

Mineware
Multifunktiekaart

Technical Manual rel. 1.01

Voorwoord

Voor u ligt het Technical Manual van het Multifunktiebord, ontwikkeld door Miniware BV. Dit manual is bedoeld voor mensen die software willen ontwikkelen voor de modules ondergebracht op dit bord. Tevens zijn in dit manual de elektrische schema's ervan opgenomen. Deze mogen uitsluitend gebruikt worden voor toepassingen in de hobbysfeer, daar de rechten ervan aan Miniware BV behoren. In dit manual zijn programmavoorbeelden opgenomen ter verduidelijking van de tekst plus compleet uitgewerkte gebruikersprogramma's. Er is echter wel van uitgegaan dat gebruiker over een zekere basiskennis beschikt m.b.t. het functioneren van deze modules.

Op het multifunktiebord zijn de volgende modules ondergebracht:

- 64 KByte RAM uitbreiding
- RAM disk (64 of 256 KByte)
- Floppy controller
- Seriele interface: RS232 en RS422
- Centronics interface
- Real Time Clock met battery backup

Heeft men opmerkingen over dit manual, suggesties voor verbeteringen of staan er storende fouten in, laat dit dan weten aan Miniware, het liefst schriftelijk of tijdens een gebruikersdag. Alvast hartelijke dank.

Baexem, juli 1985.

Inhoudsopgave.

1	Veranderingen op het nieuwe multifunktiebord	6
2	Geheugenuitbreidung	7
2.1	Memory map standaard	
	P2000T	7
2.2	Bankswitch	8
3	RAM disk	9
3.1	Principe	9
3.2	Voorbeelden	11
4	Centronics Interface	13
5	Interruptstructuur	15
5.1	Afhandeling	16
5.2	Principe CTC	17
6	Seriele Interface	19
6.1	Besturing interface	20
6.2	I/O mogelijkheden	22
6.3	Interrupts	22
6.4	Polling	23
6.5	Baudrate	23
6.6	Besturing netwerkinterface	26
6.7	Ontvangen	27
6.8	Zenden	31
6.9	Commando/Status-registers	31
6.10	Write Registers	32
6.11	Read Registers	34
6.12	Oorzaak transmissiefouten	35
6.13	Voorbeeldprogramma's	36
7	Counter Timer Circuit	39
7.1	Werking	39
7.2	Channel Control Register	40

7.3	Inhoud Channel Control	
	Register	41
7.4	Gebruikersmogelijkheden	42
8	Real Time Clock	46
8.1	Update Cyclus	49
8.2	Status/Commando-registers	49
8.3	Voorbeeldprogramma's	53
9	Floppy Disk Controller	54
9.1	Features	55
9.2	Het werken met de controller	55
9.3	Non-DAM methode	55
9.4	Semi-DMA methode	55
9.5	Commando verwerking	56
9.6	Main Status Register	61
9.7	Commando's	62
9.7.1	Specify commando	62
9.7.2	Sense Drive Status commando	63
9.7.3	Format a Track	63
9.7.4	Seek commando	67
9.7.5	Recalibrate commando	68
9.7.6	Read Data	68
9.7.6.1	Transfercapaciteit	69
9.7.7	Write Data	71
9.7.8	Read a Track	71
9.7.9	Scan commando's	72
9.7.10	Read ID	72
9.7.11	Read Deleted Data	72
9.7.12	Write Deleted Data	73
9.7.13	Sense Interrupt Status	73
9.7.14	Ongeldige Commando's	73
9.8	Besturingsregister	73
9.9	Voorbeelden besturingsregister	75

10	Beschrijving hardware	78
10.1	I/O decodering	78
10.2	64 KByte geheugenuitbreiding	80
10.3	RAM disk	82
10.4	Seriele en paralelle interfaces	83
	Real Time Clock met battery backup	84
	Floppy disk controller	85

Figuren

1	Daisy Chain Structuur	15
2	Aansluiting op de netwerkinterface	20
3	Read en Write Registers SIO	29
4	Interrupt Vector SIO	33
5	Channel Control Register	40
6	Adres map RTC	47
7	Time, Calander and Data modes	48
8	Instruction Set FDC and Command Symbol Description table	58
9	Status Register Identification	60
10	Main Status Register	
11	Data patroon track FDC	66
12	Timing ras/cas	81

Tabellen

1	Memory map standaard P2000T	7
2	I/O-poorten RAM disk	10
3	Status signalen Centronics printer port	13
		16
4	Prioriteitvolgorde Interruptstructuur	
5	I/O-poorten SIO	21
6	Commando/status-registers SIO	22
7	Baudrates SIO	25
8	adressen Channel Control Registers	40

9 Interrupt vector CTC	42
10 Interrupt vectoren van de monitor	43
11 Intervaltijd Periodieke Interrupts	50
12 Commandotabel FDC	62
13 Sectorgrootte FDC	65
14 Transfer Capacity FDC	70
15 I/O decodering	78
16 Bankindeling RAM	80

Bijlagen

Aansluitingen sub D-connector
 Aansluitingen Floppy disk interface
 I/O poorten
 Electrisch schema Multifunktiebord
 Stuklijst
 Component bestukking
 Listing Tijd in beeld
Listing Centronics printer-driver
 Listing Printer-spooler via Centronics-
 interface

1 Veranderingen op het nieuwe Multifunktiebord.

Het nieuwe multifunktiebord wijkt in geringe mate af van de oude versie en het Philips Extension board. Dit levert echter geen nadelen op, doch bij het uitvoeren van functie's die alleen van toepassing zijn op het nieuwe bord, kunnen fouten optreden wanneer men ze op de andere borden toepast.

Het oude multifunktiebord bevatte nog geen RAM disk. Toch kunnen zij werken met software ontwikkeld voor het nieuwe bord. Is die software geschikt voor besturing van de RAM-disk, dan kan de bankswitch omgezet worden. Hierdoor kan dus de geheugenindeling veranderen. Dit is een gevolg van het feit dat voor de bank switch in de oude versie en op het Philips Extension Board elk adres geaccepteerd wordt tussen &H94 en &H97. Ook het Miniware 64KByte uitbreidingsprintje accepteert elk adres tussen &H94 en &H97.

In de nieuwe versie is dit alleen nog &H94. De andere drie adressen worden gebruikt voor het adresseren van de RAM disk.

Funktioneel is er verder niets veranderd. Alleen de basisfrequentie voor de Real Time Clock is veranderd van 4.194304 MHz in 32768 ~~K~~Hz. Hiermee dient men met het programmeren van dit device rekening te houden.

2 Geheugen multifunktiebord.

Op het multifunktiebord is een geheugenuitbreiding aangebracht van 64 KByte. Een gedeelte hiervan is in 8 KByte banken geschakeld en is daardoor niet direct toegankelijk. De P2000 heeft origineel 40 KByte van de adresruimte in gebruik.

2.1 Memory map Standaard P2000

Het standaard T-model heeft de volgende indeling:

&H0000	Monitor ROM
&H0FFF	
&H1000	ROM-pack
&H4FFF	
&H5000	Video-RAM
&H57FF	
&H5800	Video-RAM (niet in T-model)
&H5FFF	
&H6000	User RAM
&H9FFF	Indeling van het
&HA000	Multifunctie-
	bord
&HDFFF	Bank B0 B1 B2 B3 B4 B5
&HE000	
&HFFFF	tabel 1.

Hierbij dient aangetekend te worden dat het monitor-ROM voor het opslaan van systeemvariabelen de ruimte van &H6000 tot &H61FF gebruikt. Wanneer het BASIC ROM pack wordt gebruikt, is ook het gebied van &H6200 tot &H6547 in gebruik.

Het laatste gedeelte van het geheugen komt eigenlijk zes maal voor. Men heeft

echter maar direct toegang tot een van die stukken (banken). Dit principe noemt men bank switching. Door een software switch om te zetten, wordt er een andere bank in het geheugen geschakeld. De informatie in de bank waar men eerst toegang tot had, gaat echter niet verloren! Deze krijgt men terug als men die oude bank weer selecteert. Deze banken zijn elk 8 KByte groot en nemen het gebied &HE000 tot &HFFFF in.

Het User RAM is nu als volgt ingedeeld:

&H6000 - DFFF : aaneengesloten geheugen
&HE000 - FFFF : geheugen 6 maal uitgevoerd in aparte banken

2.2 Bankswitch.

De bankswitch is ondergebracht in poort &H94 (write only). Hiervan worden de drie minst significante bits gebruikt. Daar er echter maar 6 mogelijkheden zijn (0 tot 5), zal wanneer men 6 of 7 kiest, bank 0 geselecteerd worden. Bij het opstarten wordt bank 0 geselecteerd.

3 RAM disk.

De RAM disk bestaat uit 64 of 256 KByte geheugen, afhankelijk van de versie die men aangeschaft heeft. Informatie wordt hierin op dezelfde manier opgeslagen als op een diskette: in aaneensluitende blokken van 256 bytes. Het grote voordeel van een RAM disk is, dat het ophalen of wegschrijven van informatie van een RAM disk veel sneller gaat dan van een normale diskette. De RAM disk zal dan meestal een complete copie bevatten van een diskette, die na afloop van een sessie hiernaar teruggeschreven wordt. Het is echter niet zo dat de RAM disk ook door de floppy controller bestuurd wordt. Dit moet door de software of door het Operating System gebeuren. Wil men de eigen software van de RAM disk gebruiken laten maken, dan is het nuttig het volgende stukje door te nemen.

3.1 Principe.

Net als een gewone diskette, bestaat de RAM disk uit tracks en sectoren. Elk sectorblok bestaat uit 256 bytes. Deze kan men volledig, of alleen de eerste n bytes, op halen. Wanneer men alleen geinteresseerd is in de laatste m bytes, dan zal men toch de bytes die hiervoor komen ook moeten laden. Voor toegang tot een sectorblok laadt men eerst het sector- en het trackregister, waarna men via een in- of out-instructie naar het data_in/data_out-register steeds het volgende byte van het sectorblok op kan halen of weg kan

schrijven.

adres track-register	:	&H95
adres sector-register	:	&H96
adres datain/dataout-register	:	&H97

tabel 2.

Er zijn maximaal 16 sectoren, zodat alleen de 4 minst significante bits van het sector-register gebruikt worden. Heeft men een 256 KByte RAM disk, dan zullen van het trackregister alleen de 6 minst significante bits gebruikt worden terwijl voor een 64 KByte versie alleen de 4 minst significante bits gebruikt worden.

Als extra is op de RAM disk een voorziening aangebracht, welke ervoor zorgt dat de informatie behouden blijft wanneer de resetknop ingedrukt wordt. Dit is vooral bruikbaar wanneer een programma zichzelf 'ophangt'.

3.2 Voorbeelden.

```
; routine voor transfer van 256 bytes
; naar RAM disk
;
; b bevat tracknummer
; a bevat sectornummer
; hl bevat adres eerste byte
; na afloop geldt: hl = hl + 256
;
    push bc
    push af
    ld    a,b      geef tracknummer op
    out   (95),a
    pop   af      geef sectornummer op
    out   (96),a
    ld    b,0      b := 0
    ld    c,97      c := poortnummer
                  datain/dataout reg.
    otir            stuur 256 bytes naar
                  dataout reg.
    pop   bc
    ret
```

```
routine voor transfer van 256 bytes
van RAM disk naar geheugen

; b bevat tracknummer
; a bevat sectornummer
; hl bevat beginadres waar data moet
; terechtkomen
; na afloop geldt: hl = hl + 256

    push bc
    push af
    ld    a,b      geef tracknummer op
    out   (95),a
    pop   af      geef sectornummer op
    out   (96),a
    ld    b,0      b := 0
    ld    c,97      c := poortnummer
                    datain/dataout reg.
    inir            haal 256 bytes op
                    uit datain reg.
    pop   bc
    ret
```

4 Centronics Interface.

Deze interface wordt gebruikt voor het aansturen van parallele printers. De signalen die hiervoor nodig zijn kunnen betrokken worden van de printerconnector op de achterkant van de P2000. Hoe deze connector geconfigureerd is kan men in bijlage 1 zien. De interface voorziet in de hiervoor benodigde signalen.

Het dataregister van deze interface bevindt zich op adres &H98 (write only), terwijl het statusregister (read only) is gelokeerd op adres &H99. Bit 7, 6 en 5 hiervan zijn altijd 0. De overige bits bevatten de volgende terugmeldingen van de printer:

D ₄	- error (negatieve logica)
D ₃	- printer on
D ₂	- papier op
D ₁	- busy
D ₀	- ackn

tabel 3.

Het strobe signaal op pen 9 (actief laag) wordt geset door het adresseren van poort &H9A. Strobe wordt weer gereset door het adresseren van poort &H9B. Het maakt niet uit of er gelezen of geschreven wordt naar dit adres, nog wat de data is bij een write.

In de bijlage is een voorbeeldprogramma voor deze interface opgenomen. Hiermee is een printerspooler gemaakt, waardoor het te printen stuk in een buffer

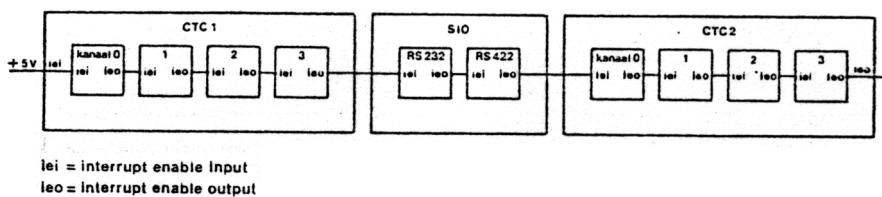
gedumpt wordt. Op interruptbasis wordt deze buffer naar de printer gestuurd. Hierdoor is het mogelijk dat men gewoon door kan gaan met de computer tijdens het printen.

Ook is een testprogramma (in BASIC) opgenomen waardoor karakters naar de printer gestuurd worden: continu hetzelfde karakter of karakters die men intikt vanaf het toetsenbord. Bij dit laatste dient aangetekend te worden dat sommige printers niet beginnen te printen voordat zij een hele regel hebben ontvangen.

5 Interruptstructuur.

Een aantal modules maakt gebruik van interrupts, die alle een verschillende prioriteit hebben. Omdat de processor deze niet zelf kan onderscheiden, is hiervoor gebruik gemaakt van een aantal bouwstenen die dit voor hun rekening nemen. Hiermee wordt een 'Daisy Chain' opgebouwd, in goed Nederlands een ketting. Hoe verder je in deze ketting achteraan zit, hoe lager je prioriteit is. Dit is in figuur 1. schematisch aangegeven.

Daisy Chain structuur



figuur 1.

De meest linkse bouwsteen is permanent aan de voedingsspanning verbonden. Die ingang bepaalt namelijk of de bouwsteen een interrupt mag genereren. Is hij hoog (= logisch 1) dan kan dat. Is die ingang laag, dan moet hij wachten. Wanneer nu zo'n bouwsteen een interrupt wil genereren, kijkt hij of die ingang hoog is of wacht totdat die hoog wordt. Is dit het geval dan zal hij aan het volgende de-

vice (bouwsteen) dat op hem aangesloten is te kennen geven dat die geen interrupt mag geven, door zijn uitgang laag te maken. Hierna zal dan een interrupt naar de processor gegeven worden. Een device dat hoger in prioriteit is, mag hem wel onderbreken, ook al is zijn interrupt nog niet afgehandeld.

De volgende items maken, in afnemende prioriteit, deel uit van de interruptstructuur van het multifunktiebord:

- 1 floppy disk controller
- 2 error detectie floppy controller
- 3 real time clok
- 4 toetsenbord afhandeling
- 5 Seriele interface
- 6-9 Baudrate generator Seriele interface

tabel 4.

Op nummer 5 na worden deze interrupts afgehandeld door twee CTC's (Counter Timer Circuit).

5.1 Afhandeling.

Wanneer een device een interrupt geneert, dan zal deze een byte (vector) op de databus zetten. Dit byte is daar onder softwarebesturing in weggeschreven. Het vormt samen met het I-register van de processor een 16 bits adres. Het I-register het meest significante deel (=hoge byte; meestal &H60), de vector het minst significante deel (=lage byte). Op dit adres kan men in twee bytes wegschrijven wat het beginadres is van de interruptroutine.

