
USER MANUAL PC-1 MICROCOMPUTER

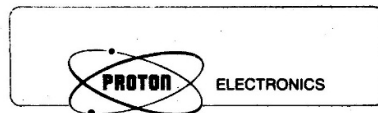


ROVC

Postbus 117
6710 BC Ede
08380-32514

INHOUD

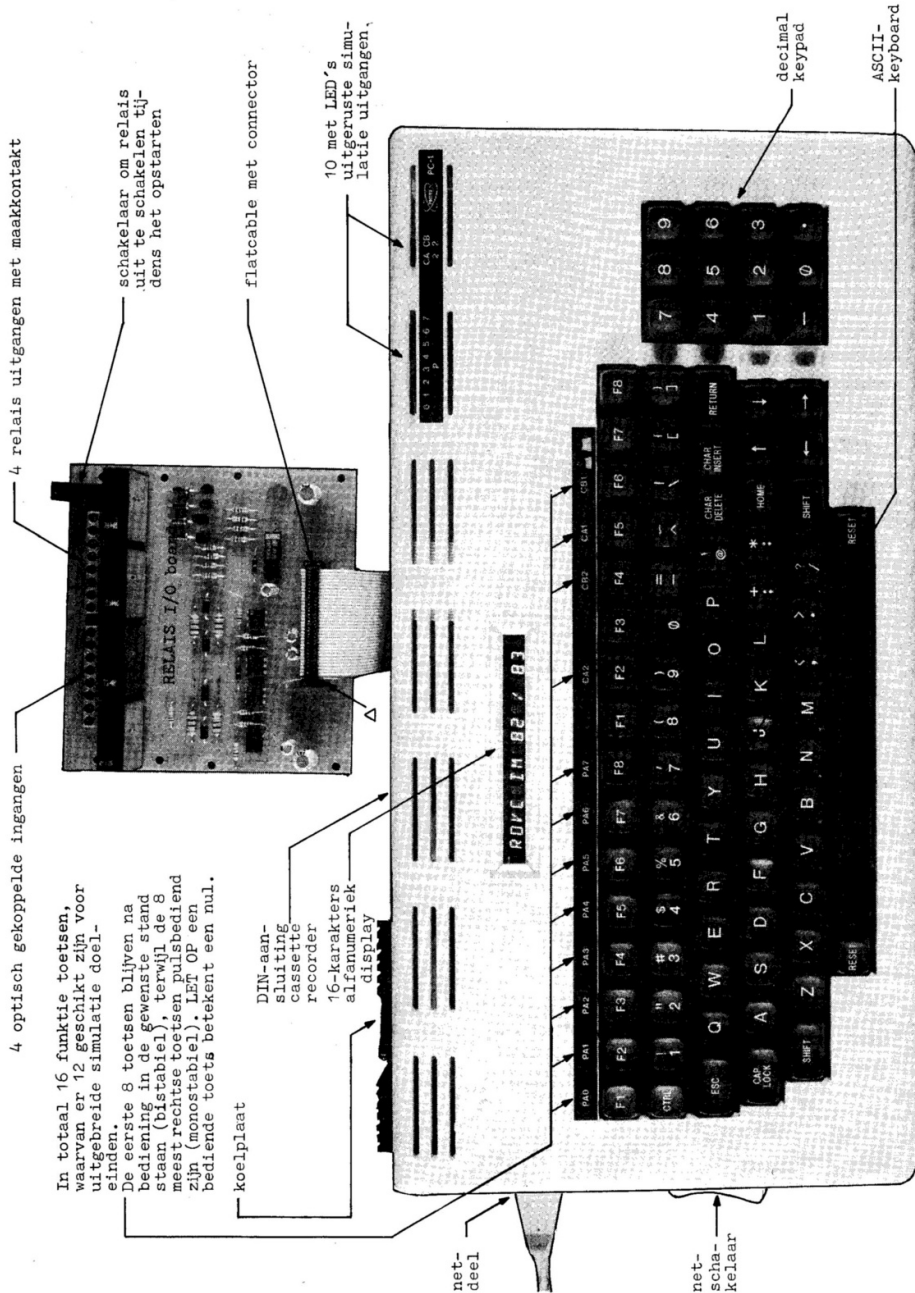
- Installatie
- Overzicht PC-1 kommando's
- Memory-map
- PC-1 monitor



Energiestraat 30 / 1411 AT Naarden
Telefoon 02159 - 48 224 / Telex 73415

1

PROTON PC-I UNIVERSELE COMPACT COMPUTER



INSTALLATIE.

In figuur 1 is het complete bovenaanzicht van de mikrokomputer afgebeeld, zoals deze op de cursus gebruikt gaat worden.

De mikrokomputer bestaat in grote lijnen uit de mikro-komputer PC-1 en een speciale I/O-print met relais om bijvoorbeeld lampen, motoren, elektrisch bediende ventielen, enz. aan te sturen. De verbinding tussen de komputer PC-1 en de relais I/O-kaart - of op zijn Engels "board" - vindt plaats met een speciaal plat snoertje, bestaande uit 34 draadjes, een zogenaamde "flatcable". Wanneer u de mikrokomputer met het relais I/O board verbindt, moet u er op letten, dat de plug - of op zijn Engels connector - van de flatcable op de juiste manier in de connector op het relais I/O board wordt gedrukt. Hiertoe is op de beide connectors een driehoek Δ aangebracht, die bij elkaar horen te zitten. Let er bij het relais I/O board op, dat deze niet in aanraking komt met metaal en voorwerpen, waardoor elektrische sluitingen kunnen ontstaan met alle gevolgen van dien. Het relais I/O board zelf is voorzien van de mogelijkheid om vier signaalgevers en vier uitvoerorganen aan te sluiten. Zie bijvoorbeeld figuur 2.

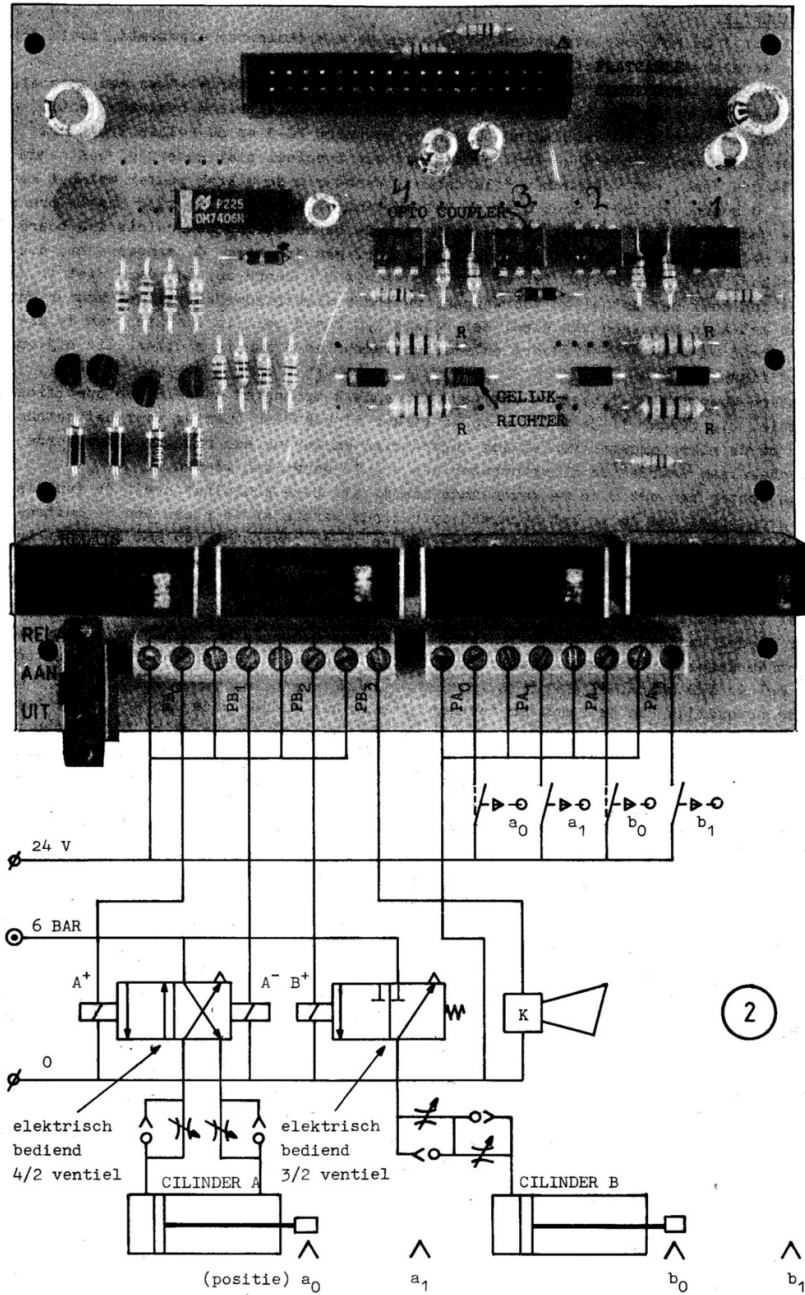
De mikro-komputer PC-1 bestaat uit verschillende componenten met elk hun specifieke functie. Zie figuur 1. Zo treft u op de linker zijkant een opening voor het netsnoer aan, om de mikrokomputer van stroom te voorzien en tevens een robuust uitgevoerde netschakelaar waarmee de mikrokomputer aan en uit gezet kan worden.

