-ontvanger/zender. Stageverslag TUE vakgroep EME, Eindhoven 1986.
- [7] Muris, M.N.M.; De programmatuur voor een universele TV-monoselector met ingebouwde infrarood-ontvanger/zender. Stageverslag TUE vakgroep EME, Eindhoven 1986.
- [8] Microcontroller Handboek 1983. Intel Corporation, SantaClare 1983.

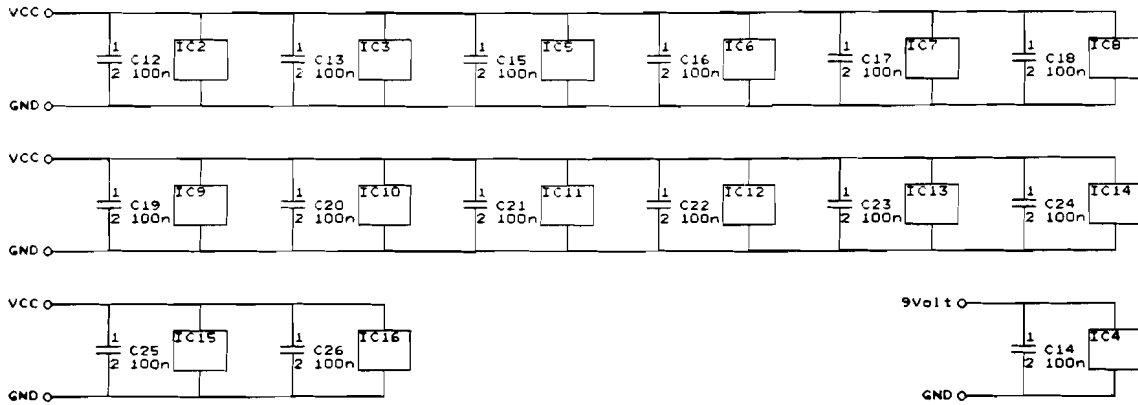
Bijlage A Ontwikkelingsverloop TVM

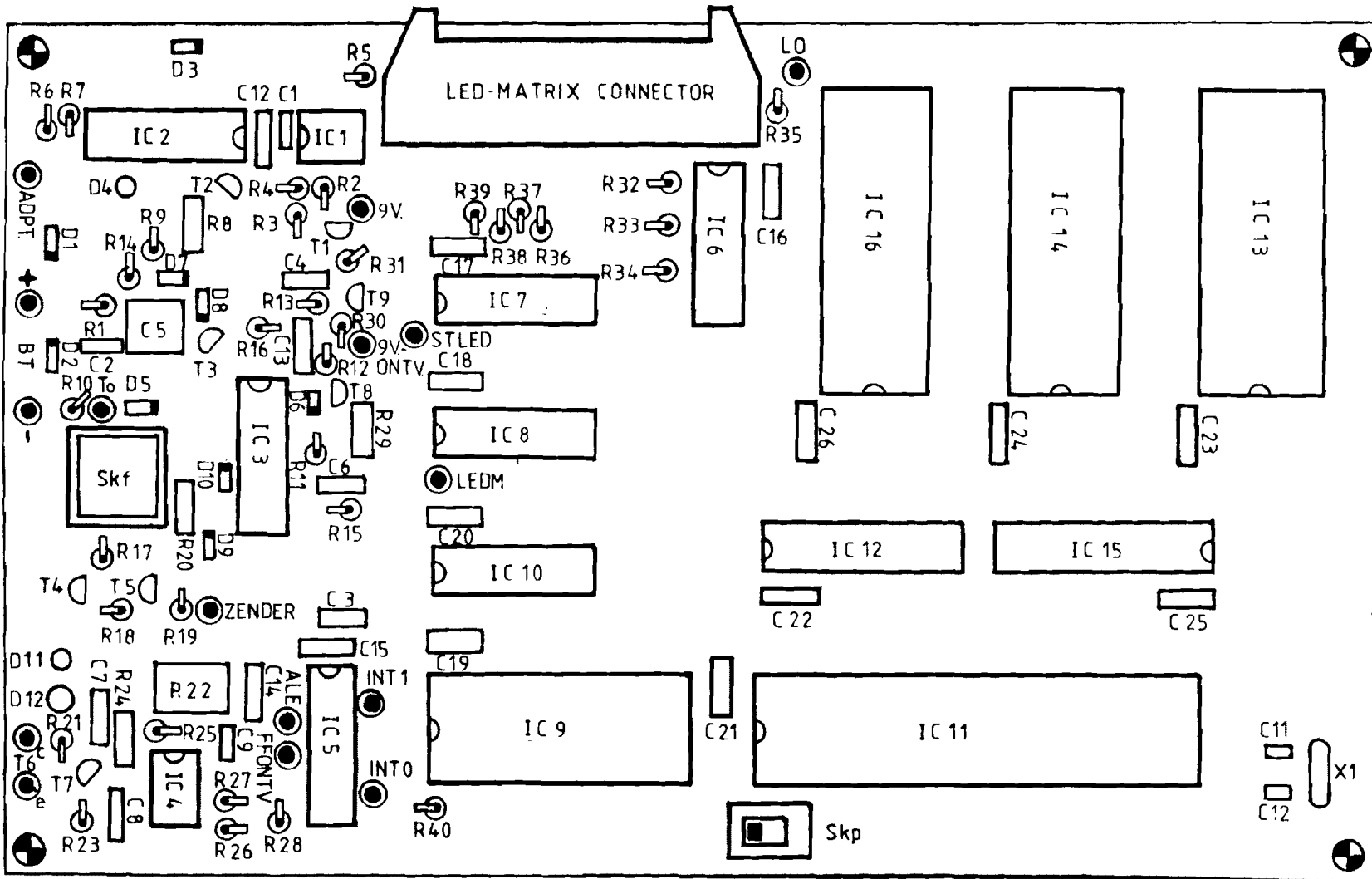


Bijlage B Schematische werking van het programma





Bijlage D Ontkoppelingen over de voeding



Bijlage E Lay-out van de print

Bijlage F StuklijstIC's

| | |
|---------|----|
| ICL7663 | 1x |
| 4093 | 2x |
| LM318 | 1x |
| 4013 | 1x |
| 4050 | 1x |
| 4094 | 2x |
| 4067 | 1x |
| 4051 | 1x |
| 80C31 | 1x |
| 74HC573 | 1x |
| 27C256 | 1x |
| 43256C | 1x |
| 40245 | 1x |
| DQ28C65 | 1x |

Weerstanden

| | | |
|-----|------------|-----|
| 1,5 | Ohm | 1x |
| 10 | Ohm | 1x |
| 100 | Ohm | 1x |
| 820 | Ohm | 1x |
| 1 | kOhm | 10x |
| 2,2 | kOhm | 1x |
| 3,3 | kOhm | 2x |
| 4,7 | kOhm | 2x |
| 10 | kOhm | 1x |
| 22 | kOhm | 1x |
| 39 | kOhm | 1x |
| 100 | kOhm | 7x |
| 220 | kOhm | 1x |
| 680 | kOhm | 1x |
| 1 | MOhm | 4x |
| 5,6 | MOhm | 1x |
| 10 | MOhm | 3x |
| 50 | kOhm potm. | 1x |

Condensatoren

| | | |
|-----|----|----|
| 10 | pF | 2x |
| 10 | nF | 3x |
| 33 | nF | 1x |
| 47 | nF | 2x |
| 100 | nF | 2x |
| 470 | nF | 1x |

Diodes

| | |
|-----------|-----|
| 1N4148 | 8x |
| 9V zener | 1x |