Hoe men een CTC als interruptcontroller gebruikt, wordt in de volgende paragraaf uitgelegd.

5.2 Principe CTC.

De CTC is in feite een counter/timer met een interruptstructuur. In timer mode kan men een teller instellen op een waarde tussen 1 en 256. Deze zal steeds verlaagd worden door de systeemclock (2,5 MHZ) gedeeld door 16 of 256, naar believen in te stellen. Wanneer deze bij nul komt, wordt hij opnieuw met de ingestelde waarde geladen en begint het proces weer opnieuw. Ook zal dan, wanneer zo geprogrammeerd, een interrupt gegenereerd worden.

De counter mode werkt bijna hetzelfde. Het verschil is dat de teller steeds verlaagd wordt door een extern signaal. Hierop is geen deelfactor van 16 of 256 van toepassing.

Welnu, op die externe triggeringangen heeft men de interruptlijnen van een aantal modules aangesloten, zoals de floppycontroller, toetsenbord routine e.d. Wanneer men de counters met 1 laadt, dan zullen zij bij een triggersignaal van een andere module op nul komen: er wordt een interrupt gegenereerd (mits zo geprogrammeerd).

Dit triggeren kan zowel op de positieve als negatieve flank van een signaal gebeuren, en kan men in een van de commandoregisters van de CTC opgeven.

Elke CTC bestaat uit vier kanalen. Hierdoor zijn de nummers 1 tot en met 4 op een CTC aangesloten. Intern heeft 1 de hoogste, 4 de laagste prioriteit.

Twee kanalen van de tweede CTC worden gebruikt als baudrategenerator voor de seriële interface. De twee andere kanalen zijn nog vrij en kunnen bijvoorbeeld gebruikt worden voor het opwekken van een periodiek interruptsignaal.

Eigenlijk ligt de functie van deze bouwstenen dus al vast. Men kan ze echter wel als timer voor eigen doeleinden gebruiken, maar hierdoor verliest men de functie die dat kanaal oorspronkelijk had: interruptdevice voor floppy controller, toetsenbord e.d.

In een aparte paragraaf zal nog uitgelegd worden hoe men deze bouwstenen kan programmeren.

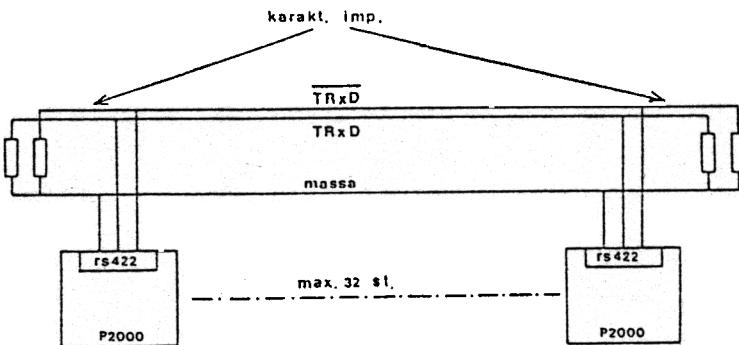
6 Seriele Interface.

Voor de seriële interface is de Z80/SIO-0 gebruikt. Deze bevat twee kanalen die simultaan kunnen werken. Rond kanaal A is een RS232 interface gebouwd, met kanaal B is een netwerkinterface gerealiceerd (RS422). De RS232 interface beschikt over aparte lijnen voor zenden, ontvangen en besturings-(handshake)signalen. De netwerkinterface heeft maar een lijn, die zowel voor zenden als ontvangen gebruikt wordt. Hiervoor gebruikt men meestal een 2-adrig afgeschermde kabel. Deze kabel dient aan weerszijden afgesloten te worden met zijn karakteristieke impedantie. (zie figuur 2.)

De RS232-interface kan tegelijk op verschillende snelheden voor zenden en ontvangen werken, bijvoorbeeld op 1200/75 baud zoals bij Videotex-syste- men.

De twee interfaces zijn samen met de centronics interface op een connector aangesloten. In bijlage 1. is hiervan een tekening opgenomen.

Aansluiting op de netwerkinterface.



figuur 2

De seriële interfaces zijn hardwarematig zo geconfigureerd dat alleen gebruik gemaakt kan worden van asynchrone transmissie.

6.1 Besturing interface.

Elke interface, dus ook deze, moet geinitialiseerd worden. Hierdoor wordt aan hem meegedeeld op welke manier het verzenden en ontvangen van data moet gebeuren. Hiervoor heeft elk kanaal een besturingsregister en een dataregister.

Deze zijn op de volgende adressen ondergebracht:

RS232 interface:

&H84 - dataregister kanaal A
&H85 - besturings/status-register van kanaal A

RS422 interface:

&H86 - dataregister kanaal B
&H87 - besturings/status-register van kanaal B

tabel 5

Beide besturingsregisters bestaan eigenlijk uit meerdere registers: 8 waarin men kan schrijven (WR0-WR7) en 3 waaruit men kan lezen (RR0-RR2). Normaal heeft men bij lezen of schrijven toegang tot register 0. Wil men lezen of schrijven in een andere register, dan geeft men in Write Register 0 op welke dit moet zijn. De volgende keer dat men dan het commando/status-register adresseert, heeft men toegang tot het zojuist opgegeven register.

WR0 - registerpointers, resetcommando's
WR1 - interrupt mode
WR2 - interruptvector (alleen in kanaal B)
WR3 - ontvangstparameters
WR4 - zendparameters
WR5 - modus definitie
WR6 - niet van toepassing voor asynchrone transmissie
WR7 - ,

RR0 - zender/ontvanger bufferstatus,
interruptstatus
RR1 - ontvangerstatus
RR2 - interruptvector (alleen in kanaal
B)

tabel 6.

In de volgende paragrafen zal uitleg worden hoe men hiermee moet omgaan. Hoewel hierin soms staat dat men een gegeven bit in een register moet setten of resetten, betekent dit niet dat geen rekening gehouden hoeft te worden met de andere bits van dat register. Hiervoor kan men het beste figuur 3. raadplegen. Hierin staat nog eens duidelijk vermeld wat de functie is van alle registers.

6.2 I/O mogelijkheden.

In de bestaande configuratie kan men op twee manieren met de interface communiceren:

in interrupt mode
via polling

6.3 Interrupts.

Wil men op interruptbasis werken, dan krijgt men te maken met de Daisy Chain structuur die al eerder besproken is. De interruptvector, die voor beide kanalen geldt, kan men in register 2 van kanaal B opgeven. Via register WR1 geeft men op dat men in interrupt mode wil werken. Kanaal A heeft een hogere prioriteit dan

kanaal B. Dit betekent dat wanneer de software bezig is met het afhandelen van een interrupt van kanaal B, toch een interrupt van kanaal A op kan treden. Deze kan dan eerst afgehandeld worden. Men kan de interface zo programmeren dat interrupts optreden na:

elk ontvangen karakter
alleen het eerste karakter van een
boodschap
een speciale conditie

De laatste mogelijkheid is alleen toepasbaar wanneer men ook een van de twee andere methoden selecteert. Hierdoor kunnen bijvoorbeeld transmissiefouten gesigneerd worden zoals pariteitsfouten en een verbroken verbinding. De oorzaak van fouten kan men nagaan in RR0 en RR1.

6.4 Polling.

In deze mode kunnen geen interrupts optreden. Wil men daarom weten of een nieuw karakter is binnengekomen, een karakter al verzonden is of nagaan of er transmissiefouten zijn opgetreden, dan moet men steeds controleren wat de status van RR0 en RR1 is.

6.5 Baudrate.

De transmissiesnelheid kan men zelf bepalen. Hiervoor wordt het Counter Timer Circuit nummer 2 (CTC2) gebruikt. Deze wekt de frequentie op die van be-

lang is voor de baudrate. Kanaal 0 hiervan bepaalt de ontvangstsnelheid, kanaal 1 de zendsnelheid van de RS232 interface. Voor de netwerkinterface (RS422) wordt die snelheid bepaald door de systeemclock (2,5 MHz). Door een jumper om te leggen, kan men hem ook op kanaal 2 van CTC2 aansluiten. De baudrates worden afgeleid van de waarden en deeltallen die de respectievelijke kanalen bezitten. Deze deeltallen zijn: 1/1, 1/16, 1/32 en 1/64. Het deeltal 1/1 is voor ontvangen in de bestaande configuratie niet mogelijk.

De kanalen van CTC2 die voor het bepalen van de baudrate gebruikt worden, mogen niet in interruptmode geprogrammeerd worden.

De baudrates zijn afgeleid van de systeemclock, die een periodetijd van 0,4 us heeft. Om aan de baudrate te komen, moet men deze vermenigvuldigen met het deeltal van de timer/counter en dat van de SIO. 1 gedeeld door dit getal is de baudrate *).

In tabel 7. zijn we er van uitgegaan dat de SIO per bit 16 keer sampeld. De getallen achter de baudrate zijn de inhouden van de counter/timer.

*) In counter mode geldt voor de kanalen ① tot en met 2 van CTC2 een extra deelfactor van 2 voor de systeemclock.

Baudrates SIO.

1		16	*)
19200: 4		4800: 2	**)
9600: 8		2400: 4	
4800: 16		1200: 8	
2400: 33 <i>32</i>		600: 16	
1200: 65 <i>64</i>		300: 33	
600: 130 <i>128</i>		150: 65	
		110: 89	
		75: 130	
		50: 195	

*) Deeltallen CTC.

Voor het deeltal 1 moet de CTC in countermode geprogrammeerd worden, voor deeltal 16 in timermode.

**) resp. baudrate en te programmeren inhoud van teller. Getallen zijn in decimale notatie.

tabel 7.

De baudrate van de netwerkinterface (RS422) ligt vast en wordt alleen bepaald door hoeveel sampels men per bit wil nemen: 1, 16, 32 of 64. Doordat 1 niet mogelijk is, zijn de volgende baudrates instelbaar: 156250, 78125 en 39062.5 baud.

Wanneer men een jumper op het bord omllegt, kunnen dezelfde baudrates als in de tabel ingesteld worden via kanaal 2 van CTC2. Door dit kanaal wordt zowel de snelheid van de zender als de ontvanger bepaald.

Wanneer bijvoorbeeld een modem wordt gebruikt, maakt men meestal gebruik van handshaking. Hiervoor zijn voor kanaal A vier lijnen naar buiten gevoerd (RTSA: Request To Send; DTRA: Data Terminal Ready; CTSA: Clear To Send; DCDA: Data Carrier Detect;).

In de meeste asynchrone applicaties worden de RTSA-, DTRA-lijnen en de Auto Enables mogelijkheid alleen gebruikt tijdens de initialisatie. Hierna volgt er meestal een continue datastroombaan.

Een verandering van deze lijnen zal in interruptmode een interrupt veroorzaken. In RRO kan men dan de waarde van de DCD- en CTS-pin terugvinden.

Wanneer de Auto Enables mogelijkheid niet gebruikt wordt, kunnen de RTS- en CTSA-lijnen vrij gebruikt worden. Hun waarde kan afgelezen worden in RRO. De uitgangssignalen DTRA en RTSA zullen dezelfde waarde aannemen als bit 7 en 5 van WR5.

6.6 Besturing netwerkinterface.

Bij de netwerkinterface wordt dezelfde lijn gebruikt voor zenden en ontvangen. Welk station er mag zenden of ontvangen, moet men vastleggen in een protocol. Er mag maar 1 station tegelijk zenden.

Hierom zal er steeds omgeschakeld moeten worden tussen zenden en ontvangen. Hier- voor schakelt men de Auto Enables mode uit. Door bit 1 en bit 7 van WR5 (kanaal B) hoog te maken, kan men ontvangen. Door bit 7 laag te maken, staat hij op zenden. Laat men in dit geval bit 1 hoog, dan zullen de verzonden bytes ook

in de eigen ontvanger terecht komen. Wil men dit niet, dan moet men bij zenden ook bit 1 laag maken.

6.7 Ontvangen.

Het dataregister bevat drie buffers. Dit betekent dat wanneer een karakter ontvangen is, maar nog niet gelezen door de processor, toch 2 nieuwe karakters ontvangen kunnen worden. Door deze drie buffers zal bij het binnengaan van een vierde karakter zonder dat de vorige gelezen zijn, het derde overschreven worden. Mocht dit gebeuren, dan zal dit gemeld worden in RRO. Op dit soort fouten dient men elke keer als een karakter door de processor wordt binnengehaald te controleren.

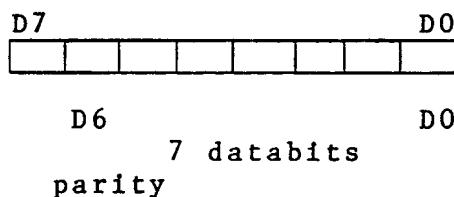
Is een karakter eenmaal uit het dataregister gelezen, dan is het niet meer mogelijk het nog een keer te lezen.

Elk karakter dat verzonden wordt, heeft een formaat. Dit moet men zelf opgeven aan de interface. Het is noodzakelijk dat zender en ontvanger hetzelfde formaat gebruiken. Hierin dienen de volgende zaken afgesproken te worden:

- aantal bits per byte (5, 6, 7 of 8)
- aantal stopbits (1, 1.5 of 2)
- pariteit (geen, even of oneven)
- baudrate
- handshaking

De ontvanger zal niet eerder gaan werken voordat men hem enabled. Dit doet men door het zetten van het Receive Enable

bit (WR3, bit 0). Wil men gebruik maken van handshaking, dan moet men het Auto Enables bit setten (WR3, bit 5). Men kan controleren of een karakter binnengekomen is door het lezen van RRO. Is dit het geval, dan kan men dit karakter uitlezen via het dtatregister van gewenste kanaal. Indien het aantal bits/karakter niet 8 is, dan bevatten de niet gebruikte databits informatie over eventueel het parity bit en de stopbit(s). Bij voorbeeld viditel; 7 databits, even pariteit, 1 stopbit, levert het volgende resultaat op bij het lezen van een karakter:



Bit 7 moet in deze toepassing door de software onderdrukt worden.

WRITE REGISTER 0

```
0 0 0 NULL CODE
0 0 1 SEND ABORT (SOLC)
0 1 0 RESET EXT STATUS INTERRUPTS
0 1 1 CHANNEL RESET
1 0 0 ENABLE INT ON NEXT Rx CHARACTER
1 0 1 RESET TxINT PENDING
1 1 0 ERROR RESET
1 1 1 RETURN FROM INT (CM-A ONLY)
```

D 0 NULL CODE
D 1 RESET As CRC CHECKER
I 0 RESET To CRC GENERATOR
I 1 RESET To UNDERFLOW LATCH

WRITE REGISTER 1

WRITE REGISTER 2 (CHANNEL B ONLY)

WRITER REGISTER 3

The diagram illustrates the timing sequence for the 8250 serial port. It shows the relationship between the Rx ENABLE signal, SYNC CHARACTER LOAD INHIBIT, ADDRESS SEARCH MODE (SDLC), Rx CRC ENABLE, ENTER HUNT PHASE, and AUTO ENABLES. The Rx ENABLE signal is asserted at the start of a character. The SYNC CHARACTER LOAD INHIBIT signal is asserted during the first character of a frame. The ADDRESS SEARCH MODE (SDLC) signal is asserted during the second character of a frame. The Rx CRC ENABLE signal is asserted during the third character of a frame. The ENTER HUNT PHASE signal is asserted during the fourth character of a frame. The AUTO ENABLES signal is asserted during the fifth character of a frame.