Links achter bevindt zich de zogenaamde koelplaat. Deze koelplaat heeft de functie een groot deel van de in het apparaat optredende warmte af te staan aan de buitenlucht, waarvoor eveneens de ventilatie-gaatjes in de bovenkant van het apparaat bestemd zijn. Vooral de koelplaat kan in de praktijk lekker warm worden. Ca $\pm 60^{\circ}\text{C}$. Reden, dat u er altijd voor moet zorgen, dat de lucht goed bij de koelplaat kan komen. Naast de koelplaat vindt u op de achterkant een opening, waarop de kassetterecorder door middel van een zogenaamde DIN-plug kan worden aangesloten. Daarover straks meer. Naast de kassetterecorderaansluiting bevindt zich de reeds besproken flatcable voor het relais I/O board.

Op de bovenzijde van de komputer bevindt zich in het midden onderaan het grote toetsenbord met letters en cijfers zoals ook op een typemachine aanwezig is. Dat toetsenbord noemen we vanaf nu het ASCII-keyboard (zeg aski twee ki boort). Rechts daarvan is nog een klein toetsenbord met o.a. cijfertoeetsen aangebracht. Dit toetsenbord zullen we in eerste instantie nog niet gebruiken. Overigens noemen we dat het decimal keypad (spreek uit: dissie mel ki pat). Met beide keyboards kunnen we de mikrokomputer vertellen wat we willen doen. De bovenste rij toetsen van het grote keyboard hoort niet bij het ASCII-schrijfmachinegedeelte, maar dient om eindschakelaars en signaalgevers te simuleren. De bovenste rij toetsen worden funktietoetsen genoemd. Voor deze toepassing worden niet alle toetsen gebruikt, maar wanneer u later het apparaat zou willen uitbreiden is het toch wel gemakkelijk dat ze aanwezig zijn.

Boven de funktietoetsen bevindt zich het zogenaamde display (spreek uit: displee). Op het display kan de mikrokomputer ons vertellen wat wij aan het doen zijn. Hiermee kunnen we onder andere ook zien wat er in het geheugen staat en wat er in de mikroprocessor gebeurt. Daarover later meer. Boven het decimal keypad bevinden zich 10 rode lampjes of ook wel LED's. Deze LED's dienen ervoor aan te geven of een uitgang van de komputer een één of een nul is. Daarmee kunnen uitvoerorganen worden gesimuleerd. Tot zover de benamingen en globaal de functie van de diverse onderdelen. Verbindt het relais I/O board op de juiste wijze met de mikrokomputer. Steek de stekker in het stopkontakt en de andere kant in het netdeel en schakel de mikrokomputer in met de hoofdschakelaar.

I/O RELAIKAART AANSLUITINGEN



Hierdoor licht één van de display's een moment fel op waarna de naam van de mikrokomputer op het display verschijnt. Gebeurt dit niet - kans van 1 op 100 - druk dan even beide resettoetsen - die zich links en rechts van de lange toets bevinden (spatiebalk) - tegelijk in en laat deze onmiddellijk weer los.

PC-1 kommando's.

Om de mikrokomputer wat zinnigs te laten doen, moeten we op één of andere manier met hem kunnen praten. Dat is nodig om de komputer te kunnen vertellen wat hij moet doen. Hiervoor is de mikrokomputer voorzien van een vast speciaal programma (monitor), welke ervoor zorgt, dat er bij het indrukken van toetsen iets gebeurt. Dat kan bijvoorbeeld zijn het laden van een bepaalde geheugenplaats, het starten van een zelfgeschreven programma, het schrijven op kassette, enz. We spreken hierbij van kommando's. In figuur 3 zijn deze kommando's weergegeven. Deze kommando's zullen we aan de hand van een aantal voorbeelden uitdiepen.

KOMMANDO	FUNKTIE
M van memory location RETURN:	hiermee kan een adres worden opgegeven. toon de inhoud van het huidige adres na het adres met M te hebben opgegeven en/of schrijf een nieuwe inhoud weg naar het geheugen en toon afhankelijk van het laatst bediende pijltje het volgende of het vorige adres.
→	toon na het indrukken van de RETURN-toets de volgende geheugenplaats totdat het pijltje ← wordt bediend. Tevens het herhaald uitlezen van een adres.
←	toon na het indrukken van de RETURN-toets de vorige geheugenplaats totdat het pijltje → wordt bediend. Tevens het herhaald uitlezen van een adres.
G (van go):	start een programma op het door u opgegeven adres gevolgd door een RETURN.
S (van single step):	voer één instructie uit en geef de inhoud van het volgende adres weer.
O (van offset):	bereken de offset van een branch-instructie door het gewenste adres op te geven. Na het bedienen van de RETURN-toets wordt de berekende offset op de gewenste geheugenplaats gezet.
CHAR INSERT:	voeg een geheugenpositie tussen.
CHAR DELETE:	verwijder een geheugenpositie.
R (van read):	lees een programma van de kassette in het geheugen. Wel eerst het nummer van het programma opgeven op adres \$ 0100.
W (van write):	schrijf een programma vanuit het geheugen naar de kassette. Vergeet niet op adres \$ 0100 het nummer van het programma op te geven, alsmede het eind- en startadres.
O - 9 & A - F:	Hexadecimale cijfertoeetsen.

3

Het bekijken en wijzigen van het geheugen

Wanneer we de monitor vragen de inhoud van het geheugen te tonen, moeten we eerst opgeven welke geheugenplaats we willen zien. We moeten het ADRES opgeven met de toets M van memory opgeven. Wanneer het adres opgegeven is vragen we de computer de inhoud van dit adres te laten zien met de RETURN toets. Stel: we vragen de inhoud van adres \$ 0200.

toets in	display toont	
M	0000	geef nu het adres op.
2	0002	
0	0020	
0	0200	
RETURN	0200 XX	XX = de inhoud van \$0200

Nu zien we wat de inhoud van geheugenplaats \$0200 is. Willen we dit veranderen, dan toetsen we op dit moment de nieuwe waarde in. Deze waarde wordt dan na het indrukken van de RETURN toets op de geheugenplaats \$0200 geschreven. Tikt u bijvoorbeeld "88" (is \$88) in en daarna RETURN, dan staat er op adres \$ 0200 88. Het display toont hierbij geen dollar teken maar geeft wel alvast het volgende adres weer. In dit geval dus \$ 0201.

Om nu te kontroleren of er inderdaad \$88 op adres \$ 0200 staat, kunt u op twee manieren te werk gaan.

1. toets in	display toont
M	0000
2	0002
0	0020
0	0200
RETURN	0200 88

2. Alvorens met deze methode het eerst genoemde probleem op te lossen moeten we wel even terug naar adres \$ 0201 door eventueel na het proberen van oplossing 1 nog éénmaal op de RETURN toets te drukken.

Handel dan als volgt:

toets in	display toont
← (terug)	0201 ?? willekeurige inhoud.
RETURN	0200 88

Met andere woorden in plaats van vooruit in het geheugen na het bedienen van de RETURN toets te gaan, gaan we door het bedienen van ← juist terug in het geheugen na het bedienen van de RETURN toets. Wanneer we daarin tegen →, M of de beide resettoetsen (tegelijk) bedienen wordt automatisch weer verder gelezen -na het bedienen van de RETURN toets-.

Om wat met de pijltjes en de RETURN toets te oefenen gaan we eens kijken wat er op adres \$ 0300 staat. Vervolgens doorlopen we het geheugen zowel op en neer en doen dit met behulp van de pijltjes toetsen en de RETURN toets.

Merk hierbij op, dat wanneer de RETURN toets ingedrukt houdt, automatisch die opdracht door de microcomputer wordt herhaald.

Het intoetsen van een programma geschiedt op dezelfde manier als in het voorgaande voorbeeld. Dit doen we door op de gewenste adressen met de machine instructies en bijbehorende data in te vullen.

Wanneer nu na het intikken van een programma blijkt dat we een locatie hebben overgeslagen, kunnen we de overgeslagen locatie vrij maken door alles vanaf dat punt in het geheugen tot aan het einde van het programma één positie op te schuiven. Hiervoor moeten we echter eerst het eindadres opgeven aan de monitor. Dit eindadres moeten we invullen op locatie \$102/\$103 (EOL/EOH). Stel ons programma bestaat de adressen \$0200 t/m \$0260 en we zijn op locatie \$0221 iets vergeten. We vullen eerst het eindadres in:

toets in	display toont	
M	0000	we nemen \$0260 als eindadres
1	0001	en geven dat aan de monitor.
0	0010	
2	0102	
RETURN	0102 XX	geef het laatste deel (low byte)
6	0102 X6	
0	0102 60	
RETURN	0103 XX	nu het eerste deel (high byte)
0	0103 X0	
2	0103 02	
RETURN	0104 XX	en klaar.

Nu weet de monitor tot waar het geheugen verschoven dient te worden en moet hij alleen nog weten welke locatie moet worden vrijgemaakt.

Stel we zijn op locatie \$0221 het getal \$12 vergeten:

toets in	display toont	
M	0000	geef de 'insert'-locatie
2	0002	
2	0022	
1	0221	
RETURN	0221 XX	
CHAR INSERT	0221 YY	nu schuift alles van \$0221 t/m \$0260
1	0221 Y1	één plaats op.
2	0221 12	voer nu het vergeten getal in
RETURN	0222 XX	

We hebben nu op locatie \$0221 het getal \$12 tussengevoegd en alle posities van \$0221 t/m \$0260 één plaats opgeschoven.

De omgekeerde bewerking van INSERT is DELETE. Hiermee wordt een locatie die 'te veel' is weggegooid en alle locaties daarachter tot EOL/EOH teruggeschoven. Willen we nu bij het bovenstaande voorbeeld van locatie \$0221 het ingevoegde getal (\$12) weer weghalen, dan toetsen we:

toets in	display toont	
M	0000	geeft het 'verwijder'-adres
2	0002	
2	0022	
1	0221	
RETURN	0221 12	
CHAR DELETE	0221 XX	en delete

Het in uitvoer brengen van een programma

Hiervoor zijn 2 methoden:

G = 'GO' Starten op een adres.
S = 'SST' Single step van één instructie.