| LED groen | 5x |
| LED rood | 64x |
| LD271 | 2x |

Transistoren

| | |
|-------|----|
| BC337 | 2x |
| BC560 | 2x |
| BC550 | 2x |
| BC516 | 1x |
| BC414 | 1x |
| BP103 | 1x |

Diversen

| | |
|----------------|----|
| Batterij 9V | 1x |
| Crystal 12 MHz | 1x |
| Drukknop | 1x |
| Schakelaar | 1x |
| Connector | |
| 20 polig | 1x |

Aanvulling bij: De TV-monoselector, een hulpmiddel voor motorsich
gehandicapten.

90 EME 01.

- [1] In het schema van de ontvanger (blz. 21) moet R21 (100kOhm) vervangen worden door een weerstand van 1kOhm. Dit omdat de fototransistor een parasitaire capaciteit van collector naar emitter bezit met een waarde van 9 pF. In serie met R21 levert dit een RC-tijd van ongeveer 1 us. Hierdoor wordt de ontvanger te traag voor de infrarood signalen. Wanneer R21 gelijk is aan 1kOhm dan wordt de RC-tijd ongeveer 10 ps., waardoor de fototransistor sneller kan schakelen en het infrarood signaal wel kan volgen.

- [2] Een druk op de functie schakelaar Skf wordt gedetecteerd m.b.v. T0 van de microprocessor. Om T0 als ingang te gebruiken moet deze hoog worden. T0 detecteert een verandering van hoog naar laag. Aangezien bij een druk op Skf T0 van laag naar hoog gaat moet tussen Skf en T0 een inverter geplaatst worden. In het schema van de reset schakeling (blz. 20) wordt deze inverter geplaatst tussen D5 en T0. De reset schakeling blijft op deze manier direct op Skf aangesloten.