WRITE REGISTER 4

WRITE REGISTER S

G7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	0	0	Is CRC ENABLE
0	0	0	0	0	0	0	1	RTS
0	0	0	0	0	0	1	0	SOLID/CRC-16
1	0	0	0	0	0	0	1	Tx ENABLE
1	1	0	0	0	0	0	1	SEND BREAK
DTR								
0	0	0	0	0	0	0	0	Tx: 5 BITS/CHARACTER
0	0	0	0	0	0	1	1	Tx: 7 BITS/CHARACTER
1	0	0	0	0	0	0	1	Tx: 6 BITS/CHARACTER
1	1	0	0	0	0	0	1	Tx: 8 BITS/CHARACTER

WRITE REGISTER 6

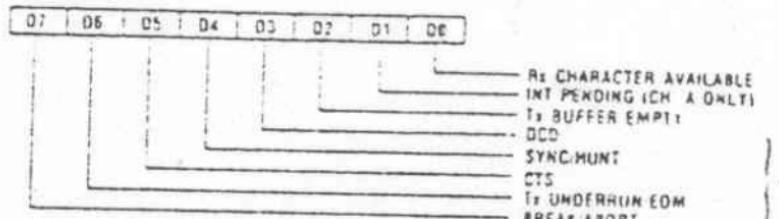
The diagram illustrates a 10-bit frame structure. It consists of two main sections: a header and a payload. The header section contains seven bits labeled "SYNC BIT 0" through "SYNC BIT 6". The payload section contains three bits labeled "D9" through "D0". A bracket groups the first eight bits (Sync Bits 0-7) as the "Sync Word".

- ALSO SEE ADDRESS FIELD

WRITE REGISTER 7

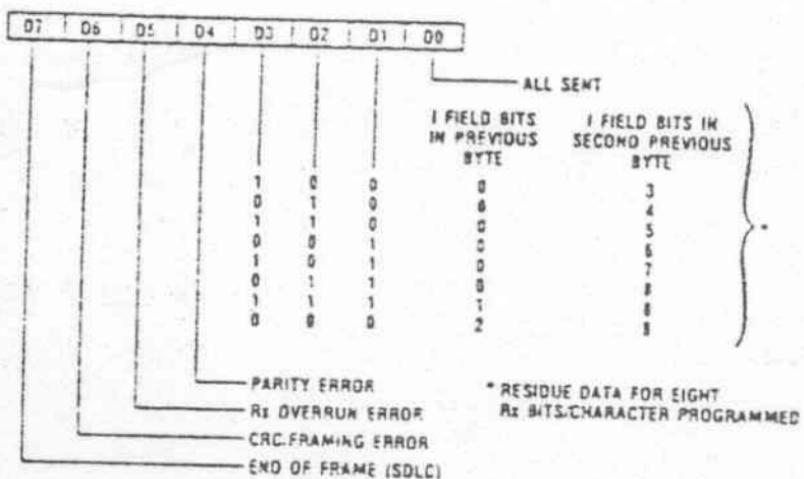
-FOR SDLC IT MUST BE PROGRAMMED
TO "01111110" FOR FLAG RECOGNITION

READ REGISTER 0



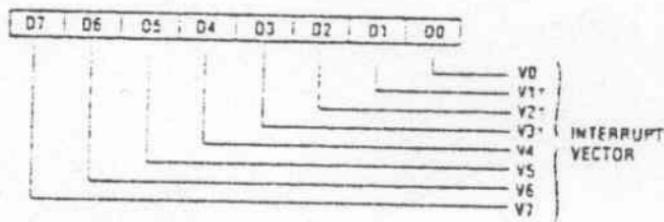
* USED WITH "EXTERNAL STATUS INTERRUPT" MODE

READ REGISTER 1†



† USED WITH SPECIAL RECEIVE CONDITION MODE

READ REGISTER 2



*VARIABLE IF "STATUS AFFECTS VECTOR" IS PROGRAMMED

6.8 Zenden

Voor het zenden moeten soortgelijke afspraken gemaakt worden als voor ontvangen. De SIO voegt zelf informatie toe volgens de geprogrammeerde afspraken, zoals start- en stopbits, pariteitbit e.d.

Wanneer men het Transmit interrupt enable bit in register WR1 set, zal elke keer dat de zendbuffer leeg is, een interrupt gegenereerd worden naar de vector bepaald door WR2 en bit 2 van WR1. Bij initialisatie moet men eerst WR4 laden voordat WR1, WR2 en WR5 geladen mogen worden.

In de Auto Enables mode kunnen de zendbuffers wel geladen worden wanneer CTS nog laag is. De data zal pas verzonden worden wanneer de CTS-lijn door het modem hoog gemaakt wordt.

Elke keer dat men een byte wil verzenden (door het naar het dataregister te sturen) dient men te controleren of de zendbuffer leeg is. Doet men dit niet, dan gaat er informatie verloren.

6.9 Commando/Status-registers.

In figuur 3. is de betekenis van elk bit van alle registers aangegeven. Wanneer men een actie wil uitvoeren, kan men hierin terugvinden wat de waarde van elk bitje moet zijn. Niet alle modes zijn mogelijk, zodat een aantal bits niet gebruikt mogen worden. Daarom geeft het onderstaande hierover nog een summiere toelichting.

Normaal heeft men bij het lezen of

schrijven toegang tot register 0. Wil men een ander register gebruiken, dan moet men het nummer van het gewenste register wegschrijven in WRO en de overige bits (bit 3 - 7) nul maken.

De afkorting Tx en Rx in figuur 3. staan voor Transmitter (zender) resp. Receiver (ontvanger).

6.10 Write Registers.

Write Register 0.

Bit 6 en 7 zijn van geen belang. De Sio kan alleen in asynchrone mode werken. De Channel Reset heeft hetzelfde effect als een hardware reset, hij werkt echter alleen op het geselecteerde kanaal. Return From Interrupt heeft hetzelfde effect als een RETI commando, en is van toepassing in niet-Z80 systemen.

Write register 1.

bit 5, 6 en 7 dienen 0 te zijn. Deze mogelijkheid is namelijk niet geïmplementeerd.

Met bit 0, 1, 3 en 4 geeft men aan of

men de interruptmode wil gebruiken en zoja, welke.

Bit 2 kan alleen via kanaal B opgegeven worden, doordat dit kanaal alle interruptvector-aangelegenheden regelt. Is de waarde van dit bit 0, dan zal bij een interrupt, mits geenabled, de vector geprogrammeerd in WR2 op de bus gezet worden. Maakt men dit bit 1, dan worden bit 1, 2 en 3 bepaald volgens de fig.4. Hierdoor wordt bij een interrupt, afhankelijk van de oorzaak, naar verschillende afhandelingsroutines gesprongen.

Interrupt vector SIO; bit 1, 2 en 3

	V₃	V₂	V₁	
Ch B	0	0	0	Ch B Transmit Buffer Empty
	0	0	1	Ch B External/Status Change
	0	1	0	Ch B Receive Character Available
	0	1	1	Ch B Special Receive Condition*
Ch A	0	0	0	Ch A Transmit Buffer Empty
	0	0	1	Ch A External/Status Change
	1	0	0	Ch A Receive Character Available
	1	0	1	Ch A Special Receive Condition*

*Special Receive Conditions: Parity Error, Rx Overrun Error, Framing Error, End Of Frame (SDLC).

figuur 4

Write Register 3.

Bit 2, 3 en 4 dienen 0 te zijn. Deze zijn alleen voor synchrone communicatie.

Write Register 4.

Bit 4 en 5 zijn niet van belang. De combinatie 00 voor bit 6 en 7 is bij ontvangen niet mogelijk. De waarde van die bits bepaalt de baudrate in samen-

werking met CTC2.

Write Register 5.

Bit 2 is niet van belang.

Write Register 6 en 7 hebben geen betekenis. Ze zijn bedoeld voor het SDLC-protocol, dat op het multifunktiebord niet tot de mogelijkheden behoort.

6.11 Read Registers.

Read Register 0.

Bit 0 (Receiver Character Available) geeft aan dat er karakter ontvangen is dat gelezen kan worden.

Bit 1 (interrupt pending) geeft aan dat er een interrupt gegenereerd is die nog niet afgehandeld is.

Bit 2 (Transmit Buffer Empty) betekent dat de zendbuffer leeg is.

Bit 3 en 5 (DCD en CTS; alleen kanaal A) geven de waarde aan van hun lijnen op de connector.

Bit 4 en 6 hebben geen betekenis.

Bit 7 (Break/Abort) geeft aan dat er een break gedetecteerd is: de ontvangstlijn op de connector is logisch nul geworden.

Read Register 1.

Bit 1, 2, 3 en 7 zijn in asynchrone mode niet van belang.

Bit 0 (All Sent): Alle zendbuffers zijn leeg.

Read Register 2.

Dit register bevat de interruptvector van WR2. Wanneer bit 2 van WR1 is geset, worden bit 1, 2 en 3 bepaald door de interrupt met de hoogste prioriteit die nog niet afgehandeld is. Zijn alle interrupts afgehandeld dan zullen zij resp. de waarden 1, 1 en 0 hebben.

6.12 Oorzaak transmissiefouten.

Er kunnen een aantal transmissiefouten gedetecteerd worden. Hierop dient men te testen wanneer een nieuw karakter is ontvangen. Deze fouten zijn:

Pariteitsfout (RR1, bit 4). Dit kan het gevolg zijn van een storing, of zender en ontvanger gebruiken niet dezelfde pariteit

- Overrun error (RR1, bit 5). Er is een vierde karakter binnengekomen zonder dat er een karakter is uitgelezen.
Mogelijke oorzaak: verkeerde mode (interrupt/polling) of een device met een hogere prioriteit is bezig met interruptafhandeling, waardoor karakters niet tijdig uitgelezen worden.

- Framing error (RR1, bit 6). Er is (zijn) geen (genoeg) stopbit(s) gedetecteerd.
Dit kan als oorzaak hebben dat zender en ontvanger niet hetzelfde aantal databits per karakter hebben. Een storing behoort ook tot de mogelijkheden.

Break Detect (RR0, bit 7). Dit hoeft geen fout te zijn, daar het ook door de zender onder softwarebesturing kan optreden. Het kan ook het gevolg zijn van een verbroken verbinding.

Men kan dit soort fouten via polling controleren. Ook is het mogelijk een interrupt te laten genereren. Hiervoor moet bit 3 en/of bit 4 van WR1 geset worden.

Elk bit dat voor foutcondities gebruikt wordt, wordt pas gereset na het Error Reset commando (WR0 bit 5 en 4). Dit geld niet voor de Framing error. Dit bit wordt gereset nadat het byte gelezen is waarin de fout optrad.

6.13 Voorbeeld programma's.

```
;  
; printeroutine voor de RS232 interface  
; van het Multifunktiebord  
;  
; baudrate: 1200 Bd  
; geen protocol  
; geen interrupts  
;  
$init      push af  
;  
    ld      a,00011000t      reset interface  
    out   (85),a
```

```

ld    a,00010101t      selecteer WR5
out   (85),a
ld    a,11101010t      8 databits/ka-
rakter
                           set DTR en RTS
                           enable transmitter
out   (85),a

ld    a,00010100t      selecteer WR4
out   (85),a
ld    a,010001100t     1/16 clock
                           2 stopbits
                           geen pariteit
out   (85),a

;
; initialiseer CTC
;
ld    a,01000101t      geen interrupts
                           counter mode
                           prepareer voor
                           tijdconstante
out   (81),a
ld    a,82                82(hex.)=130(dec.)
                           =1200 Baud
out   (81),a

pop   af
ret

;
; routine voor uitvoer van een karakter
; register a bevat te printen karakter
;
$pprt push af           karakter naar
                           stack

$prrt
in    a,(85)             repeat
                           lees status-
                           register 0
                           test zendbuf-
                           fer
jr    z,$prrt            until buffer leeg

```

```
;  
pop af  
out (84),a  
ret
```

haal karakter
van stack
karakter naar da-
taregister RS232

7 Counter Timer Circuit.

Zoals reeds eerder is vermeld, zijn er twee van deze ic's op het multifunktiebord in gebruik. Elk bezit vier kanalen. CTC1 wordt volledig gebruikt voor het genereren van interrupts. Van CTC2 zijn kanaal 0 en 1 in gebruik voor het bepalen van de baudrate van de RS232 interface. Eventueel kan kanaal 2 gebruikt worden voor het bepalen van de baudrate van de netwerkinterface. Hiervoor moet een jumpertje omgelegd worden.

7.1 Werking.

De timer/counter bevat een teller (Time Constant Register) waarin een getal (initiele waarde) tussen 1 en 256 geladen kan worden. &H00 wordt beschouwd als 256. De timer/counter zal dit steeds aflagen, totdat hij bij nul komt. Evenueel wordt dan een interrupt gegenereerd (wanneer zo geprogrammeerd) en zal de initiele waarde weer geladen worden, waarna het proces zich herhaald. Dit aflagen hangt af van de modus en een deeltal. In timer mode wordt dit aflagen bepaald door de systeemclock en de waarde van dit deeltal: 16 of 256. De periodesetijd is daardoor $16 \cdot 0.4 = 6,4$ us of $256 \cdot 0.40 = 102,4$ us. In counter mode wordt het aflagen bepaald door een extern signaal: het triggersignaal. Naar keuze gebeurt dit op de negatieve of positieve flank. Hierop is geen deeltal van toepassing *).

*) In counter mode geldt voor de kanalen 0 tot en met 2 van CTC2 een extra deelfactor van 2 voor de systeemclock.

7.2 Channel Control Register

Voordat een kanaal aan een countactie kan beginnen, moet een Time Constant Data Word naar het Channel Control Register geschreven worden.

Adressen Channel Control Registers

CTC1

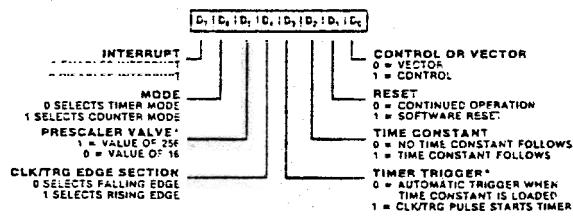
kanaal 0:	poort	&H88
,,	1:	,, &H89
,,	2:	,, &H8A
,,	3:	,, &H8B

CTC2

kanaal 0:	poort	&H80
,,	1:	,, &H81
,,	2:	,, &H82
,,	3:	,, &H83

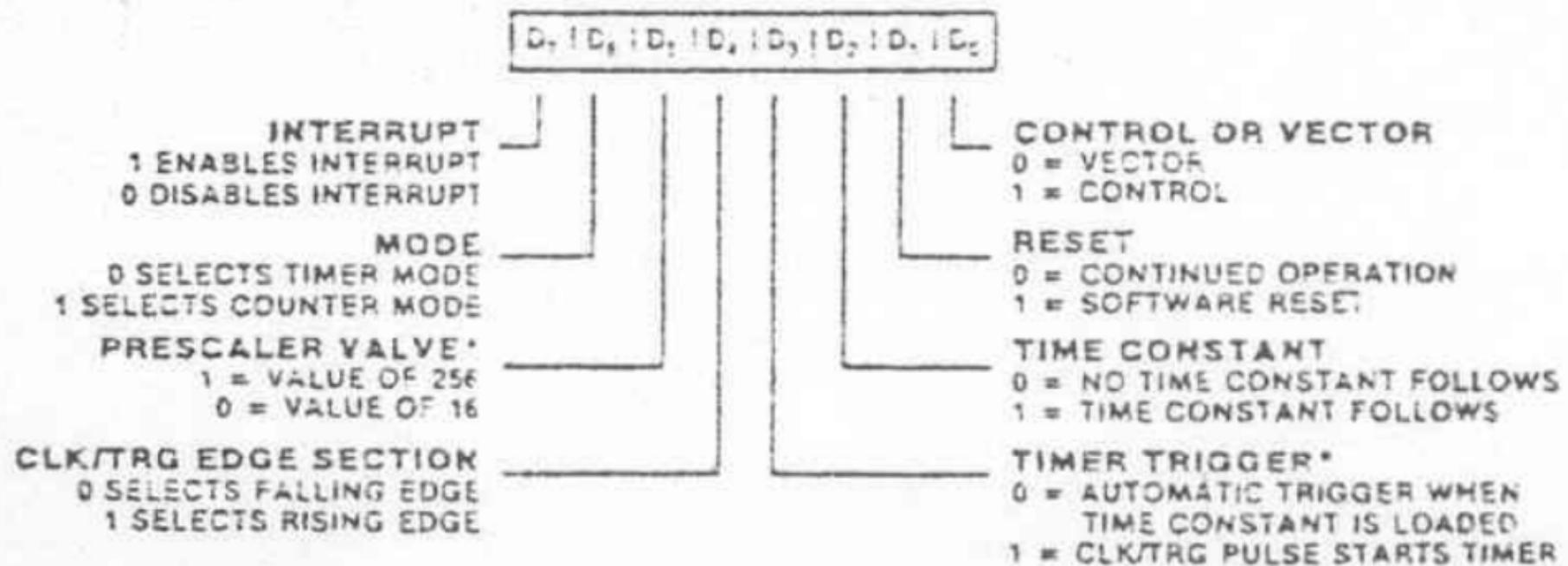
tabel 8.