Vóór het geven van een GO of een SST moet eerst een startadres worden opgegeven aan de monitor. Dit doen we door middel van:
M xxxx RETURN . Daarna drukken we op de GO of SST toets.
Bij GO start de monitor het programma, te beginnen bij het opgegeven startadres. Het programma stopt bij het uitvoeren van een BRK instructie (\$00), waarna de monitor weer actief wordt.
Met SST wordt de instructie die op het opgegeven adres staat uitgevoerd en wordt de monitor direct weer actief, zodat de processorregisters kunnen worden bekeken. Het adres van de volgende instructie moet echter weer worden opgegeven als de registers bekeken zijn, of wanneer het adresveld tussentijds veranderd is.

Offset berekening

De monitor heeft de functie-toets 0 (niet nul) voor het berekenen van sprongafstanden (branch-offset). Deze branch-offset is het verschil tussen twee adressen. Wanneer een programma een Branchinstructie heeft, wordt het sprongadres opgegeven en moet achter de instructie code de offset worden ingevuld.

Bijvoorbeeld: Op adres \$ 0200 staat BMI naar adres \$ 0210.

De monitor kan de 'offset' op locatie \$0201 zelf berekenen uit het sprongadres en het huidige adres. De berekening is:

OFFSET = \$0210 - \$0201 - 1 = \$0E (OFFSET=TARGET-ADDR-1).

Het toetsvoorbeeld is:

toets in	display toont	
M	0000	we schrijven vanaf \$0200
2	0002	
0	0020	
0	0200	
RETURN	0200 XX	
3	0200 X3	
0	0200 30	code voor 'BMI'
RETURN	0201 XX	Hier moet de offset komen
0 en geen nul	0000	geef nu het 'target'adres (sprongadres)
2	0002	
1	0021	
0	0210	
RETURN	0201 0E	de offset is nu berekend.
RETURN	0202 XX	de offset is ingevuld op adres \$ 0201.

Het gebruik van een cassette recorder voor het opslaan van programma's.

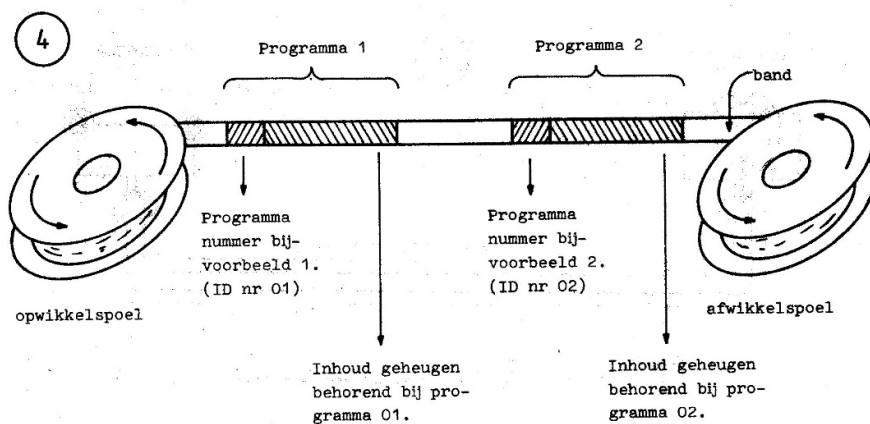
Met een cassette of tape recorder kunnen op een relatief goedkope manier programma's worden opgenomen en weergegeven. Dit gebeurt als volgt:

Na de instructiedump komen er toontjes uit de mikroprocessor, welke kunnen worden opgenomen op band. Deze toontjes stellen énen en nullen voor

Bij het opnemen (laden) van een programma gebeurt het volgende:

de toontjes op band worden nu aan de computer toegevoerd, verwerkt en in het geheugen gezet. Omdat de mikroprocessor zelf niets weet, moet hem door middel van instructies verteld worden wat er gedumpt of geladen moet worden. Hiervoor zijn verschillende gegevens nodig.

Op de eerste plaats welke geheugenplaatsen uit de mikroprocessor op de band moeten worden gezet en omdat er op een cassette meerdere programma's gezet kunnen worden, het programma wat hij moet laden. In figuur 4 is een schematische voorstelling gegeven van twee programma's, die op een band staan.



Een onhebbelijkheid van cassette-bandjes is, dat er stukjes in zitten, die geen magnetisch signaal opnemen. Zulke stukjes worden "dropouts" genoemd. Dropouts bij het dumpen van een programma kunnen grote problemen veroorzaken. Kies dus voor het opslaan van programma's een goede kwaliteitsband, bijvoorbeeld:

TDK, AGFA, BASF C 60, dus geen C 90 of erger C 120.

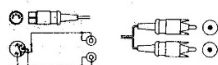
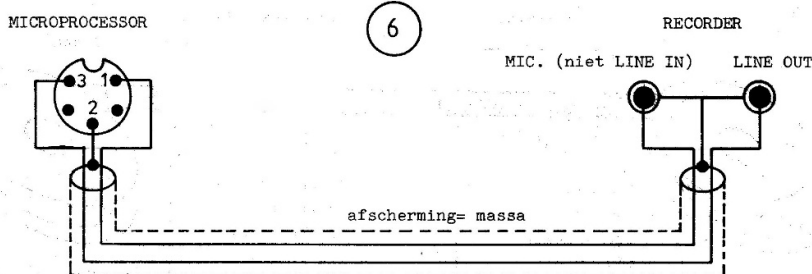
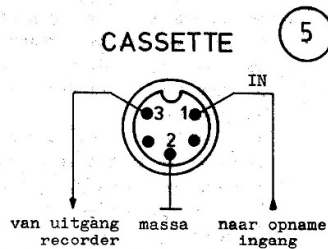
Aan de cassette recorder worden geen hoge eisen gesteld. Een eenvoudig huis/tuin/keuken apparaat voldoet goed. Let er echter op, dat de bandsnelheid wel konstant is, daar anders bij het afdraaien, de opgenomen toontjes lager of hoger worden. De mikroprocessor kan dat verschil in toonhoogte niet goed verwerken.

Het aansluiten van de cassette-recorder op de mikroprocessor.

Vanwege de grote hoeveelheid cassette-en tape-recorders, die er in de handel zijn, is het niet mogelijk voor alle types de juiste aansluiting op de mikroprocessor te tekenen. Eén ding hebben de meeste recorders echter gemeen en dat is een ingang voor opname en een uitgang voor weergave.

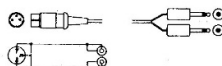
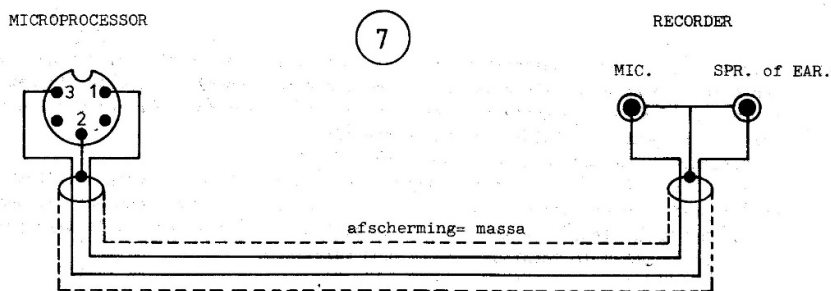
In figuur 5 is de aansluiting voor een recorder op de mikroprocessor gegeven. Hierbij dient punt 1, die een spanning afgeeft van ± 10 mV uit de mikroprocessor verbonden te worden met de mikrofooningang van de recorder of met de line low-ingang van een recorder met een gemeenschappelijke in/uitgangsplug.

Punt 3 van de ingangsplug dient verbonden te worden met de uitgang line, earphone of speaker (spr) van de recorder. Punt 2 is een gemeenschappelijke massa-aansluiting. Voor verschillende typen recorders zijn speciale verloopsnoeren verkrijgbaar. In figuur 6 tot en met 8 zijn de meest voorkomende type recorderaansluitingen getekend met vermelding van het type verloopsnoer. De snoeren zijn van het merk AMROH en verkrijgbaar bij de meeste HIFI-zaken.



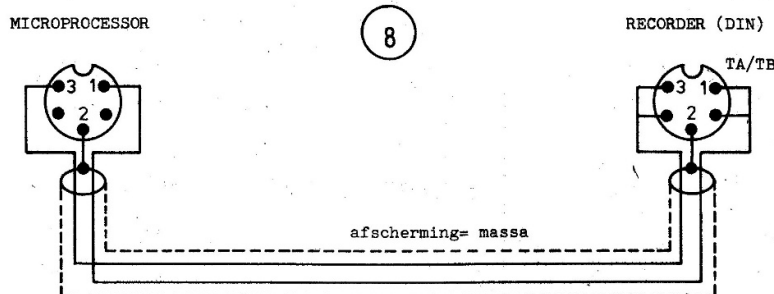
5-polige contraststekker - 2x einich stekers

ZB 147

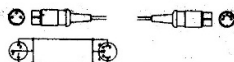


3-polige stekker - 2x 3,5 mm jackplug

ZB 156



LET OP!
In één plug
de draadjes
van punt 1
en 3 omwisselen.



3-polige stekker - 5-polige stekker

ZB 716

Het gebruik van een recorder met DIN aansluiting kan soms grote problemen opleveren bij het afspelen van de band. Dit komt omdat sommige DIN recorders een te lage uitgangsspanning afgeven, waardoor de microprocessor niet goed functioneert. Het is in die gevallen raadzaam om als ingang de microfoon ingang en als uitgang de hoofdtelefoon aansluiting van de recorder te gebruiken.