Channel Control Register.



figuur 5.

Channel Control Register.



*TIMER MODE ONLY

figuur 5.

7.3 Inhoud Channel Control Register.

D₇ (Interrupt Enable). Wanneer men dit bit 1 maakt, zal elke keer dat het kanaal de waarde 0 bereikt, een interrupt gegenereerd worden.

D₆ (Counter Mode Selected). Is dit bit 1 (Counter Mode), dan zal de counter na elke clock flank verlaagd worden. Is dit bit 0 (Timer Mode), dan is hierop nog een deeltal van toepassing, bepaald door D₅.

D₅ = 1: Deeltal 256.
0: Deeltal 16.

Deze gelden alleen in timermode.

D₄. In countermode zal de teller verlaagd worden op de negatieve flank van de triggeringang wanneer dit bit 0 is en op de postieve flank wanneer dit bit 1 is.

D₃ (Timer mode). Wanneer dit bit 0 is zal de timer onmiddelijk starten wanneer het Time Constant Register wordt geladen. Is dit bit 1, dan zal de teller pas starten wanneer de triggeringang geactiveerd wordt.

D₂ (Load Time Constant). Wanneer dit bit 1 is, zal het volgende byte dat naar dit kanaal geschreven wordt als tijdconstante beschouwd worden en in het Time Constant Register worden geladen. Wanneer op dat moment de counter/timer reeds in werking is, zal het aftellen gewoon doorgaan. Wanneer hij bij nul aangekomen is zal bij het begin van de volgende

cyclus de nieuwe tijdconstante gebruikt worden.

D₁ (Reset). Wanneer dit bit 1 is, dan de counter/timer onmiddelijk ophouden. Wanneer zowel D₁ als D₂ 1 zijn, zal de counter/timer na het laden van de tijdconstante gelijk beginnen.

D₀ = 1: D₁-D₇ hebben de functie zoals hierboven behandeld.
= 0: D₃-D₇ gelden voor alle kanalen van de geadresseerde CTC. D₂ en D₁ worden bepaald door tabel 9. terwijl het minst significante bit van een interruptvector altijd 0 is. (Een vector bestaat immers uit 2 bytes)

kanaal	D ₂	D ₁
0	0	0
1	0	1
2	1	0
3	1	1

tabel 9.

7.4 Gebruikersmogelijkheden.

CTC1.

CTC1 wordt gebruikt als interruptcontroller. De triggeringen zijn verbonden met de interruptlijnen van een aantal devices:

kanaal 0: floppycontroller
kanaal 1: errordetectie floppycontroller
kanaal 2: real time clock
kanaal 3: Scannen toetsenbord

Hiertoe worden de kanalen in countermode geprogrammeerd. De tellers worden met 1 geladen. Wanneer een interrupt van een device optreedt, komen deze tellers op nul. Door de CTC in interrupt mode te programmeren wordt dan een interrupt gegenereerd.

Wanneer men de BASIC NL gebruikt, worden kanaal 0, 1 en 3 door de BASIC gevuld, dus ook het Interrupt Vector Register. Deze vector wijst dan naar &H6020.

Adres	Bevat beginadres routine voor
-----+-----	
&H6020-21	floppy controller
&H6022-23	errordetectie floppycontroller
&H6024-25	real time clock
&H6026-27	Scannen toetsenbord

tabel 10.

Op adres &H6024-25 kan men dus een adres wegschrijven van een (eigen) routine waarin de real time clock wordt gebruikt. In de bijlage is een voorbeeldprogramma opgenomen dat constant de tijd op het beeldscherm displayed.

CTC2.

Kanaal 0 bepaalt de baudrate voor ontvangen van de RS232 interface, kanaal 1 de baudrate voor zenden van deze interface. Wanneer een jumper op het bord wordt omgelegd, wordt de baudrate voor zowel ontvangen als zenden van de RS422 interface door kanaal 2 bepaald.
De triggeringangen van kanaal 0-2 zijn

op de systeem clock (2,5 MHz.) aangesloten.

Wel geldt voor deze ingangen een extra deelfactor van 2. De triggeringang van kanaal 3 is aangesloten op de uitgang van kanaal 2. Op deze uitgang ontstaat een korte puls wanneer kanaal 2 de waarde nul bereikt. Hierdoor kan kanaal 3 op twee manieren gebruikt worden:

In counter mode: Het aflagen van de teller van dit kanaal gebeurt nu elke keer wanneer kanaal 2 de waarde nul bereikt.

In timermode: Wanneer D₃ 0 is wordt de systeem clock gebruikt. Is dit bit echter gelijk aan 1, dan zal de teller verlaagd worden met de frequentie van de systeem clock, doch dit zal pas starten wanneer hij getriggerd wordt (Kanaal 2 bereikt de waarde 0).

Door deze configuratie kan men op interrupt basis een tweede programma (machine taal) laten runnen.

Men kan nagaan wat de waarde van de teller is van een bepaald kanaal door een in-instructie uit te voeren naar het betreffende kanaal.

In tabel 7. kan men een lijst vinden met waarden die in de teller geladen moeten worden voor het bepalen van de baud-rates.

Voorbeeldprogramma.

```
;  
; initialisatie routine voor kanaal 2  
; van CTC2  
; tabel interruptroutine-entries  
; moet ondergebracht worden op &Hfff0  
; &Hfff7  
;  
    push af  
    ld   a,ff  
    ld   i,a          i := hoge byte  
                      interruptvector  
    ld   a,f0          lage byte inter-  
                      ruptvector naar  
                      CTC2  
    (82 ,a  
;  
    ld   a,10100111      enable inter-  
                      rups  
                      timer mode  
                      prescaler = 256  
                      reset teller  
    out  (82),a  
    ld   a,c3          a := 195 (in-  
                      houd teller)  
    (82),a  
  
    pop  af  
    ret
```

8 Real Time Clock.

Het extension board is uitgerust met een Real Time Clock met battery back up. Hierdoor gaat het geheugen van dit ic niet verloren wanneer de computer uitgeschakeld wordt. Dit geheugen bestaat uit 64 bytes. De eerste 14 hiervan worden gebruikt voor het bijhouden van de tijd, datum en alarm. De overige 50 bytes kunnen vrij gebruikt worden, om bijvoorbeeld data op te slaan die men bij de volgende keer opstarten gelijk beschikbaar wil hebben.

Wanneer de battery backup te ver leeg raakt, wordt deze afgeschakeld. Hierdoor wordt schade aan de battery voorkomen.

Wanneer de computer de volgende keer opgestart wordt, kan men testen of hierdoor informatie verloren is.

De battery backup wordt automatisch opgeladen wanneer de computer op het lichtnet aangesloten is.

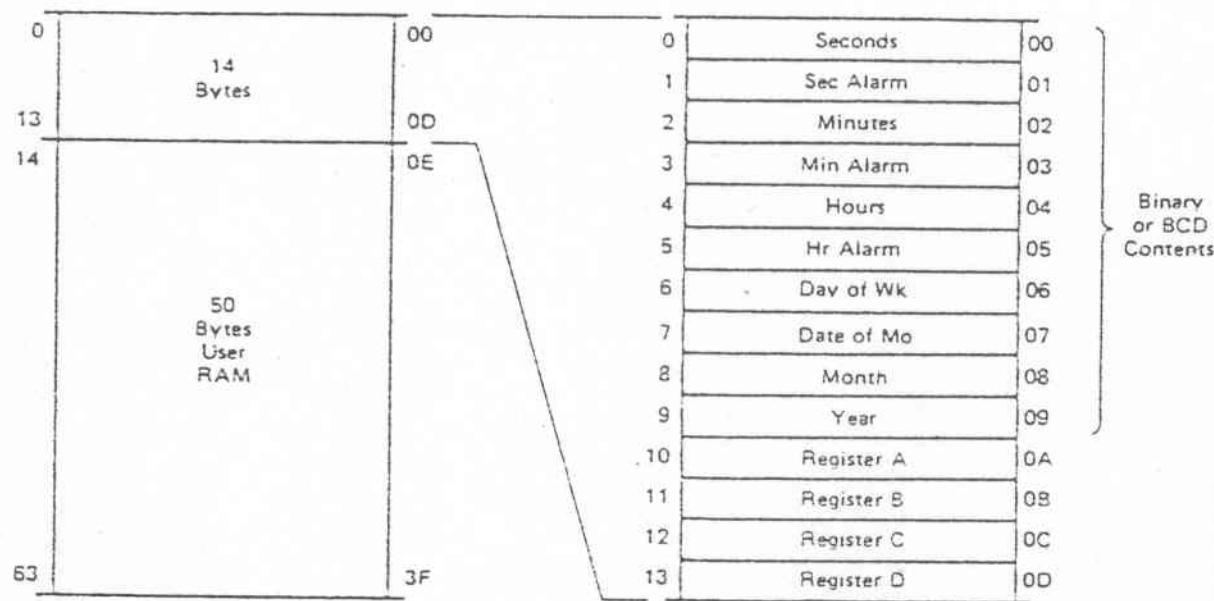
De tijd/datum/alarm mogelijkheden kunnen softwarematig a gevraagd worden. Dit kan eventueel ook in een interruptroutine gebeuren. Deze interrupts kunnen op drie manieren gegenereerd worden:

- bij bereiken van een alarmtijd, en eenmaal per seconde tot eenmaal per dag.
- periodiek vaneen maal per 30.5 us tot 500 ms
- na afloop van een clock update cyclus (eenmaal per seconde).

Voor het geven van interrupts is de RTC via kanaal 2 van CTC1 in een Daisy Chain opgenomen. Hierdoor zal een interrupt van de RTC uitgesteld worden wanneer de

interruptroutine van een device met een hogere prioriteit nog niet afgehandeld is.

Adres map.



figuur 6.

Time, Calendar, and Alarm Data Modes

Address Location	Function	Decimal Range	Range		Example*	
			Binary Data Mode	BCD Data Mode	Binary Data Mode	BCD Data Mode
0	Seconds	0~59	\$00~\$3E	\$00~\$59	15	21
1	Seconds Alarm	0~59	\$00~\$3E	\$00~\$59	15	21
2	Minutes	0~59	\$00~\$3E	\$00~\$59	3A	58
3	Minutes Alarm	0~59	\$00~\$3E	\$00~\$59	3A	58
4	Hours (12 Hour Mode)	1~12	\$01~\$0C (AM) and \$81~\$8C (PM)	\$01~\$12 (AM) and \$81~\$92 (PM)	05	05
	Hours (24 Hour Mode)	0~23	\$00~\$17	\$00~\$23	05	05
5	Hours Alarm (12 Hour Mode)	1~12	\$01~\$0C (AM) and \$81~\$8C (PM)	\$01~\$12 (AM) and \$81~\$92 (PM)	05	05
	Hours Alarm (24 Hour Mode)	0~23	\$00~\$17	\$00~\$23	05	05
6	Day of the Week Sunday = 1	1~7	\$01~\$07	\$01~\$07	05	05
7	Day of the Month	1~31	\$01~\$1F	\$01~\$31	0F	15
8	Month	1~12	\$01~\$0C	\$01~\$12	02	02
9	Year	0~99	\$00~\$63	\$00~\$99	4F	79

* Example: 5:58:21 Thursday 15th February 1979

figuur 7.

Men heeft toegang tot deze registers door in de eerste slag het adres (6 bits) van het gewenste register naar poort &H9C te schrijven en in de volgende slag naar poort &H9D de data weg te schrijven of uit te lezen. De eerste 10 registers worden gebruikt voor het programmeren van het alarm, tijd en datum. Register &H0A tot &H0C zijn statusregisters, terwijl de overige vrij ter beschikking van de gebruiker staan. Alarm, tijd en datum moeten in BCD of binaire notatie ingevoerd worden, afhankelijk van de opgegeven modus.

8.1 Update cyclus.

Eenmaal per seconde wordt alle data (tijd/datum/alarm) gecheckt en bijgesteld. Tijdens deze periode, die ca 2 ms duurt, is data die gelezen wordt onbetrouwbaar. Wanneer een programma random naar deze data kijkt, zal dit eens in de 504 maal optreden. Men kan echter controleren of het ic in zo'n cyclus verkeerd door bit 7 van register A (Update in Progres) te testen.

De RTC heeft vier commando/status registers die ook tijdens een update cyclus toegankelijk zijn. In al deze registers kan zowel gelezen als geschreven worden, tenzij anders vermeld.

8.2 Status/Commando registers.

Register A(&H0A)

bit 7: Update In Progres. Wanneer dit bit 1 is, is de RTC bezig met een update cyclus en is het niet verstandig een tijd-, datum- of alarmregister uit te lezen. Dit bit is read-only.

Bit 6,5 en 4 van dit ic bevatten een code waaruit afgeleid wordt wat de waarde is van het gebruikte kristal. Voor het nieuwe bord moet men hier 0, 1 en 0 neerzetten. Voor het oude bord moeten deze alle drie 0 zijn.

Bit 3, 2, 1, en 0 bepalen het interrupt interval wanneer men periodiek een interrupt wenst.

D₃ D₂ D₁ D₀ **tijd tussen twee inter-**
rupts:

				nieuwe bord	oude bord	
0	0	0	0	-	-	*)
0	0	0	1	3.90625	ms	30.517 us
0	0	1	0	7.8125	ms	61.035 us
0	0	1	1	122.070	us	122.070 us
0	1	0	0	244.144	us	244.141 us
0	1	0	1	488.281	us	488.281 us
0	1	1	0	976.562	us	976.562 us
0	1	1	1	1.953125	ms	1.953125 ms
1	0	0	0	3.90625	ms	3.90625 ms
1	0	0	1	7.8125	ms	7.8125 ms
1	0	1	0	15.625	ms	15.625 ms
1	0	1	1	31.25	ms	31.25 ms
1	1	0	0	62.5	ms	62.5 ms
1	1	0	1	125	ms	125 ms
1	1	1	0	250	ms	250 ms
1	1	1	1	500	ms	500 ms

*) Geen periodieke interrupts.

tabel 11.

Register B(&H0B)

bit 7: Set bit. Wanneer men dit bit 1 maakt, wordt een eventueel bezig zijnde update cyclus afgebroken. Dit is handig wanneer men eigen data in de tijd-, datum- of alarmregisters wil schrijven. Hierdoor vindt geen update midden in deze sessie plaats.

bit 6: Periodic Interrupt Enable (PIE). Hierdoor worden periodieke interrupts genereerd (via kanaal 78, CTC1), waarvan de periode gedefinieerd is in register

A. Dit bit wordt gereset na een reset van de computer.

bit 5: Alarm Interrupt Enable (AIE). Wanneer geset, zal een interrupt gegenereerd worden bij het bereiken van de alarmtijd (via kanaal 28, CTC1).

bit 4: Update-ended Interrupt Enable (UIE). Elke keer dat het ic de tijd verandert (een maal per seconde) wordt een interrupt gegenereerd (via kanaal 28, CTC1).

bit 3: Square Wave Enable. De waarde is niet van belang. De pin waarop dit bit invloed heeft, is niet naar buiten gevoerd. PEN 23

bit 2: Data Mode. Dit bit bepaalt het formaat van de dataregisters:

0 - BCD notatie

1 - Binaire notatie

Verandert men dit bit, dan moeten alle tijden e.d. opnieuw geprogrammeerd worden.

bit 1 : 24/12. Met dit bit bepaalt men of men de tijd in 12 of in 24 uurs notatie wil hebben. In 12 uurs notatie geeft zoals in figuur 7. te zien valt, bit 7 van register 4 aan of het 's morgens of 's middags betreft. Verandert men dit bit, dan moeten alle tijden e.d. opnieuw geprogrammeerd worden.