Bij stereo recorders kan het best het linker kanaal worden genomen (een recorder met DIN-aansluiting doet dit automatisch). Ook is het mogelijk beide kanalen te gebruiken. Het aansluiten daarvan voert echter te ver.

Het opnemen van een programma op band (WRITE).

Wanneer een programma op de band moet worden opgenomen, begint u met het programma een nummer te geven. Het ID-nummer. Dit nummer mag liggen tussen 1(HEX) en FE(HEX) (1 dec en 254 dec)*

Dit ID-nummer wordt op geheugenlokatie \$ 0100 gezet. Verder moet de microprocessor weten welk stuk geheugen hij op de band moet gaan zetten. Hiervoor is het begin- en het stop-adres nodig. Het stop-adres moet hierbij het eerst worden opgegeven. Dit gebeurt op geheugenlokatie \$ 0102 voor LS Bijte en op \$ 0103 voor het MS Bijte. Het begin-adres wordt als laatste opgegeven, waarna gedumpt kan worden.

Voorbeeld.

In het geheugen staat het volgende programma dat gedumpt moet worden. (Toets dit eventueel om uit te proberen in).

\$0300	00	\$0306	06	\$030C	0C
\$0301	01	\$0307	07	\$030D	0D
\$0302	02	\$0308	08	\$030E	0E
\$0303	03	\$0309	09	\$030F	0F
\$0304	04	\$030A	0A	\$0310	
\$0305	05	\$030B	0B		

1. Geef een nummer aan het programma bijvoorbeeld 02.

2. Bepaal het start-adres = \$0300

3. Bepaal het stop-adres = \$0310 (anders staat \$030F er niet op).

* Het stop-adres is het eerste adres na het einde van het op te nemen programma.

Handel nu als volgt:

toets in	display toont	kommentaar
M	0000	
1	0001	
0	0010	
0	0100	
RETURN	0100 XX	
0	0100 X0	} we geven het ID nr op (nr 02).
2	0100 02	
RETURN	0101 XX	
M	0000	
1	0001	
0	0010	
2	0102	
RETURN	0102 XX	
1	0102 X1	} we geven het LSB van het stopadres op (stopadres \$0310).
0	0102 10	
RETURN	0103 XX	
0	0103 X0	} we geven het MSB van het stopadres op (stopadres \$0310).
3	0103 03	
RETURN	0104 XX	
M	0000	
3	0003	} we geven het startadres op. (Startadres is \$0300 met inhoud 00).
0	0030	
0	0300	
RETURN	0300 00	

Zet de recorder op opname (stel eventueel de VU-meter in) en start de recorder. Pas als de aanlooptape gepasseerd is verder gaan.

W	display uit	Het display dooft tijdens het schrijven. Na het schrijven licht het display weer op.
---	-------------	--

Het programma staat nu op de band, zodat de recorder kan worden gestopt. Controleer echter wel of het programma met voldoende volume op de band staat.

Het lezen van een programma van de band (READ).

Om een programma van de band in de mikroprocessor te krijgen gebruiken we de funktie READ. De mikroprocessor moet ook hier weer verteld worden welk programma hij moet verwerken. Op adres \$0100 moet weer het ID-nummer ingevuld worden. Bij de funktie READ hebben de ID-nummers 00 en FF een speciale betekenis evenals de inhoud van adres \$0101. Normaal zijn ze echter niet nodig. Voor de liefhebbers vindt u er aan het eind van de bijlage meer over.

Voorbeeld.

Stel dat u het voorgaande voorbeeldprogramma in de mikroprocessor wilt stoppen? Zet dan als eerste de mikroprocessor even uit, zodat u zeker weet dat het programma er niet meer in zit. Schakel hierna de mikroprocessor weer in.

Handel dan als volgt:

toets in	display toont	kommentaar
M	0000	
1	0001	
0	0010	
0	0100	
RETURN	0100 XX	
0	0100 X0	} We geven het ID-nummer op van het programma dat geladen moet worden.
2	0100 02	
RETURN	0101 XX	

Zet de recorder klaar maar start de recorder nog niet. (Denk er om dat u niet halverwege het programma begint maar liefst een stuk er voor).

W | Het display gaat nu uit.

Start de recorder. Zodra op het display de tijd verschijnt, is het programma geladen en kan de recorder gestopt worden. Controleer of het programma in het geheugen zit.

Speciale mogelijkheden met het binnenhalen van programma's in de mikroprocessor.

Wanneer voor het laden het ID-nummer op \$00 gezet wordt, laadt de mikroprocessor het eerstvolgende programma wat hij op de band tegenkomt. Dus onafhankelijk wat voor een ID-nummer dat programma heeft.

Wordt voor het laden het ID-nummer op \$FF gezet, dan laadt de mikroprocessor evenals bij \$00 het eerste programma dat hij tegenkomt. Echter u kunt zelf het startadres opgeven. Dus is een programma geschreven op de adressen \$0300 tot \$0350 dan kunt u met het ID-nummer \$FF het programma laden op bijvoorbeeld de adressen \$0200 tot \$0250. Wanneer men een programma laadt en op andere adressen plaatst, dan moeten wel alle jumpadressen worden aangepast.

Om te controleren of het juiste programma geladen is, kan adres \$ 0101 bekeken worden. Hier staat namelijk het ID-nummer van het programma dat als laatste is geladen.

Voorbeeld.

Stel dat u onderstaand programma met het ID-nummer 02 van de tape in de mikroprocessor wilt laden. Echter in plaats van op adres \$0300 tot en met \$030F op adres \$0250 tot en met \$025F.

\$0300	00	\$0306	06	\$030C	0C
0301	01	0307	07	030D	0D
0302	02	0308	08	030F	0F
0303	03	0309	09		
0304	04	030A	0A		
0305	05	030B	0B		

Handel nu als volgt.

1. Geef aan de processor op dat een bestaand programma op tape op een ander adres geschreven moet worden.
2. Geef op op welk adres hij dan wel moet beginnen.

toets in	display toont	kommentaar
M	0000	
1	0001	
0	0010	
0	0100	
RETURN	0100 XX	
F	0100 FF	
F	0100 FF	
RETURN	0101 XX	
M	0000	
2	0002	
5	0025	
0	0250	
RETURN	0250 XX	
R	display uit	Start de band en wacht tot het display oplicht.

Het programma is dan geladen op de adressen \$0250 tot en met \$025F. Controleer dit. Om te kijken of het juiste programma is geladen kan op adres \$0101 gekeken worden. Hier moet \$02 staan. Namelijk het ID-nummer van het geladen programma.

PC-1 MEMORY-MAP

Adres in Indeling
hexadeci-
malecode

9

\$ 0000	ZERO PAGE (PAGINA NUL)
\$ 00EF	
\$ 00F0	ZERO PAGE VOOR DE
\$ 00FF	MONITOR
\$ 0100	GEBRUIKT DOOR DE
\$ 0136	MONITOR
\$ 0137	
\$ 01FF	STACK
\$ 0200	
\$ 07FF	RAM VOOR DE
\$ 0800	GEBRUIKER
\$ 0800	
\$ DFFF	LEEG GEBIED
\$ E000	I/O - BLOK
\$ E00F	VOOR INTERN
\$ E010	GEBRUIK
\$ E010	
\$ E01F	I/O - BLOK
\$ E020	VOOR DE
\$ E020	GEBRUIKER
\$ E01F	
\$ E020	
\$ F7FF	LEEG GEBIED
\$ F800	
\$ F800	MONITOR
\$ F800	IN EPROM
\$ FFFF	

De memory map van de PC-1.

In figuur 9 is de memory-map (adres indeling) van de PC - 1 uC weergegeven.

De Zeropage en de Ram zijn vrij door de programmeur te gebruiken. Programma's kunnen het beste vanaf \$0200 worden geschreven, terwijl de data van deze programma's in de zeropage kan staan.

De Monitor Ram is het werkgeheugen van de PC - 1 monitor. Er bevinden zich in die geheugenplaatsen een aantal belangrijke gegevens:

\$F0 NMIVEC non-maskable interrupt
\$F2 INTVEC vector voor IRQ
\$F IRQVEC vector na een IRQ

PROCESSOR REGISTERS

\$F8, F9 PC low byte, high byte
\$FA ACCU
\$FB INDEX REGISTER X
\$FC INDEX REGISTER Y
\$FD STATUS REGISTER
\$FE STACK POINTER

CASSETTE

\$0100 IDnr van de cassettefile
\$0101 TAPID IDnr van de laatste file
\$0102,0103 EOL,E0H einde van het programma

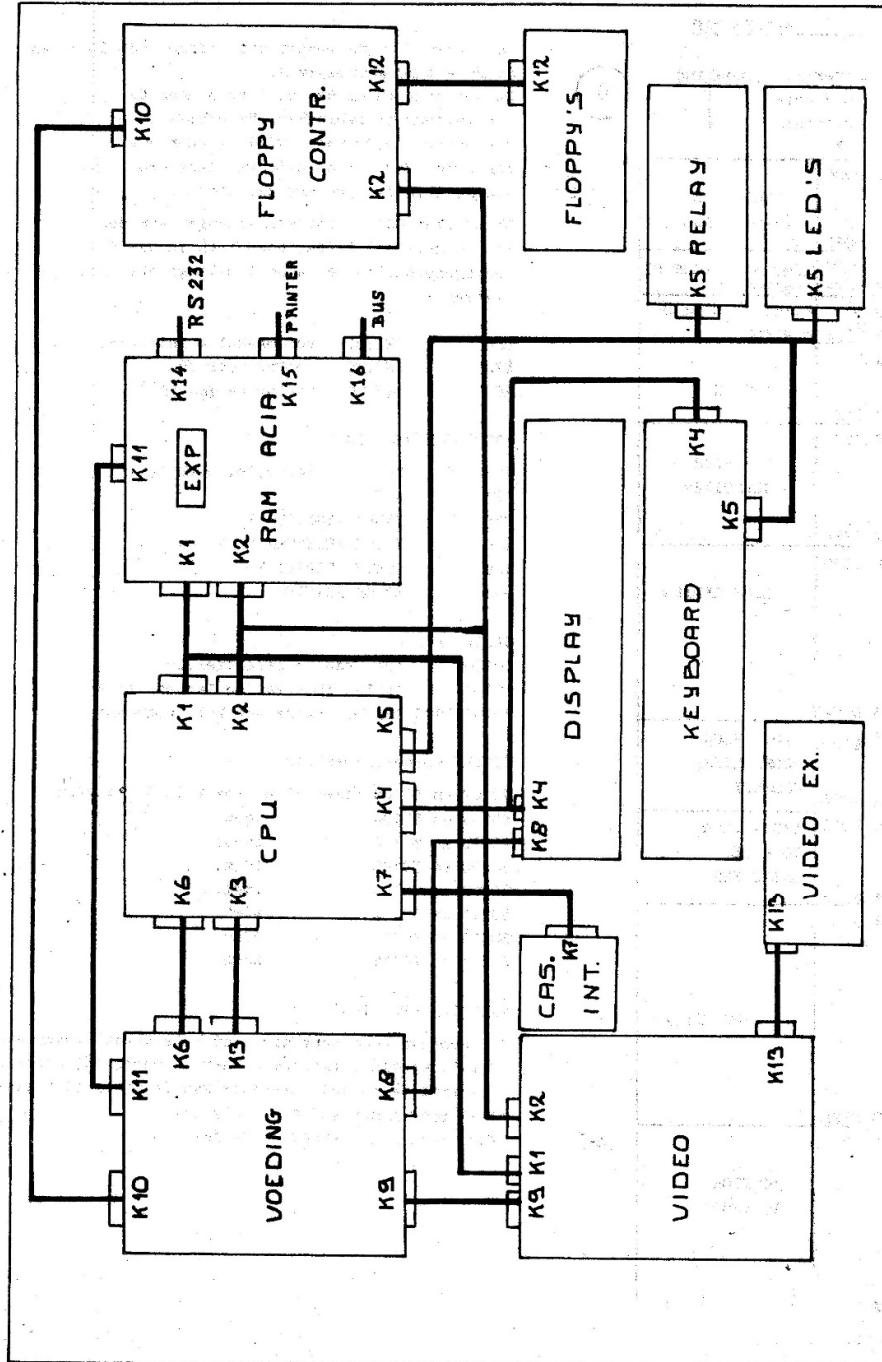
TIMERS (programmeerbaar)

\$0107 en \$0108 timer 10 mS tot ruim 10 minuten.
\$0109 en \$010A idem.
\$010B en \$010C idem.
\$010D en \$010E idem.
\$010F en \$0110 idem.
\$0111 en \$0112 idem.
\$0113 en \$0114 idem.
\$0115 en \$0116 idem.

VERBODEN REGISTERS

In verboden registers mag niet iets worden geschreven of iets worden uitgelezen, daar u anders het monitor programma beïnvloed. Hierdoor kan de uP op tilt slaan.

\$0137 tot en met \$01FF de stack.
\$E000 tot en met \$E00F de interne I/O.



BEKABELING PC

28-7-82

*** KOMPONENTENLIJST PIM82 SBC ***

PAGE: 0001

FILE = SBC82# SYSTEEM-DOCUMENTATIE

Halfgeleiders

IC1 = CD4040
IC2 = 74LS367
IC3 = 74LS260
IC4 = 74LS02
IC5 = TBP2BL22
IC6 = 74LS245
IC7 = 74LS245
IC8 = 74LS245
IC9 = 74LS374
IC10 = 74LS86
IC11 = 74LS393
IC12 = 74LS04
IC13 = R6502
IC14 = R6522
IC15 = 74LS138
IC16 = 74LS245
IC17 = (P)ROM/RAM 1
IC18 = (P)ROM/RAM 2
IC19 = (P)ROM/RAM 3
IC20 = (P)ROM/RAM 4
IC21 = 7438

Dip-switches

S1 = 2 x maak
S2 = 8 x maak
S3 = 10 x maak
S4 = 10 x maak

Transistor

T 1 = 21133 (klein)

Weerstanden

R1 = 10 K
R2-6 = 3,3 K
R7,8 = 2,2 K
R9 = 1,5 K
R10-19 = 3,3 K

Condensatoren

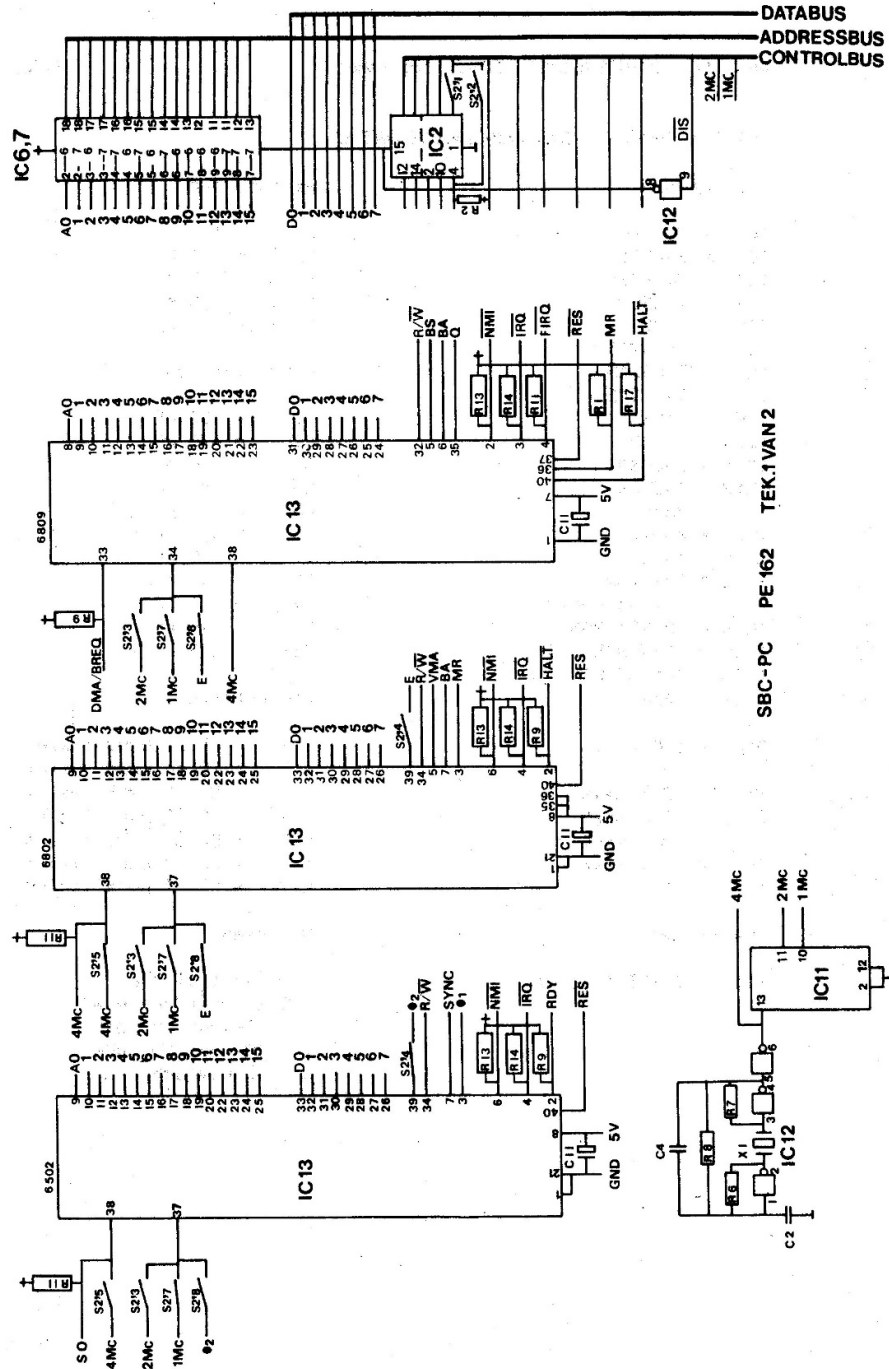
C1 = 10 nF
C2 = 470 pF
C3 = 10 nF
C4 = 47 nF
C5 = 10 nF
C6 = 4,7 uF T
C7 = 10 nF
C8 = 10 nF
C9 = 10 nF
C10 = 10 nF
C11 = 4,7 uF T