0 - 12 uurs notatie

1 - 24 uurs notatie

bit 0 : Daylight Savings Enable. Dit bit

moet nul zijn, doordat deze functie niet geïmplementeerd is.

Register C(&H0C)

Dit register is read only. De vier minst significante bits ervan zijn niet gebruikt en hebben altijd de waarde 0.

bit 7: Interrupt Request Flag (IRQF). Voor deze flag geldt de volgende booleaanse vergelijking: IRQF = PF.PIE + AF.AIE + UF.UIE

Hierin zijn PF, AF en UF de flags die geset worden wanneer een bepaalde conditie optreedt en PIE, AIE, en UIE de registers die bepalen of, wanneer een van die flags geset is, een interrupt gegenereerd mag worden.

Wanneer de IRQF flag geset wordt, zal ook de IRQ-pin meegaan (laag worden). Deze is op de triggeringang van kanaal 2 van CTC1 aangesloten.

bit 6 : Periodic Interrupt flag (PF). Dit bit wordt geset wanneer het moment is aangebroken dat een periodieke interrupt gegenereerd moet worden. De periode ervan wordt bepaald door de waarde van register A. Dit bit wordt gereset na een RES of na lezen van register C.

Bit 5 : Alarm Interrupt Flag (AF). Dit bit wordt geset wanneer de tijd gelijk is aan de alarmtijd. Het wordt op dezelfde wijze gereset als bit 6.

bit 4 : Update-ended Interrupt flag (UF). Dit bit wordt geset na afloop van elke updatecycle. Het wordt op dezelfde

wijze gereset als bit 6.

Register D(&H0D)

Ook dit register is read only. Alleen bit 7 is gebruikt. De overige bits zijn altijd 0.

bit 7 : Valid RAM and Time bit. Dit bit wordt 0 wanneer de batterij te ver leeg raakt, zodat hij afgeschakeld wordt. Hierdoor gaat alle informatie in het RAM van de RTC verloren.

8.3 Voorbeeldprogramma's

In bijlage 7. is een voorbeeldprogramma opgenomen dat onder BASIC draait. Het zorgt ervoor dat de tijd constant in beeld is.

9 Floppy Disk Controller

De controller gebruikt op het Miniware extension board werkt op dezelfde manier als die op het Philips extension board. Deze is speciaal bedoeld voor drives die werken met de Shugart interface. Wel dienen jumpers in de drive aangepast te worden zoals Philips dit voorschrijft. Uw dealer zal dit gewoonlijk voor u doen.

Verzuimt men dit, dan zal de controller een verkeerde drive adresseren.

De eerste serie van 100 stuks is op de markt gebracht met als floppycontroller de uPD 7265. De latere met het type uPD 765.

uPD 7265 : Sony-compatible Recording Format
uPD 765 : IBM-compatible Recording Format

De uPD 7265 is een later ontwikkelde versie bedoelt voor de nieuwe microfloppies (3.5 inch) van Sony. De 765 is bedoelt voor de 5.25 en 8 inch floppies. De floppy controller is compatible met de controller op het Philips bord. Op beide borden krijgen de chips een 4 MHz klok (i.p.v. 8MHz) toegevoerd. Dit kan problemen opleveren bij het formateren.

Features.

max 4 drives
Single Side/Dual Side (voor micro
floppy's)
Single Density/Double density
DMA/non-DMA mode
Data/scan capability: vergelijking van
een sector of track met data in het
geheugen.
Parallel Seek Operation op max. 4
floppy's

Het werken met de controller.

Het belangrijkste deel van de communicatie met de floppycontroller gebeurt via twee registers: Het Main Status Register en het Data Register.

Het opslaan en teruglezen van data kan gebeuren op semi-DMA en non-DMA basis.

Het werken op semi-DMA basis betekent dat het signaal waarmee de controller een DMA-request aanvraagt, softwarematig gecheckt moet worden.

9.3 Non-DMA methode

Hierbij dient men in een test uit te voeren op bit 7 van het Main Status Register. Wanneer dit bit 1 wordt, kan men een byte naar het Data Register (poort &H8D) sturen danwel uitlezen.

9.4 Semi-DMA methode.

Deze methode wijkt niet zo gek veel af van de vorige. In plaats van het Main Status Register test men nu het DMA-request signaal via bit 0 van poort &H90.

9.5 Commando verwerking.

Er kunnen vijftien commando's uitgevoerd worden, varierend van het positioneren van de lees/schrijfkop tot het lezen of schrijven van een track. Elk commando bestaat uit een aantal bytes die men in het Data Register (poort &H8D) moet wegschrijven. Op de volgende pagina's treft men aan welke gegevens hiervoor nodig zijn. Na elke transfer, van of naar dit Data register, wordt het Main Status Register (poort &H8C) aangepast. Dit aanpassen duurt echter 24 us. In deze tijd mag men het Main Status Register niet uitlezen. Hierna kan men via bit 6 en 7 van dit register nagaan of de volgende parameters danwel een nieuw commando in het Data Register geladen mogen worden.

De communicatie met deze twee registers staat los van het feit of men op semi-danwel non-DMA basis werkt.

De meeste commando's bestaan uit drie fases:

Commando-faze: De controller ontvangt alle gegevens betreffende een commando.

Executie-faze: De controller voert het commando uit. Wanneer de controller in

deze faze om data vraagt, hoeft geen wachttijd van 24 us in aanmerking genomen te worden.

Controle-faze: Status informatie wordt aangepast en moet door de processor gelezen worden.

Elk commando is opgebouwd uit een lijst van gegevens en parameters. Deze worden meestal met een letter aangeduid. De betekenis van deze letters kan men terugvinden in de Command Symbol Description tabel (figuur 8.).

In de controle-faze zullen een aantal bytes met gegevens teruggestuurd worden: status informatie (via ST0-ST3; zie Status Register Identification tabel) en identificatiegegevens. Men moet al deze bytes uitlezen voordat een nieuw commando in het Data Register geschreven kan worden. Men kan dit controleren via bit 6 van het Main Status Register. Dit bit geeft aan of er nog gegevens in het Data Register staan die eerst uigelezen moeten worden, of dat men al een nieuw commando/parameters kan sturen.

Bij de meeste commando's zullen in de controle-faze de eerste drie statusregisters (ST0 - ST2) geretourneerd worden. Hierin kan men terugvinden of het commando het gewenste effect had.

Instruction Set ① ②

Phase	R/W	Data Bus								Remarks
		D ₀	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	
Read Data										
Command	W	MT	MF	SK	C	I	0	1	C	Command Codes
	W	X	X	X	X	I	HD	US ₁	US ₂	Sector ID information prior to command execution.
	W				C					4 bytes are commanded against header on Floppy Disk.
	W				H					
	W				R					
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										
Result	R				ST 0					Data transfer between the FDD and main system
	R				ST 1					Status information after command execution
	R				ST 2					
	R				C					Sector ID information after command execution
	R				H					
	R				R					
	R				M					
	R				R					
	R				N					
Read Deleted Data										
Command	W	MT	MF	SK	0	1	1	0	D	Command Codes
	W	X	X	X	X	I	HD	US ₁	US ₂	Sector ID information prior to command execution.
	W				C					4 bytes are commanded against header on Floppy Disk.
	W				H					
	W				R					
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										
Result	R				ST 0					Status information after command execution
	R				ST 1					
	R				ST 2					
	R				C					Sector ID information after command execution
	R				H					
	R				R					
	R				M					
	R				R					
	R				N					
Write Data										
Command	W	MT	MF	SK	0	1	0	1	C	Command Codes
	W	X	X	X	X	I	HD	US ₁	US ₂	Sector ID information prior to command execution.
	W				C					4 bytes are commanded against header on Floppy Disk.
	W				H					
	W				R					
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										
Result	R				ST 0					Data transfer between the FDD and main system
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				M					
	R				R					
	R				N					

#46
(MT:C6)

4C
(CC)

Notes: ① Symbols used in this table are described at the end of this section.
② An equals binary 1 for all operations
③ X = Don't care; usually made to equal binary 0

Instruction Set

Phase	R/W	Data Bus								Remarks
		D ₀	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	
Write Deleted Data										
Command	W	MT	MF	SK	0	1	0	1	C	Command Codes
	W	X	X	X	X	I	HD	US ₁	US ₂	Sector ID information prior to command execution.
	W				C					4 bytes are commanded against header on Floppy Disk.
	W				H					
	W				R					
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										
Result	R				ST 0					Data transfer between the FDD and main system
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				M					
	R				R					
	R				N					
Read A Track										
Command	W	D	MF	SK	0	0	1	0	C	Command Codes
	W	X	X	X	X	I	HD	US ₁	US ₂	Sector ID information prior to command execution.
	W				C					
	W				H					
	W				R					
	W				N					
	W				EOT					
	W				GPL					
	W				DTL					
Execution										
Result	R				ST 0					Data transfer between the FDD and main system. FDC reads 48 data bytes from track 0 to EOT.
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				M					
	R				R					
	R				N					
Format A Track										
Command	W	D	MF	0	0	1	1	0	C	Command Codes
	W	X	X	X	X	I	HD	US ₁	US ₂	Burst Sector
	W				C					Sectors Track
	W				H					Gag 3
	W				R					Finer Sete
Execution										
Result	R				ST 0					FDC formats an entire track.
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				M					
	R				R					
	R				N					
SCRA Equal										
Command	W	MT	MF	SK	1	0	0	0	1	Command Codes
	W	X	X	X	X	I	HD	US ₁	US ₂	Sector ID information prior to command execution.
	W				C					
	W				H					
	W				R					
	W				N					
	W				EOT					
	W				GPL					
	W				STP					
Execution										
Result	R				ST 0					Data compared between the FDD and main system.
	R				ST 1					
	R				ST 2					
	R				C					
	R				H					
	R				R					
	R				M					
	R				R					
	R				N					

49
(cg)

4D
(-)

51
(D1)

Instruction Set (Cont.)

Phase	R/W	Data Bus								Remarks
		D ₀	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	
Scan Low or Equal										
Command	W	MT	MF	SK	1	1	0	0	1	Command Codes
	W	X	X	X	X	X	HD	US.	US.	Sector ID information prior to command execution
	W	—	—	C	—	—	—	—	—	—
	W	—	—	H	—	—	—	—	—	—
	W	—	—	R	—	—	—	—	—	—
	W	—	—	N	—	—	—	—	—	—
	W	—	—	EOT	—	—	—	—	—	—
	W	—	—	GPL	—	—	—	—	—	—
	W	—	—	STP	—	—	—	—	—	—
Execution										
Result	R	—	—	ST 0	—	—	—	—	—	Data compared between the FDD and main system
	R	—	—	ST 1	—	—	—	—	—	Status information after command execution
	R	—	—	ST 2	—	—	—	—	—	—
	R	—	—	C	—	—	—	—	—	Sector ID information after command execution
	R	—	—	H	—	—	—	—	—	—
	R	—	—	R	—	—	—	—	—	—
	R	—	—	N	—	—	—	—	—	—
	R	—	—	EOT	—	—	—	—	—	—
	R	—	—	GPL	—	—	—	—	—	—
	R	—	—	STP	—	—	—	—	—	—
Scan High or Equal										
Command	W	MT	MF	SK	1	1	1	0	1	Command Codes
	W	X	X	X	X	X	HD	US.	US.	Sector ID information prior to command execution
	W	—	—	C	—	—	—	—	—	—
	W	—	—	H	—	—	—	—	—	—
	W	—	—	R	—	—	—	—	—	—
	W	—	—	N	—	—	—	—	—	—
	W	—	—	EOT	—	—	—	—	—	—
	W	—	—	GPL	—	—	—	—	—	—
	W	—	—	STP	—	—	—	—	—	—
Execution										
Result	R	—	—	ST 0	—	—	—	—	—	Data compared between the FDD and main system
	R	—	—	ST 1	—	—	—	—	—	Status information after command execution
	R	—	—	ST 2	—	—	—	—	—	—
	R	—	—	C	—	—	—	—	—	Sector ID information after command execution
	R	—	—	H	—	—	—	—	—	—
	R	—	—	R	—	—	—	—	—	—
	R	—	—	N	—	—	—	—	—	—
Recalibrate										
Command	W	0	0	0	0	0	1	1	1	Command Codes
	W	X	X	X	X	X	0	US.	US.	—
Execution										
		Sense Interrupt Status								Head retracted to Track 0
Command	W	0	0	0	0	1	0	0	0	Command Codes
Result	R	—	—	ST 0	—	—	—	—	—	Status information about the FDC at the end of seek operation
	R	—	—	PCN	—	—	—	—	—	—
SPECIFY										
Command	W	0	0	0	0	0	0	1	1	Command Codes
	W	—	SRT	—	—	—	HLT	—	—	—
	W	—	—	—	—	—	—	HD	—	—
Sense Drive Status										
Command	W	0	0	0	0	0	0	1	0	Command Codes
	W	X	X	X	X	X	HD	US.	US.	—
Result	R	—	—	ST 0	—	—	—	—	—	Status information about FDC
Seek										
Command	W	0	0	0	0	0	1	1	1	Command Codes
	W	X	X	X	X	X	HD	US.	US.	—
	W	—	—	NCH	—	—	—	—	—	—
Execution										
		Head is positioned over proper cylinder on diskette.								—
Command	W	—	—	Invalid Codes	—	—	—	—	—	Invalid Command Codes (NoOp — FDC goes into Standby state.)
Result	R	—	—	ST 0	—	—	—	—	—	ST 0 = 8C ₁₆

Command Symbol Description

Symbol	Name	Description
A ₀	Address Line 0	A ₀ controls selection of Main Status Register (A ₀ = 0) or Data Register (A ₀ = 1)
C	Cylinder Number	C stands for the current/selected cylinder (track) numbers 0 through 75 of the medium.
D	Data	D stands for the data pattern which is going to be written into a sector.
D ₇ -D ₀	Data Bus	8-bit Data Bus, where D ₇ stands for a most significant bit, and D ₀ stands for a least significant bit.
DTL	Data Length	When N is defined as 00, DTL stands for the data length which users are going to read out or write into the sector.
EOT	End of Track	EOT stands for the final sector number on a cylinder. During Read or Write operations, FDC will stop data transfer after a sector number equal to EOT.
GPL	Gap Length	GPL stands for the length of Gap 3. During Read/Write commands this value determines the number of bytes that VCOs will stay low after two CRC bytes. During Format command it determines the size of Gap 3.
H	Head Address	H stands for head number 0 or 1, as specified in ID field.
HD	Head	HD stands for a selected head number 0 or 1 and controls the polarity of pin 27. (H = HD in all command words.)
HLT	Head Load Time	HLT stands for the head load time in the FDD (2 to 24 ms in 2 ms increments).
HUT	Head Unload Time	HUT stands for the head unload time after a Read or Write operation has occurred (16 to 240 ms in 16 ms increments).
MF	FM or MFM Mode	If MF is low, FM mode is selected, and if it is high, MFM mode is selected.
MT	Mutitrack	If MT is high, a mutitrack operation is performed. If MT = 1 after finishing Read/Write operation on side 0, FDC will automatically start searching for sector 1 on side 1.
N	Number	N stands for the Number of data bytes written in a sector.
NCN	New Cylinder Number	NCN stands for a New Cylinder Number which is going to be reached as a result of the Seek operation. Desired position of head.
ND	Non-DMA Mode	ND stands for operation in the Non-DMA mode.
PCN	Present Cylinder Number	PCN stands for the cylinder number at the completion of Sense interrupt Status command. Position of Head at present time.
R	Record	R stands for the sector number which will be read or written.
R/W	Read/Write	R/W stands for either Read (R) or Write (W) signal.
SC	Sector	SC indicates the number of Sectors per Cylinder.
SK	Skip	SK stands for Skip Deleted Data Address mark.
SRT	Step Rate Time	SRT stands for the Stepping Rate (1 to 16 ms in 1 ms increments). Stepping Rate applies to all drives (F = 1 ms, E = 2 ms, etc.).
ST0	Status 0	ST 0-3 stands for one of four registers which store the status information after a command has been executed. This information is available during the result phase after command execution. These registers should not be confused with the main status register selected by A ₀ = 01. ST 0-3 may be read only after a command has been executed and contains information relevant to that particular command.
ST1	Status 1	—
ST2	Status 2	—
ST3	Status 3	—
STP	Stop	During a Scan operation, if STP = 1, the data in contiguous sectors is compared byte by byte with data sent from the processor (or DMA); and if STP = 2, then alternate sectors are read and compared.
US0, US1	Unit Select	US stands for a selected drive number 0 or 1.