Konnektoren

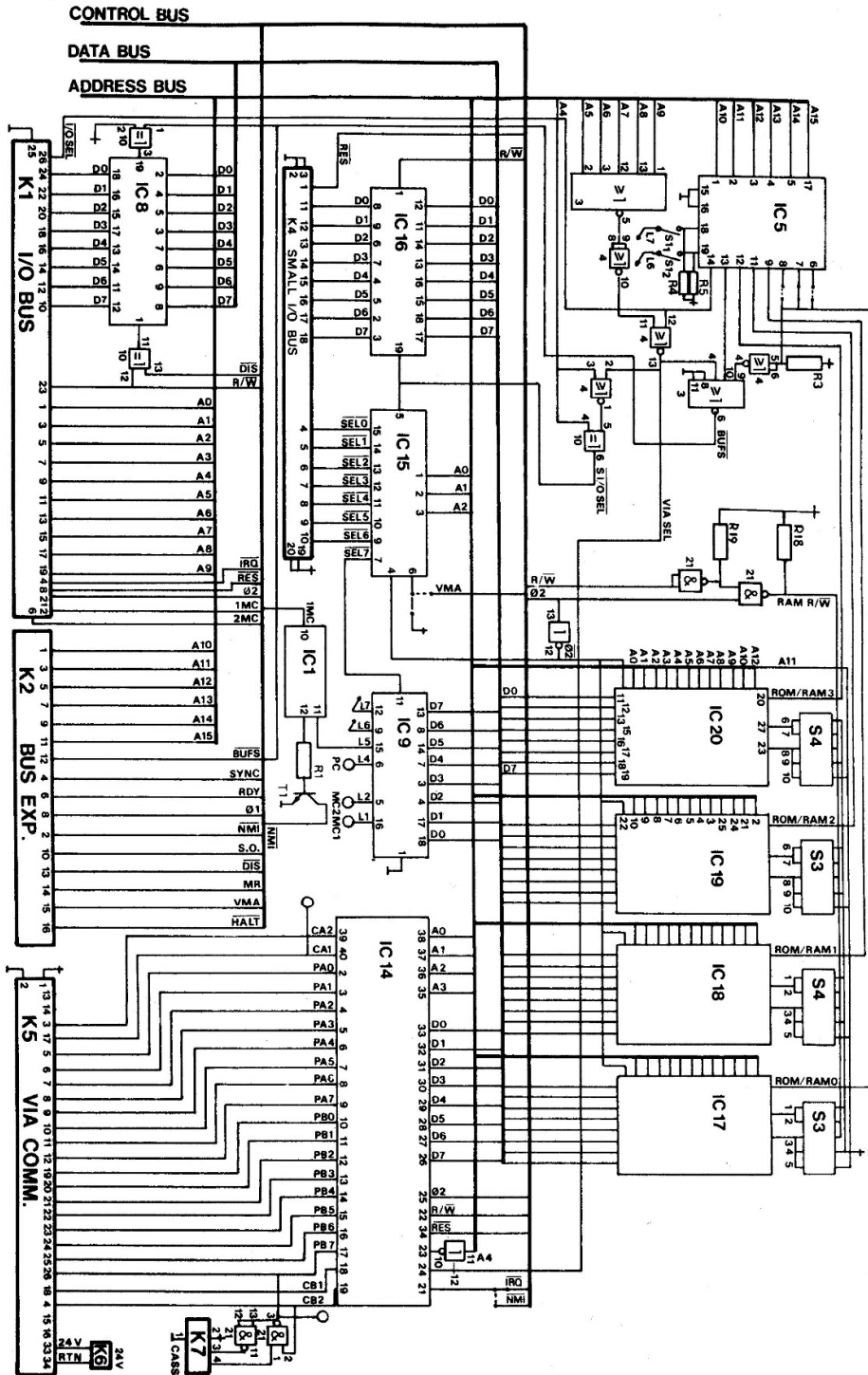
K 1 = FC 26 pens (male)
K 2 = FC 16 pens (male)
K 3 = KK 2 pens (male)
K 4 = FC 20 pens (male)
K 5 = FC 34 pens (male)
K 6 = KK 2 pens (male)
K 7 = KK 4 pens (male)

Diversen

X1 Kristal 4Mc
2 x IC-voet 40-pens
4 x IC-voet 28-pens
1 x IC-voet 20-pens
5 x soldeerpen



SBC-PC PE 162 TEK.1VAN 2



KOMPONENTENLIJST VOEDING PC-1 (ROVC) PE-175 en PE-176

PE-175

- ```

1) R1 = 270 E
1) R2 = 680 E
1) P1 = 500 E
 C1 = 0,1 uF/16V T
2) C2 = 10 uF/ 6V T
 C3 = 1 uF/ 6V T
1) D8 = 1N4001
1) D9 = 1N4001
0) T1 = LM350 (3A adjustable)
 Koelplaat S-30 (ELBOMECH)
 2 x bout M3 x 16
 2 x moer M3
 2 x soldeerlip M3
 2 x isolatiering M3
 1 x sil-pad TO-3
 1 x isolatiekap TO-3

```

- 0) Hier kan tevens de LM 123  
toegenast (=LM323)

- 1) vervalt als T1=LM 123  
2) draadbrug ' ' ' '

PE-176

- ```
C 4 = 4700 uF/16V ax
C 5 = 1000 uF/25V ax
C 6 = 4,7 uF/16V T
C 7 = 4,7 uF/16V T
C 8 = 1000 uF/25V ax
C 9 = 4,7 uF/16V T
C10 = 4,7 uF/16V T
D 1 = diode 6A
D 2 = diode 6A
D 3 = 1N4001
D 4 = 1N4001
D 5 = 1N4001
D 6 = 1N4001
D 7 = zener 400mW 6,2V
T 2 = 7812 (12V stab.)
T 3 = 7905 (-5V stab.)
TR1 = trafo TE-50
```

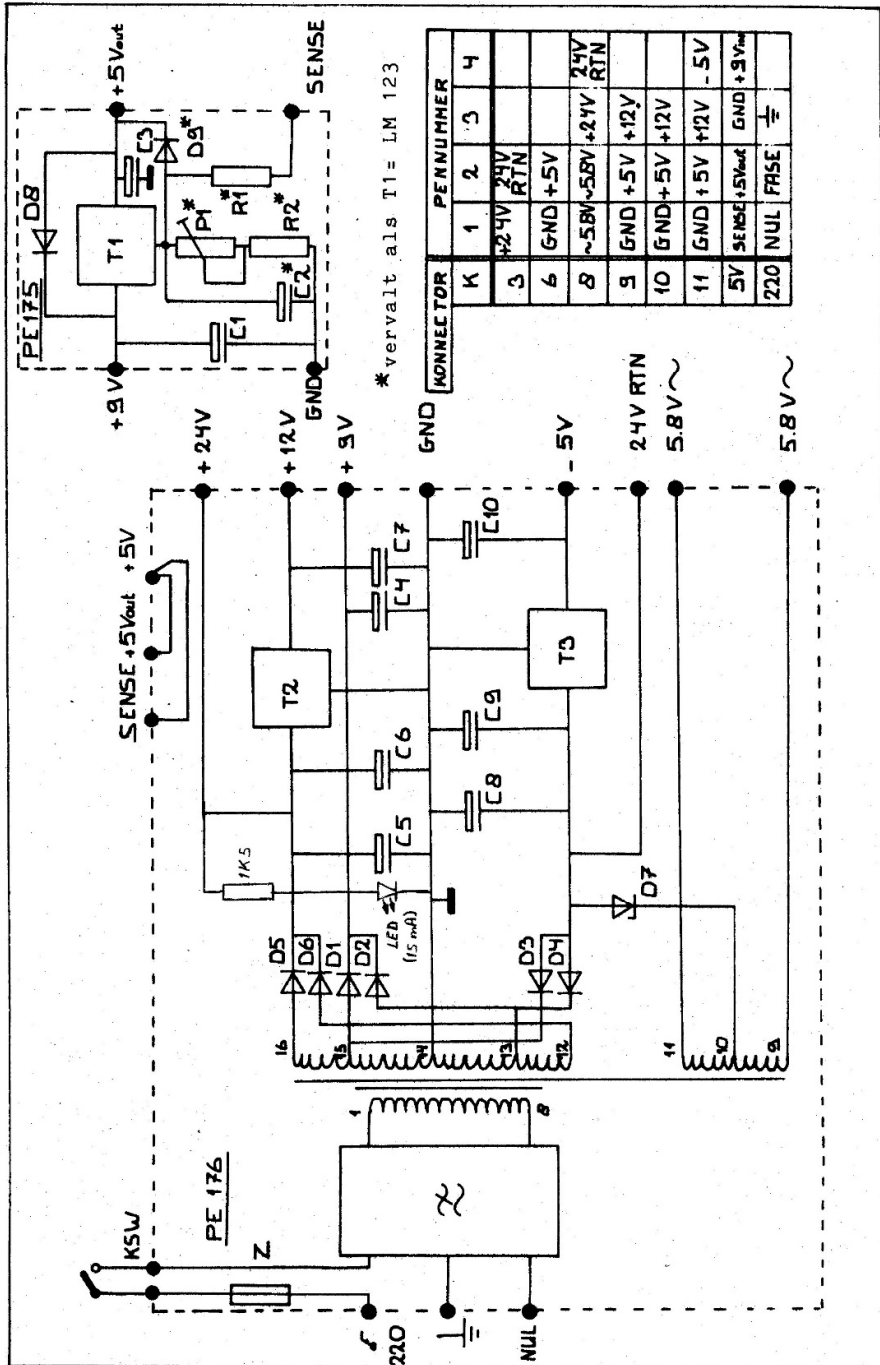
- ```

1 x zekering houder print
1 x netfilter ducati 4011-21-7002
2 x koelplaat ML-7 (elbomec)
2 x bout M3 x 12
2 x moer M3
2 x veerring M3
2 x sil-pad TO-220
1 x schakelaar 10-SC-022 (samar)
1 x netentree eurosnoer 3-polig

```

## KONNEKTORS MOLEX

- |       |         |   |       |    |      |    |       |
|-------|---------|---|-------|----|------|----|-------|
| K 3 = | 6373-02 | 2 | polig | KK | 2,54 | mm | recht |
| K 6 = | 7478-02 | 2 | polig | KK | 2,54 | mm | haaks |
| K 8 = | 6373-04 | 4 | polig | KK | 2,54 | mm | recht |
| K 9 = | 7478-05 | 5 | polig | KK | 2,54 | mm | haaks |
| K10 = | 7478-03 | 3 | polig | KK | 2,54 | mm | haaks |
| K11 = | 7478-03 | 3 | polig | KK | 2,54 | mm | haaks |
| K220= | 2599-03 | 3 | polig | KK | 5    | mm | recht |
| K5V = | 2420-04 | 4 | polig | KK | 3,96 | mm | haaks |
| Ksw = | 2599-02 | 2 | polig | KK | 5    | mm | recht |



VOEDING PE176 en 5V STAB. PE175 (BASIS VERSIE)

2.8.82

# KOMPONENTENLIJST PE 157 PIM KEYBOARD

## halfgeleiders

2x DM7416 &  
1x 74LS374 &  
1x 74LS245 &  
4x germanium diode AA119 (D1-D4) &  
1x NE555 &  
1x BC547 &

## weerstanden

24x 3K3(R6-R29)  
1x 270K(R1) &  
2x 3K3(R1,R5) &  
2x 2K2(R3,R4) &

## condensatoren

1x 1uF/50v(C1) &  
1x 1nF/KER.(C2) &  
1x 6,8uF/16v(CX) &

## schakelaars

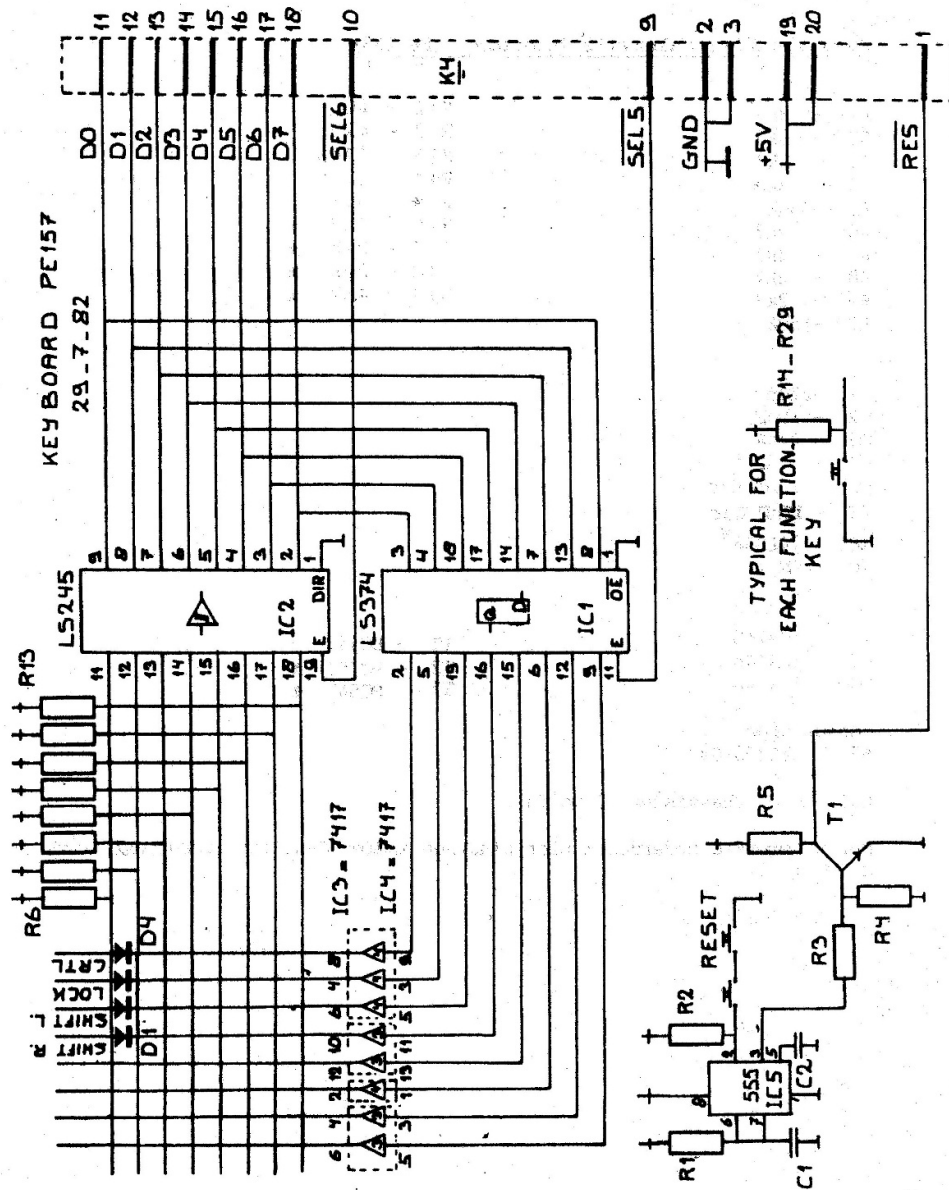
| ASCII keyboard       | funktietoetsen | decimalpad |
|----------------------|----------------|------------|
| 62x MD4P &           | 8x MD4P        | 12xMD4p    |
| 1x 4PA               | of             |            |
| 1x GT8(montageset) & | 8x 4PAon/off   |            |

## komplete set keytips

## overigen

1x 5342-2051 20polige flat cable konnektor (haaks model)  
1x 5342-2651 26polige flat cable konnektor (haaks model)  
± 50draadbruggen 1x 5342-2051 flat cable konnektor/16 pens ICvoetje(KX)

de met & gemerkte componenten komen voor op het keyboard met logica,PC-luitvoering



Komponentenlijst Cassette Interface (PE 168)

|            |             |
|------------|-------------|
| R1 - 3k3*  | R11 - 10K   |
| R2 - 3k3   | R12 - 4K7   |
| R3 - 3k3   | R13 - 470K  |
| R4 - 3k9   | R14 - 47 E  |
| R5 - 330k  | R15 - 39K   |
| R6 - 2k7   | R16 - 4K7 * |
| R7 - 4k7   | R17 - 22K * |
| R8 - 2k7   | R18 - 22K * |
| R9 - 2k2   | R19 - 4K7 * |
| R10 - 150E |             |

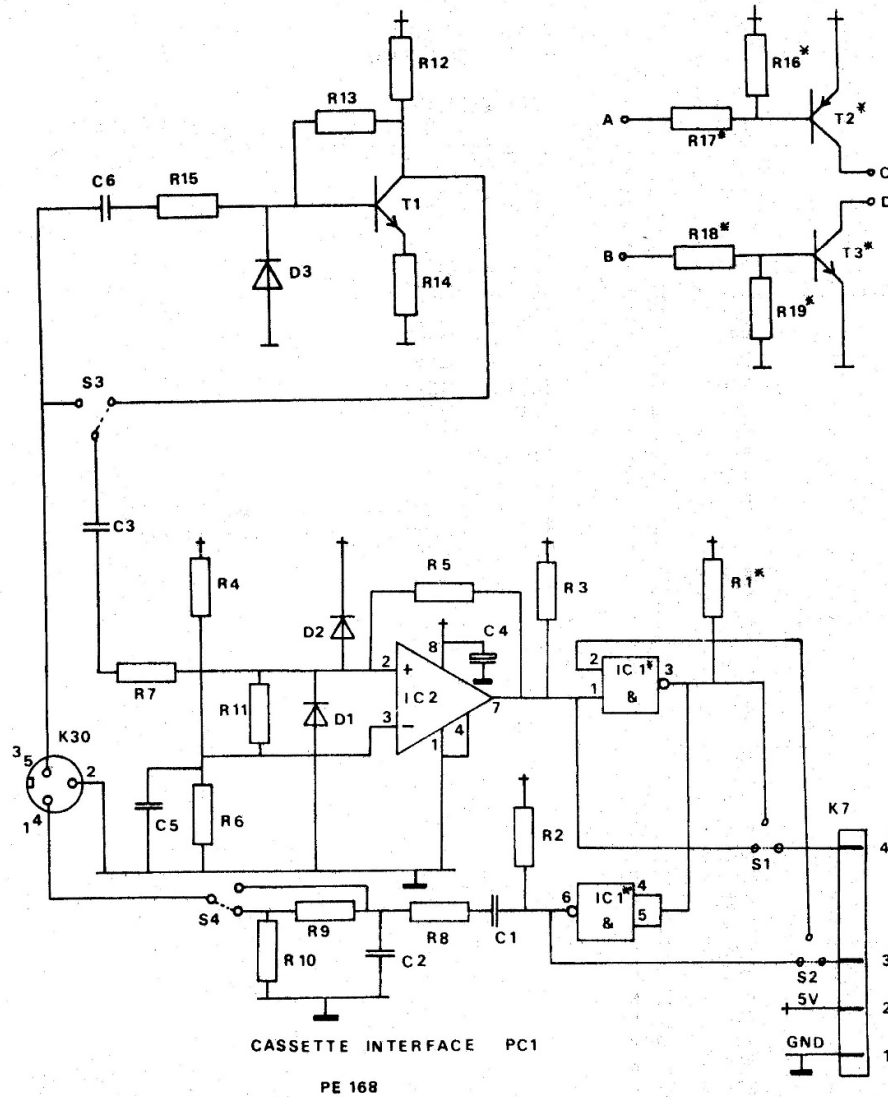
C1 - 220nF  
C2 - 15nF  
C3 - 47nF  
C4 - 1uF T  
C5 - 10nF Ker  
C6 - 10nF Ker  
IC1 - 74LS38\*  
IC2 - LM311

|             |              |
|-------------|--------------|
| D1 - 1N4148 | T1 - BC547   |
| D2 - 1N4148 | T2 - BC557 * |
| D3 - 1N4148 | T3 - BC547 * |

Konnektoren  
K7 - Kk6373-04

K30 - Din. chassisdeel 3-polig

Met \* gemerkte onderdelen niet plaatsen indien Cas. Int. toegepast wordt in de PC-1.



De met \* gemerkte komponenten worden in de PC - 1 niet toegepast.