Status Register Identification

Bit	No.	Name	Symbol	Description
Status Register 2				
D ₇		Normal Termination or Command	NTL	D ₇ = 0 and D ₆ = 0 Normal termination of command. INTL Command was completed and properly executed.
D ₆		Abnormal Termination of Command	ATL	D ₇ = 0 and D ₆ = 1 Abnormal termination of command. ATL Execution of command was started but was not successfully completed.
D ₅		Incorrect Command or Value	INC	D ₇ = 1 and D ₆ = 0 Incorrect Command or value. INC Command which was issued was never started.
D ₄		Abnormal Termination because during command execution the ready signal from FDD changed state	ATC	D ₇ = 1 and D ₆ = 1 Abnormal termination because during command execution the ready signal from FDD changed state.
D ₃		Seek End	SE	When the FDC completes the SEEK command, this flag is set to 1 (high).
D ₂		Equipment Choice	EC	If a fault signal is received from the FDD, or if the Track 0 signal fails to occur after 77 step pulses (Recalibrate Command), then this flag is set.
D ₁		Not Ready	NR	When the FDD is in the not-ready state and a read or write command is issued, this flag is set. If a read or write command is issued to Side 1 of a single-sided drive, then this flag is set.
D ₀		Head Address	HD	This flag is used to indicate the state of the head at interrupt.
D ₁		Unit Select 1	US ₁	These flags are used to indicate a Drive Unit Number at interrupt.
D ₀		Unit Select 0	US ₀	
Status Register 1				
D ₇		End of Cylinder	EN	When the FDC tries to access a sector beyond the final sector of a cylinder, this flag is set.
D ₆		Not Used		Not used. This bit always 0 now.
D ₅		Data Error	DE	When the FDC detects a CRC error in either the ID field or the data field, this flag is set.
D ₄		Overtime	OR	If the FDC is not serviced by the host system during data transfers within a certain time interval, this flag is set.
D ₃				Not used. This bit always 0 now.
D ₂		No Data	ND	During execution of READ DATA, WRITE DELETED DATA or SCAN command, if the FDC cannot find the sector specified in the IDR, 2 Register, this flag is set.
D ₁		Not Writable	NW	During execution of WRITE DATA, WRITE DELETED DATA or FORMAT A cylinder command, if the FDC detects a write protect signal from the FDC, then this flag is set.
D ₀		Missing Address Mark	MA	If the FDC cannot detect the ID Address Mark after encountering the end of track header, then this flag is set.
Status Register 2				
D ₇				Not used. This bit always 0 now.
D ₆		Control Mark	CM	During execution of the READ DATA or SCAN command, if the FDC encounters a marker which contains a Deleted Data Address Mark, this flag is set.
D ₅		Data Error in Data Frame	DC	If the FDC detects a CRC error in the data field then this flag is set.
D ₄		Wrong Cylinder	WC	This bit is related to the ID bit, and when the contents of C1 on the medium is different from that stored in the IDR, this flag is set.
D ₃		Scan Equal Hit	SH	During execution of the SCAN command, if the condition of "Equal" is satisfied, this flag is set.
D ₂		Scan Not Satisfied	SN	During execution of the SCAN command, if the FDC cannot find a sector on the cylinder which meets the condition, then this flag is set.
D ₁		Bad Cylinder	BC	This bit is related to the ID bit, and when the contents of C on the medium is different from that stored in the IDR and the contents of C is FF _{hex} , then this flag is set.
D ₀		Missing Address Mark in Data Frame	MD	When data is read from the medium, if the FDC cannot find a Data Address Mark or Deleted Data Address Mark, then this flag is set.

Status Register Identification (Cont.)

Bit	No.	Name	Symbol	Description
Status Register 2				
D ₇		Fault	FT	This bit is used to indicate the status of the Fault signal from the FDD.
D ₆		Write Protected	WP	This bit is used to indicate the status of the Write Protected signal from the FDD.
D ₅		Ready	RY	This bit is used to indicate the status of the Ready signal from the FDC.
D ₄		Track 0	TO	This bit is used to indicate the status of the Track 0 signal from the FDC.
D ₃		Two Side	TS	This bit is used to indicate the status of the Two Side signal from the FDC.
D ₂		Head Address	HD	This bit is used to indicate the status of the Side Select signal to the FDD.
D ₁		Unit Select 1	US ₁	This bit is used to indicate the status of the Unit Select 1 signal to the FDD.
D ₀		Unit Select 0	US ₀	This bit is used to indicate the status of the Unit Select 0 signal to the FDD.

Notes: ① CRC = Cycle Redundancy Check.

② IDR = Internal Data Register

③ Cylinder (C) is described more fully in the Command Symbol Description on page 7.

The bits in the Main Status Register are defined as follows:

Bit	No.	Name	Symbol	Description
D ₇	DB ₀	FDD 0 Busy	DB ₀ B	FDD number 0 is in the Seek mode. If any of the bits is set FDC will not accept read or write command.
D ₆	DB ₁	FDD 1 Busy	DB ₁ B	FDD number 1 is in the Seek mode. If any of the bits is set FDC will not accept read or write command.
D ₅	DB ₂	FDD 2 Busy	DB ₂ B	FDD number 2 is in the Seek mode. If any of the bits is set FDC will not accept read or write command.
D ₄	DB ₃	FDD 3 Busy	DB ₃ B	FDD number 3 is in the Seek mode. If any of the bits is set FDC will not accept read or write command.
D ₃	DB ₄	FDC Busy	CB	A read or write command is in process. FDC will not accept any other command.
D ₂	DB ₅	Execution Mode	EXM	This bit is set only during execution phase in non-DMA mode. When DB ₅ goes low, execution phase has ended and result phase has started. It operates only during non-DMA mode of operation.
D ₁	DB ₆	Data Input Output	DIO	Indicates direction of data transfer between FDC and Data Register. If DIO = 1, then transfer is from Data Register to the processor. If DIO = 0, then transfer is from the processor to Data Register.
D ₀	DB ₇	Request for Master	ROM	Indicates Data Register is ready to send or receive data to or from the processor. Both bits DIO and ROM should be used to perform the hand-shaking functions of "ready" and "direction" to the processor.

9.6 Main Status Register.

bit 0: FDD 0 Busy. Drive 0 is in Seek Mode. De controller zal geen read of write commando accepteren.

bit 1: FDD 1 Busy. Drive 1 is in Seek Mode. idem.

bit 2: FDD 2 Busy. Drive 2 is in Seek Mode. idem.

bit 3: FDD 3 Busy. Drive 3 is in Seek Mode. idem.

bit 4: FDC Busy (CB). De controller is bezig met een read of write commando en zal geen ander commando accepteren.

bit 5: Execution Mode (EXM). Dit bit wordt geset tijdens executie in non-DMA mode. Wanneer dit bit weer nul wordt, is de executiefase afgelopen en is de resultfase begonnen.

bit 6: Data I/O (DIO). Dit bit geeft de richting aan van datatransfer tussen dataregister en controller.

0 : data van processor naar dataregister.
1 : data van data register naar processor

bit 7: Request for Master (RQM). Wanneer dit bit 1 is, is het dataregister klaar om data te ontvangen van de processor of data voor de processor heeft.

In de volgende paragrafen wordt uitgelegd wat de betekenis is van alle commando's. In de Instruction Set tabel kan men zien welke parameters ingevuld moet

ten worden. Sommige parameters kan men terugvinden in de technische specificatie van de diskette stations (floppy drives), zoals aantal tracks, density e.d.

9.7 Commando's.

De volgende commando's kunnen door de controller worden uitgevoerd:

read data	scan high or equal
read id	scan low or equal
specify	read deleted data
read track	write data
scan equal	format track
seek	write deleted data
recalibrate	sense interrupt status
sense drive	status

tabel 12.

9.7.1 Specify commando.

Hiermee kan men de parameters opgeven die specifiek voor een bepaald type drive gelden.

De Head Unload Time (HUT) kan men pro- o (NUL)
grammeren in stapjes van 32 ms. &H01
komt overeen met 32 ms, &H0F (het maxi-
mum) met 480 ms. De Step Rate Time (SRT)
gaat in stapjes van 2 ms. tot 32 ms.
F komt overeen met 2 ms.
E komt overeen met 4 ms. enz.

De Head Load Time in stapjes van 4 ms. = 02
Het maximum hiervan is 508 ms (&H7F).
Ook geeft men met dit commando op of men
in DMA of in non-DMA mode werkt (ND = 1:
non-DMA).

E0 02 /

9.7.2 Sense Drive Status comando.

Hiermee wordt status register 3 aangepast, waarin informatie staat over welke drive geselecteerd is, of er een write-protectie opzit e.d.

Het is wel noodzakelijk dat men via het besturingsregister het 'motor on' commando geeft, daar anders geen drive geadresseerd wordt.

9.7.3 Format a Track.

Met dit commando kan men 1 track formatteren. Het formaat wordt bepaald door het type controller: de uPD 765 of de uPD 7265. Parameters die men op kan geven zijn die voor aantal databytes per sector, de gaplengte (dit is de lengte tussen het laatste databyte van een record en het informatieveld van het volgende record) en welk datapatroon in het dataveld geschreven moet worden.

Tijdens het formatteren moeten een aantal gegevens verstrekt worden. De controller vraagt namelijk om het drivenummer (C),welke zijde (H),sectornummer (R) en de grootte van de sector (N). Hierdoor is het mogelijk om bijvoorbeeld niet sequentiële sectornummers te gebruiken of zelfs verschillende sector-groottes op een track te gebruiken! Deze informatie zal dan ook voor elke sector opnieuw opgevraagd worden.

Er zijn twee manieren om de gevraagde informatie door te spelen naar de controller: in semi-DMA mode of via interrupts. In interruptmode zal de controller dan ook vier interrupts per sector

genereren. In semi-DMA mode zal de controller vier DMA requests genereren. Dit dient men te testen door het afvragen van het minst significante bit van poort &H90.

Nadat een sector is geformatteerd, zal het sectornummer in het status register echter met 1 opgehoogd worden. Dit kan men testen in de controlle-fase van elk commando.

De onderstaande tabel laat het verband zien tussen N, het aan databytes per sector, SC, het aantal sectoren per track en GPL, de gaplengte.(is alleen voor 8 MHz gegeven). Ook is schematisch de indeling van een track in beeld gebracht.

Er kunnen echter problemen ontstaan omdat de controller geen 8 MHz. klok krijgt toegevoerd.

Sectorgrootte FDC.

Format	Sector Size	N	SC	GPL ^①	GPL ^{②,③}
8" Standard Floppy					
FM Mode	128 bytes/sector	00	1A	07	1B
	256	01	0F	0E	2A
	512	02	08	1B	3A
	1024	03	04	47	8A
	2048	04	02	C8	FF
	4096	05	01	C8	FF
MFM Mode ^④	256	01	1A	0E	3E
	512	02	0F	1B	54
	1024	03	08	35	74
	2048	04	04	99	FF
	4096	05	02	C8	FF
	8192	06	01	C8	FF
5½" Minitfloppy					
FM Mode	128 bytes/sector	00	12	07	09
	128	00	10	10	19
	256	01	08	18	30
	512	02	04	46	87
	1024	03	02	C8	FF
	2048	04	01	C8	FF
MFM Mode ^④	256	01	12	0A	0C
	256	01	10	20	32
	512	02	08	2A	50
	1024	03	04	80	F0
	2048	04	02	C8	FF
	4096	05	01	C8	FF
3½" Sony Microfloppy					
FM Mode	128 bytes/sector	0	0F	07	1B
	256	1	09	0E	2A
	512	2	05	1B	3A
MFM Mode ^④	256	1	0F	0E	35
	512	2	09	1B	54
	1024	3	05	35	74

Table 1

Notes:

- ① Suggested values of GPL in Read or Write commands to avoid splice point between data field and ID field of contiguous sectors.
- ② Suggested values of GPL in format command.
- ③ All values except sector size are hexadecimal.
- ④ In MFM mode FDC cannot perform a Read/Write/format operation with 128 bytes/sector. (N = 00)

Data patroon track.

μ PD765A (FM Mode)

GAP 4a 40x FF	SYNC 6x 00	IAM FC	GAP 1 26x FF	SYNC 6x 00	IDAM FE	CYL H D SEC N O CRC	GAP 2 11x FF	SYNC 6x 00	DATA AM FB or FS		DATA ①	CRC	GAP 3 ②	GAP 4b
---------------------	------------------	-----------	--------------------	------------------	------------	---------------------------------------	--------------------	------------------	---------------------	--	-----------	-----	------------	--------



Repeat N Times

μ PD7255 (FM Mode)

GAP 1 16x FF	SYNC 6x 00	IDAM FE	CYL H D SEC N O CRC	GAP 2 11x FF	SYNC 6x 00	DATA AM FB or FS		DATA ①	CRC	GAP 3 ②	GAP 4
--------------------	------------------	------------	---------------------------------------	--------------------	------------------	---------------------	--	-----------	-----	------------	-------



Repeat N Times

μ PD765A (MFM Mode)

GAP 4a 80x 4E	SYNC 12x 00	IAM 3x C2 FC	GAP 1 50x 4E	SYNC 12x 00	IDAM 3x A1 FE	CYL H D SEC N O CRC	GAP 2 22x 4E	SYNC 12x 00	DATA AM 3x A1 FB FS		DATA	CRC	GAP 3 ②	GAP 4b
---------------------	-------------------	-----------------------	--------------------	-------------------	------------------------	---------------------------------------	--------------------	-------------------	---------------------------------	--	------	-----	------------	--------



Repeat N Times

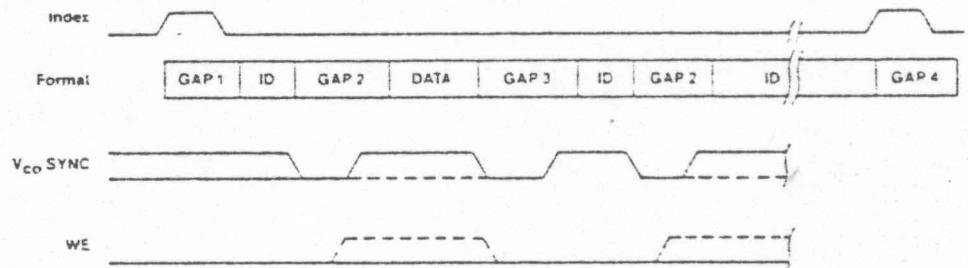
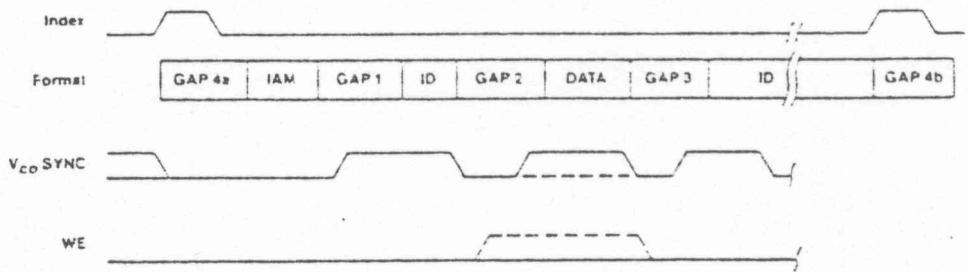
μ PD7255 (MFM Mode)

GAP 1 32x 4E	SYNC 12x 00	IDAM 3x A1 FE	CYL H D SEC N O CRC	GAP 2 22x 4E	SYNC 12x 00	DATA AM 3x A1 FB FS		DATA ①	CRC	GAP 3 ②	GAP 4
--------------------	-------------------	------------------------	---------------------------------------	--------------------	-------------------	---------------------------------	--	-----------	-----	------------	-------



Repeat N Times

figuur 11a.