REV. 1 6.8.82  
REV. 2 13.8.82

KOMPONENTENLIJST DISPLAYPRINT PE 161 16 DIGITS 14 SEGMENTS met punt en komma 21/06/82/1s

WEERSTANDEN

R 1 = 1 K  
R 2 = 12 K  
R 3 = 12 K  
R 4 = 1 K  
R 5 = 12 K  
R 6 = 12 K  
R 7 = 1 K  
R 8 = 12 K  
R 9 = 12 K  
R10 = 1 K  
R11 = 12 K  
R12 = 12 K  
R13 = 1 K  
R14 = 12 K  
R15 = 12 K  
R16 = 1 K  
R17 = 12 K  
R18 = 12 K  
R19 = 1 K  
R20 = 12 K  
R21 = 12 K  
R22 = 1 K  
R23 = 12 K  
R24 = 12 K  
R25 = 1 K  
R26 = 12 K  
R27 = 12 K  
R28 = 1 K  
R29 = 12 K  
R30 = 12 K  
R31 = 1 K  
R32 = 12 K  
R33 = 12 K  
R34 = 1 K  
R35 = 12 K  
R36 = 12 K  
R37 = 1 K  
R38 = 12 K  
R39 = 12 K  
R40 = 1 K  
R41 = 12 K  
R42 = 12 K  
R43 = 1 K  
R44 = 12 K  
R45 = 12 K  
R46 = 1 K  
R47 = 12 K  
R48 = 12 K  
R49 = 1 K  
R50 = 12 K

KONDENSATOREN

C 1 = 10 nF ker  
C 2 = 10 nF ker

HALFGELEIDERS

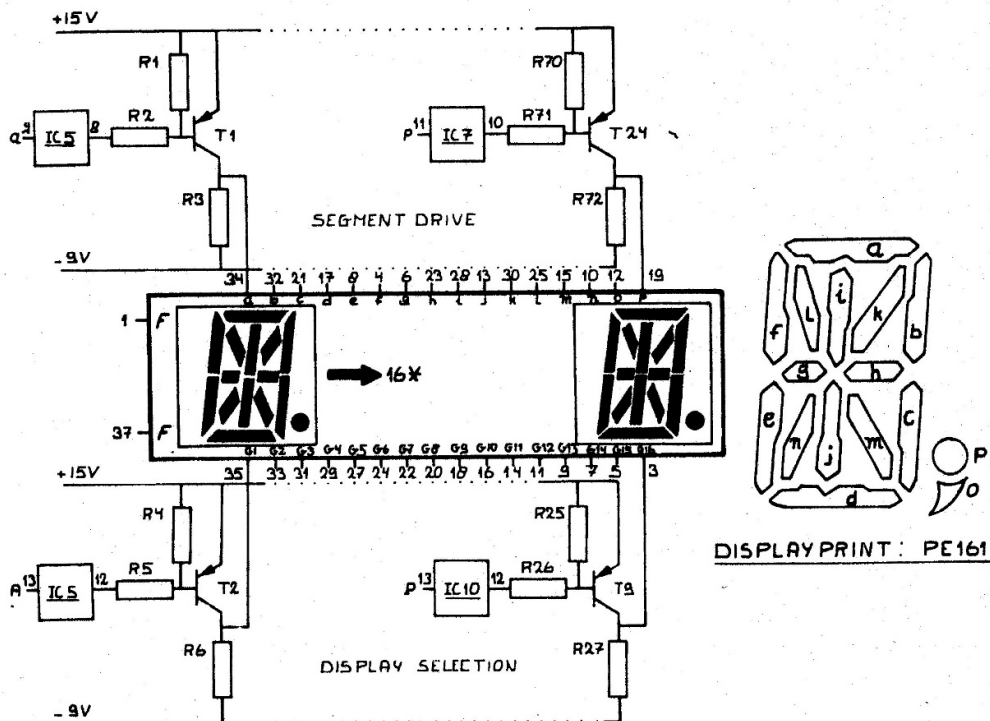
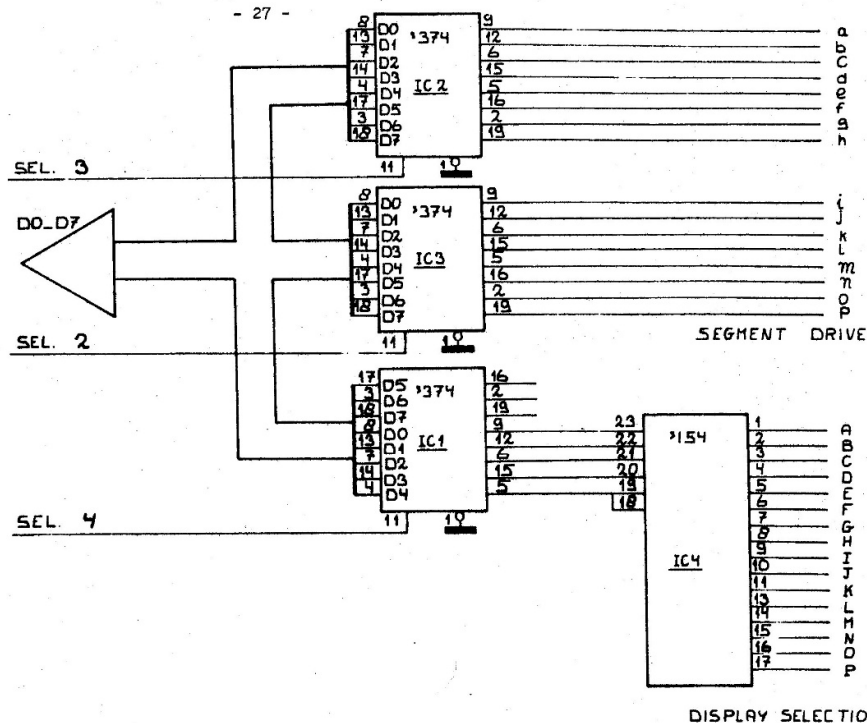
T1 - T32 = BC 557

IC 1 = 74LS374  
IC 2 = 74LS374  
IC 3 = 74LS374  
IC 4 = 74LS154  
IC 5 = 7407  
IC 6 = 7407  
IC 7 = 7407  
IC 8 = 7407  
IC 9 = 7407  
IC10 = 7407

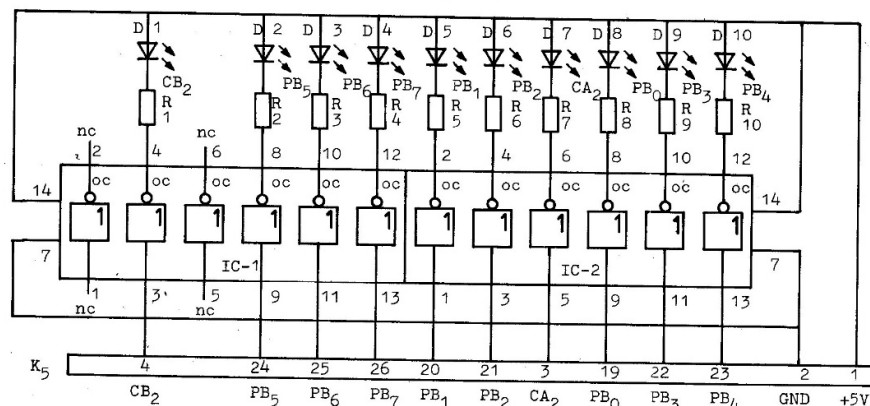
DIVERSEN

K 1 = Flatcable 20 p. haaks 5342-G1  
K 2 = Molex KK 4 p. haaks 7478-04  
DIS 1 = 16-SY-03(Z)

- 27 -



SCHEMA STATUS LED BOARD.



nc = not collect = niet aangesloten.

oc = open collector uitgang.

KOMPONENTENLIJST STATUS LED BOARD.

Halfgeleiders.

2x 7416 IC-1 en IC-2.  
10x LED rood D<sub>1</sub> tot en met D<sub>10</sub>.

Weerstanden.

10x 220 Ohm 1/8 Watt R<sub>1</sub> tot en met R<sub>10</sub>.

Overigen.

26 polige flat cable konnektor K<sub>5</sub>.

KOMPONENTENLIJST INPUT/OUTPUTBOARD.

Halfgeleiders.

8x 1N4001 D<sub>1</sub> tot en met D<sub>8</sub>.  
4x BC557 T<sub>1</sub> tot en met T<sub>4</sub>.  
4x MCT-2E OPTO<sub>1</sub> tot en met OPTO<sub>4</sub>.  
1x 7416 IC<sub>1</sub>.

Weerstanden.

4x 1k2 1/2 Watt R<sub>1</sub> tot en met R<sub>4</sub>.  
4x 3k3 1/2 Watt R<sub>5</sub> tot en met R<sub>8</sub>.  
4x 1k 1/2 Watt R<sub>9</sub> tot en met R<sub>12</sub>.  
4x 12k 1/2 Watt R<sub>13</sub> tot en met R<sub>16</sub>.  
5x 1 Ohm 1/2 Watt (draadbrugfunctie).

Condensatoren.

4x 1 uF/ 50 V C<sub>1</sub> tot en met C<sub>4</sub>.  
2x 22 uF/ 35 V C<sub>5</sub> en C<sub>6</sub>.

Overigen.

4x Relais 24Volt spoel met één maak-  
kontakt 1A 220 V RE<sub>1</sub> tot en met RE<sub>4</sub>.  
1x enkelpolige omschakelaar S<sub>1</sub>.  
1x konnektor 26 polig K<sub>5</sub>.  
2x konnektor 8 polig met schroef-  
aansluiting K<sub>6</sub> en K<sub>7</sub>.  
2x draadbrug.