Note: _____ Read
- - - Write

Notes: It is suggested that the user refer to the following application notes:
 ① #5 — for an example of an actual interface as well as a theoretical: data separator
 ② #10 — for a well documented example of a working phase - locked loop

figuur 11b.

9.7.4 Seek comando.

Met het Seek comando kan men opgeven naar welke track de kop toe moet. Hier voor zal de controller een aantal step-pulsen genereren. Hoe snel dit mag gebeuren, moet men opgeven met SRT in het comando Specify (eenmalig, bij initialisatie van de controller). Dit comando heeft geen controle-faze. Hierom zal de controller, wanneer de kop

gepositioneerd is, een interrupt genereren.

De oorzaak van een interrupt kan men opvragen met het Sense Interrupt commando. Deze zal in de controle-fase statusregister 0 aanpassen. Aan bit 5, 6 en 7 hiervan kan men dan zien of de kop inderdaad boven de juiste track is geopositioneerd.

9.7.5 Recalibrate commando

Met dit commando wordt de lees/schrijfkop naar track 0 verplaatst. Wanneer meer dan 77 tracks moeten worden overbrugd, zal na de 77-ste gestopt worden. Men zal dan meerdere recalibrate commando's moeten geven.

Doordat het Recalibrate commando, net als het Seek commando, geen controle-fase heeft zal de controller in plaats hiervan een interrupt genereren. Via het Sense Interrupt Status commando kan men dan uit statusregister 0 halen of de kop inderdaad bij track 0 is aangekomen.

9.7.6 Read Data.

Nadat alle bytes voor dit commando naar de controller gestuurd zijn, zal de controller indien nog nodig, een head load commando genereren en het eerste identificatieveld lezen. Komt de controller het adres van de juiste sector tegen, dan zal de controller de data byte voor byte binnentrekken. Hierna wordt het Sector Number verhoogd en wordt de volgende sector volledig binnengehaald. Dit noemt men een Multi Sector Read

Operation.

Hoeft er maar een sectorblok ingelezen te worden, of wil men dit commando afbreken, dan moet men bij het laatste byte van het sectorblok tegelijk met een DMA-Acknowledge (in semi-DMA mode) het besturingssignaal Terminal Count geven. Zie hiervoor de paragraaf over het besturingsregister.

9.7.6.1 Transfercapaciteit.

Wanneer de drives geschikt zijn voor dual side, dan is het mogelijk zowel een complete track van kant 1 als kant 2 binnen te halen. Dit wordt bepaald met het MT bit (Multi Track) uit het commandowoord. Wanneer men dit bit 1 maakt, dan zal de eerste sector die binnengehaald wordt sector 1 van kant 1 en de laatste de laatste sector van kant 2 zijn. Met het MFM/FM bit geeft men op of er single of double density is gebruikt. Gewoonlijk worden 16 sectoren gebruikt. Hierdoor kan men de maximum transfer capaciteit uit tabel 14. afleiden.

Transfer Capacity

Multi- Track MT	MFM/ FM MF	Bytes/ Sector N	Maximum Transfer Capacity (Bytes/Sector) (Number of Sectors)	Final Sector Read from Diskettes
0	0	00	(128) (26) = 3.328	26 at Side 0
0	1	01	(256) (26) = 6.656	or 26 at Side 1
1	0	00	(128) (52) = 6.656	
1	1	01	(256) (52) = 13.312	26 at Side 1
0	0	01	(256) (15) = 3.840	15 at Side 0
0	1	02	(512) (15) = 7.680	or 15 at Side 1
1	0	01	(256) (30) = 7.680	
1	1	02	(512) (30) = 15.360	15 at Side 1
0	0	02	(512) (8) = 4.096	8 at Side 0
0	1	03	(1024) (8) = 8.192	or 8 at Side 1
1	0	02	(512) (16) = 8.192	
1	1	03	(1024) (16) = 16.384	8 at Side 1

tabel 14.

Wanneer N (aantal bytes per sector) gelijk is aan 0, moet men met DTL in het commandowoord de werkelijke lengte opgeven van het sectorblok (128 bytes of minder). Is N ongelijk aan 0, dan heeft DTL geen betekenis en moet men hier &HFF invullen.

Nadat alle data gelezen is, zal de leeskop niet gelijk omhoog gehaald worden, afhankelijk van het Head Unload Interval, welke via een ander commando geprogrammeerd kan worden. Dit is vooral handig wanneer men van de ene diskette naar de andere copieert. Hierdoor vervalt de tijd die nodig is om de lees/schrijfkop te laten zakken.

Als de controller twee maal het indexgat tegenkomt zonder dat de juiste sector gevonden wordt, zal de ND (No Data) flag in status register 1 geset worden en het read commando afbreken.

Mochten er datafouten optreden, bijvoorbeeld een verkeerde CRC, dan zal het proces gestopt worden en de aard van de fout in status register 1 weggezet wor-

den.

Wanneer de processor verzuimt data tijdens een read uit de controller te halen, elke 13 us in MFM mode en elke 27 us in FM mode), zal de Overrun (OR) flag in status register 1 geset worden.

9.7.7 Write Data.

Na dit commando zal de drive de kop op de diskette laten zakken en het volgende identificatieveld lezen. Wanneer een veld gevonden is met de juiste identificatie (adres), bepaald door het drive-nummer, diskzijde, sectornummer en lengte van de sector, dan zal data byte voor byte via het Data Register beschikbaar komen voor de processor.

Is de sector volgeschreven, dan zal de drive data voor de volgende sector verwachten. Dit kan men op dezelfde manier afbreken als een Read commando.

Bij fouten wordt statusregister 1 aangepast.

9.7.8 Read a Track.

Dit commando vertoont grote overeenkomst met het Read Data commando. Onmiddelijk nadat het indexgat gepasseerd is, zal de controller beginnen te lezen en alle data van de gehele track aaneengesloten binnenhalen. Ook na ID- of CRC-fouten zal doorgelezen worden. De FDC controleert wel of de ID informatie overeenkomt met de informatie in het Internal Data Register. Wanneer deze niet overeenkomen, zal de ND-flag geset worden.

Het is niet mogelijk om met dit commando beide zijden van de diskette in een keer te lezen.

Het commando zal afgebroken worden wanneer de laatste sector gelijk is aan EOT. Mocht er geen data op de track staan, dan zullen statusregister 0 en 1 aangepast worden.

9.7.9 Scan commando's.

Met dit commando kan men een stuk in het geheugen vergelijken met een sector. Dit vergelijken gebeurt byte voor byte. Er zijn eigenlijk drie commando's. De eerste test op gelijk, de tweede op gelijk of minder, de derde op gelijk of hoger.

Het commando stopt wanneer de vergelijking klopt, de laatste sector is gelezen of na het geven van Terminal Count.

9.7.10 Read ID (identificatieveld).

Door dit commando verkrijgt men de positie van de leeskop. De controller zal de ID op slaan van het eerste veld dat het kan lezen. Bij fouten worden statusregister 0 en 1 aangepast.

9.7.11 Read Deleted Data.

Dit commando zorgt ervoor dat wanneer de controller een Data Address mark tegenkomt dat verwijderd is (ongeldig is gemaakt), de data van deze sector gele-

zen wordt, mits SK gelijk is aan nul.
Doet men dit laatste niet, dan zal over
deze sector heen gesprongen worden.
Wanneer de data binnengehaald is, zal de
CM flag in status register 2 worden
geset en met uitvoering van het commando
stoppen.

9.7.12 Write Deleted Data.

Door dit commando wordt het Data Address
Mark van de opgegeven sector verwijderd.

9.7.13 Sense Interrupt Status.

Het Sense Interrupt Status commando is
bedoeld voor commando's die geen con-
trolle-faze hebben, zoals het Seek en
het Recalibrate commando.
Via dit commando kan de processor nagaan
wat in die gevallen de oorzaak was van
de interrupt.

9.7.14 Ongeldig commando.

Geeft men een commandocode op die niet
bestaat, dan zal de controller overscha-
kelen naar de controle-faze waardoor
men het Main Status Register zal moeten
uitlezen. Hierin staat dan de waarde
&H80 (fout commando).

9.8 Besturingsregister.

Via dit register kan de controller in
een bepaalde toestand gedwongen worden.

Die besturingssignalen worden via poort &H90 doorgegeven aan de controller.
Hiervan worden de vier minst signifi-
cante bits gebruikt. De betekenis hier-
van is:

D₀: 0 - DMA-Acknowledge
 1 - Chip Select
D₁: 1 - Terminal Count
D₂: 0 - Reset Controller
D₃: 1 - Motor on
 0 - Motor off

Wanneer de computer opgestart wordt,
staat het besturingsregister in de re-
settoestand. Dit moet ongedaan gemaakt
worden door bit 0 en 2 gelijk aan 1 te
maken.

Het aan- en afschakelen van de motor van
de diskettestations wordt niet door de
controller geregeld. Dit moet de ge-
bruiker zelf doen via bit 3 van het
besturingsregister. Nadat het motor-on
signaal wordt gegenereerd moet er enige
tijd gewacht worden voordat er
lees/schrijf operaties kunnen worden
uitgevoerd. Deze tijd is ongeveer 0,5
sec.

Via Terminal Count wordt een commando
als Read en Write Data afgebroken. Het
moet gegeven worden bij het binnenvullen
of versturen van het laatste byte van
een sectorblok, tezamen met DMAAcknow-
ledge (in semi-DMA mode).

Wanneer de controller in semi-DMA mode
is geprogrammeerd, moet tijdens data-
transfer met de diskettes het Chip Se-
lect signaal vervangen worden door DMA-
Acknowledge.

9.9 Voorbeelden besturingsregister.

Het gebruik van dit register wordt in de volgende voorbeelden nog eens verduidelijkt.

Herstellen Reset toestand:

```
ld    a,00000101t  
out  (90),a
```

Software Reset controller:

```
ld    a,0          ;Reset  
out  (90),a  
ld    a,00000101t ;haal reset weg  
out  (90),a
```

Terminal Count:

```
ld    a,00001110t ;DMA-Acknowledge  
                  ;Terminal Count en  
out  (90),a       ;Motor on  
ld    a,00001101t ;Haal Terminal Count  
out  (90),a       ;weg
```

DMA-Acknowledge:

Het gebruik hiervan wordt in een voorbeeld duidelijk gemaakt.

```
;  
; subroutine voor het opslaan van een  
; sectorblok  
; na afloop wordt actie afgebroken met  
; een Terminal Count  
; hl bevat beginadres datablok bestemd  
; voor diskette  
;  
$wrdata  
    push af  
    push bc  
    ld    c,8d          ; c := adres  
                      ; Data Register  
                      ; RS232  
    ld    b,0          ; b := 256 (aan-  
                      ; tal bytes)  
    ld    a,000001100t  ;geef DMA-Ack-  
                      ; nowledge  
    out   (90),a  
;  
$poll  
    in    a,(90)        ; DMA-request?  
    bit   0,a  
    jr    z,$poll       ; Nee, test opnieuw  
    outi  
    ; transfer byte,  
    ; verhoog hl,  
    ; verlaag b  
    jr    nz,$poll  
  
    ld    a,000001110t  ; geef Terminal  
                      ; Count  
    out   (90),a  
    ld    a,000001101t  ; haal Terminal  
                      ; Count weg en  
    out   (90),a  
                      ; herstel Chip  
                      ; Select
```

pop bc
pop af
ret

10 Beschrijving hardware

In de bijlagen is het electrisch schema opgenomen van het Multifunktiebord. Dit betekent echter niet dat het vrijgegeven is. Hierdoor wordt echter wel de mogelijkheid geschapen om het bord naar eigen believen uit te breiden danwel te repareren.

In de volgende paragrafen wordt hierop een sumiere toelichting gegeven.

10.1 I/O decodering.

Op sheet 1 treft men de I/O decodering en systeembus-buffers aan.

IC129 is zorgt voor het doorgeven in de juiste richting van de databus terwijl IC 105 en 151 de overige bussignalen bufferen.

IC 107 is een prom dat de I/O adressen &H80 tot &H9F uit decodeerd. Welke gebieden dit zijn kan men uit de signaalnamen afleiden. IC 100, 102, 155 en 160 splitsen dit nog nog eens verder op:

CTCB: &H80 - &H83
SIO : &H84 - &H87
CTCI: &H88 - &H8B
FDC : &H8C - &H8D
WRP : &H90 - &H93
RDP : &H90 - &H93
MMBS: &H94
EMTS: &H95

EMSS : &H96
EMDS : &H97
CDO : &H98
CDI : &H99
CDN : &H9A
CSF : &H9B
RTCR: &H9C
RTCD: &H9D

Alle signalen werken met negatieve logica

tabel 15.

WRP en RDP zijn de read- en write-signalen welke alleen actief zijn in het adresgebied &H90 - &H93.

Pin 1 van IC 129 geeft aan of deze databus-transceiver voor een read of een write cyclus gebruikt wordt.

10.2 64 KByte geheugenuitbreidung.

Een gedeelte van het geheugen bestaat uit 6 banken van elk 8 KByte. Het resterende deel is een aaneensluitend stuk van 16 KByte. Wanneer we naar de memory map kijken van de gebruikte RAM ic's, dan zien we dat dit op de volgende manier is ingedeeld:

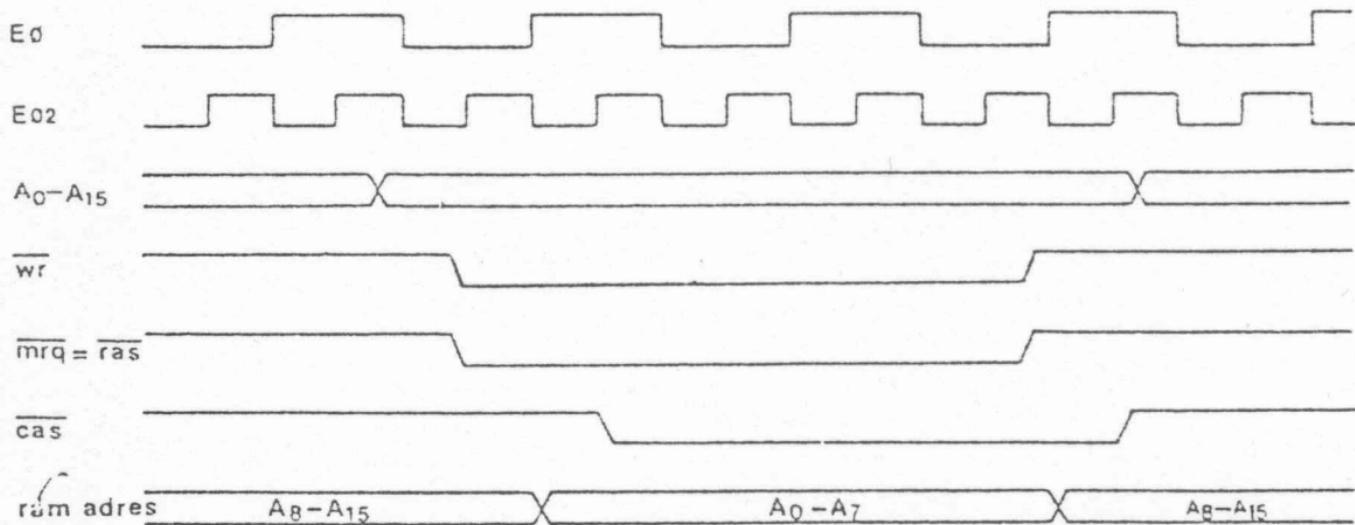
&H0000 - 1FFF : bank 1
&H2000 - 3FFF : bank 2
&H4000 - 5FFF : bank 3
&H6000 - 7FFF : bank 4
&H8000 - 9FFF : bank 5
&HA000 - DFFF : 16 KByte User RAM
&HE000 - FFFF : bank 0

tabel 16

Deze indeling wordt voor rekening genomen door IC 144, een promm. Hierin vindt de decodering plaats van het banknummer dat via de bankswitch geselecteerd is. Dit banknummer kan men in IC 142, een aantal D-flops, opslaan. Hiervoor worden de drie minst significante bits gebruikt. Wanneer hierin een 6 of een 7 weggeschreven wordt (niet bestaande banken), dan zal bank 0 geselecteerd worden.

IC 145 is een opteller welke de gedecodeerde off-set uit het prom optelt bij het adres dat op de bus staat. Hierdoor ontstaat de bovenstaande memory-map. IC 146 en 147 zorgen dat in twee slagen het rij- en kolomadres (ras/cas) aan het RAM aangeboden wordt. De timing hiervoor wordt verzorgd door IC 148.

Timing ras/cas.



figuur 12.

10.3 RAM disk

De RAM disk heeft een 64 of 256 KByte geheugen, afhankelijk van de aangeschafte versie.

Het adres van de RAM disk wordt bepaald door IC 165, 164 en 163. Dit zijn respectievelijk het trackregister, sectorregister en de bytecounter. Deze laatste wordt elke keer verhoogd wanneer een byte is gelezen of geschreven. Hij wordt gereset wanneer het sectorregister wordt geladen.

IC 167 en 168 zorgen ervoor dat op het juiste moment het rij- en kolomadres wordt aangeboden. Wanneer de resetknop is ingedrukt, wordt de refresh overgenomen door een teller, IC 161. IC 162 zorgt ervoor dat de uitgangen van deze teller gebufferd en in tri-state gestuurd kunnen worden.

10.4 Seriele en paralelle interfaces.

Voor deze interface is niet veel uitleg nodig. Ze worden voor het grootste deel geregeld door twee VLSI-ic's: een SIO (seriele interface) en een CTC (Counter Timer Circuit).

IC 124 is een RS422 driver, welke hier door kanaal B van de SIO bestuurd wordt. Deze werkt volgens het principe van gebalanceerde transmissie: hierdoor staat op pin 6 steeds de tegenovergestelde polariteit van pin 7. Pin 2 en-abeld de zender, pin 3 de ontvanger. Via pin 7 en 8 van de CTC worden de frequenties voor de baudrateinstelling van de ontvanger resp. zender van kanaal A (RS232) doorgegeven.

Rond IC 119, 130 en 132 is een Centronics interface opgebouwd. Via IC 130 (D-fliflop) wordt het datasignaal aangeboden, via IC 119 wordt de statusinformatie binnengehaald.

Met IC 132 is een SR-flipflop gemaakt waarmee een strobe-signaal gemaakt is. Twee van de poorten zijn parallel gezet om voldoende vermogen te kunnen leveren. De flipflop wordt geset door het adresseren van &H9A, gereset door &H9B.

10.5 Real Time Clock met battery backup.

IC 157 is een VLSI-ic waarin alle taken geïntegreerd zijn. De benodigde tijdbasis wordt gevormd door K1, een kristal van 32768 Hz., en kan worden afgeregeld met trimmer C2. De frequentie kan gemeten worden op pin 25 van IC157. ! P E N z / d u s . Wanneer de resetingang van dit ic hoog wordt terwijl de voedingsspanning nog aanwezig is, zal het ic overgaan naar de Standby mode: er zal minder vermogen geconsumeerd worden. Deze resetingang wordt hoog wanneer de resetknop ingedrukt wordt of wanneer men de netvoeding uitschakeld. In dit laatste geval wordt de voeding overgenomen door een batterij (B1).

Wanneer de computer aan staat, zal deze batterij automatisch worden opgeladen. Wordt de computer lange tijd niet gebruikt, dan kan de batterij te ver leeg raken. De detectie hiervan gebeurt door een zenerdiode en een weerstand (V9/R4). Wordt de spanning te laag, dan zal pin 13 van IC 158 een nul zien, waardoor de voedingsspanning van IC 157 (RTC) wegvalt. Hierdoor wordt algehele ontlading van de battery voorkomen, wat de levensduur ten goede komt.

10.6 Floppy disk controller.

Het hart van de schakeling wordt gevormd door 3 ic's: IC 112, 113 en 115. Dit zijn resp. de floppy controller, een CTC (interruptcontroller) en een dataseparator. Deze laatste zorgt ervoor dat het datasignaal dat van en naar de drives gaat voorzien of ontstaan wordt van clock- en modulatiesingalen.

Via IC 110 (een D-flipflop) worden besturingssignalen aan de controller doorgegeven. Enkele ervan (TC: Terminal Count; RES: Reset) staan continu op de controller, andere (CS: Chip Select; DACK: DMA-Acknowledge) alleen wanneer de controller geadresseerd wordt (via adres &H8C - &H8D).

Via de inputpoort wordt het DMA-signaal afgevraagd, welke dan via D_0 aan de processor doorgegeven wordt.

IC 139 decodeert US_0 en US_1 , welke gebruikt worden voor het adresseren van de drives. Deze wordt echter alleen geenableerd wanneer het "motor-on" signaal van IC110 verkregen wordt.

Met behulp van K2 (8 MHz) is een oscillator opgebouwd. Hiervan maakt IC 115 een symmetrisch signaal welke via pin 8 hiervan aan de controller aangeboden wordt. Dit signaal heeft een frequentie van 4 MHz.

IC 118 controleert, of de drives vergrendeld zijn. Hiervoor wordt het indexsignaal gebruikt. Wanneer namelijk de drives niet vergrendeld zijn tijdens datatransport, zal geen indexsignaal afgegeven worden. Na 1 seconde zal dan een interrupt gegenereerd worden. Hier-

door wordt voorkomen dat de floppy controller de computer 'ophangt'. De reme-
die in deze gevallen is de controller te
resetten en opnieuw te initialiseren.

Aansluitingen subD-connector

pennr.	Centronics	RS 232	RS 422
1			ground +5V
2		TxD	
3		RxD	
4		RTS	
5		CTS 10X	
6		DCD Gm6	
7		ground	
8	ground		
9	STROBE *)		
10	D ₀		
11	D ₂		
12	D ₄		
13	D ₆		
14			
15			TRxD TRxD *)
16	ACKN *)		
17	BUSY		
18	Papier op		
19	Printer on		
20		DTR	
21	ERROR *)		
22	D ₁		
23	D ₃		
24	D ₅		
25	D ₇		

* signalen met negatieve logica.

Vooraanzicht subD-connector

1 13 *)

- - - - - - - - -

14 25 *)

*) pinnummers

Connector Floppy disk interface.

1 L	18 DIRECTION
2 NC	19 L
3 L	20 STEP *)
4 NC	21 L
5 L	22 WRITE DATA *)
6 DRISELO *)	23 L
7 L	24 WRGATE *)
8 INDEX *)	25 L
9 L	26 TRACK00 *)
10 DRISEL1 *)	27 L
11 L	28 WRPROT *)
12 DRISEL2 *)	29 L
13 L	30 READ DATA *)
14 DRISEL#3 *)	31 L
15 L	32 HEADSEL *)
16 MOTORON *)	33 L
17 L	34 NC

*) signalen met negatieve logica

Vooraanzicht connector floppy controller

2	34 *)
-----	-----
-----	-----
1 - .	35 *)

*) pinnummers

Gebruikte I/O-poorten.

&H80 CTC2 kanaal 0
&H81 CTC2 kanaal 1
&H82 CTC2 kanaal 2
&H83 CTC2 kanaal 3

&H84 Dataregister RS 232
&H85 Commando/Status-register RS 232
&H86 Dataregister RS 422
&H87 Commando/Status-register RS 422

&H88 CTC1 kanaal 0
&H89 CTC1 kanaal 1
&H8A CTC1 kanaal 2
&H8B CTC1 kanaal 3

&H8C Commandoregister FDC
&H8D Dataregister FDC

&H90 - &H93 Besturingsregister FDC *)
&H90 DMA-request signaal (read only)

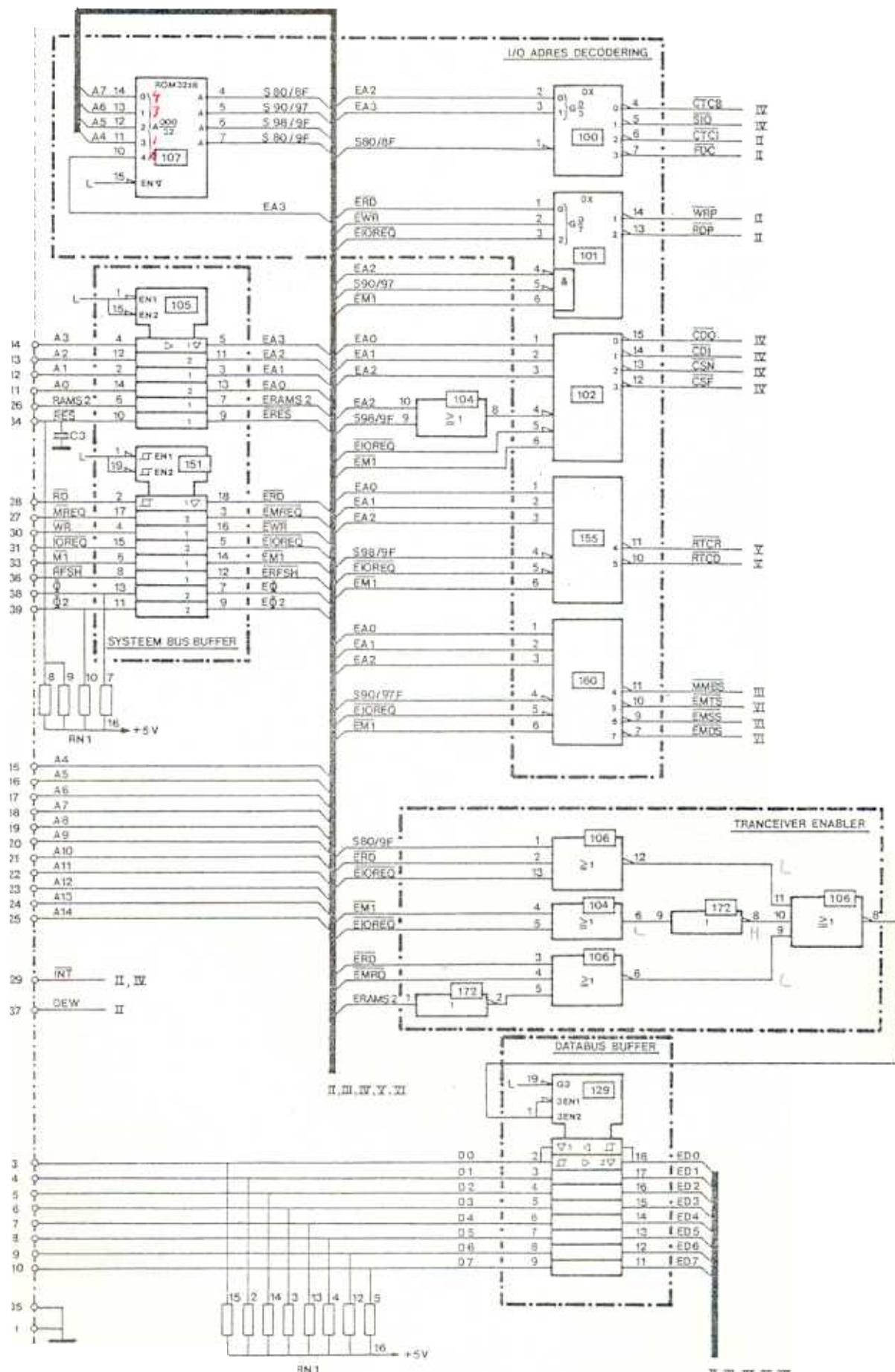
&H94 Bank switch geheugenuitbreiding *)
&H95 Trackregister RAM disk *)
&H96 Sectorregister ,,*)
&H97 Data I/O-register ,,*)

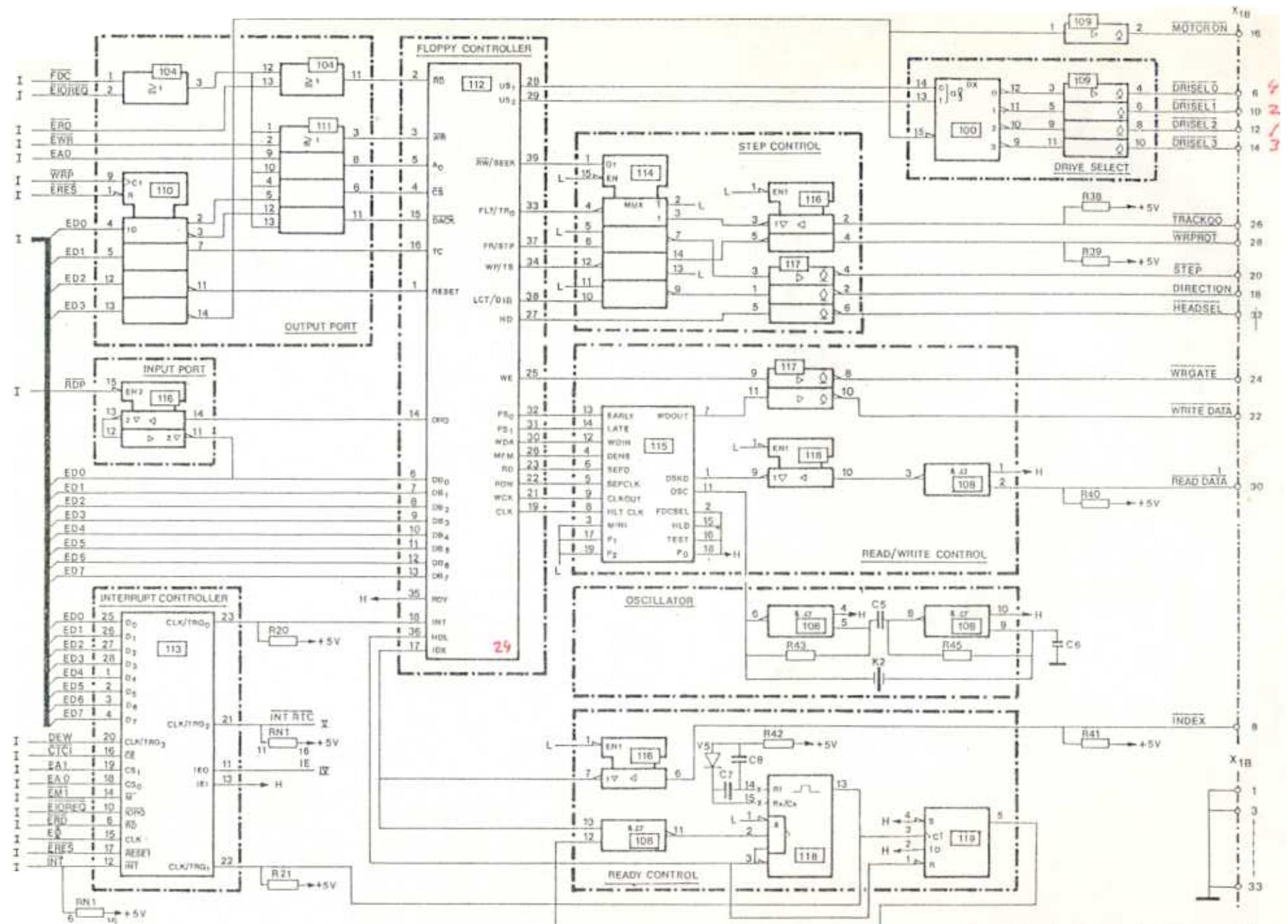
&H98 Data-register Centronics *)
&H99 Statusregister Centronics **)
&H9A Strobe aan Centronics
&H9B Strobe uit Centronics

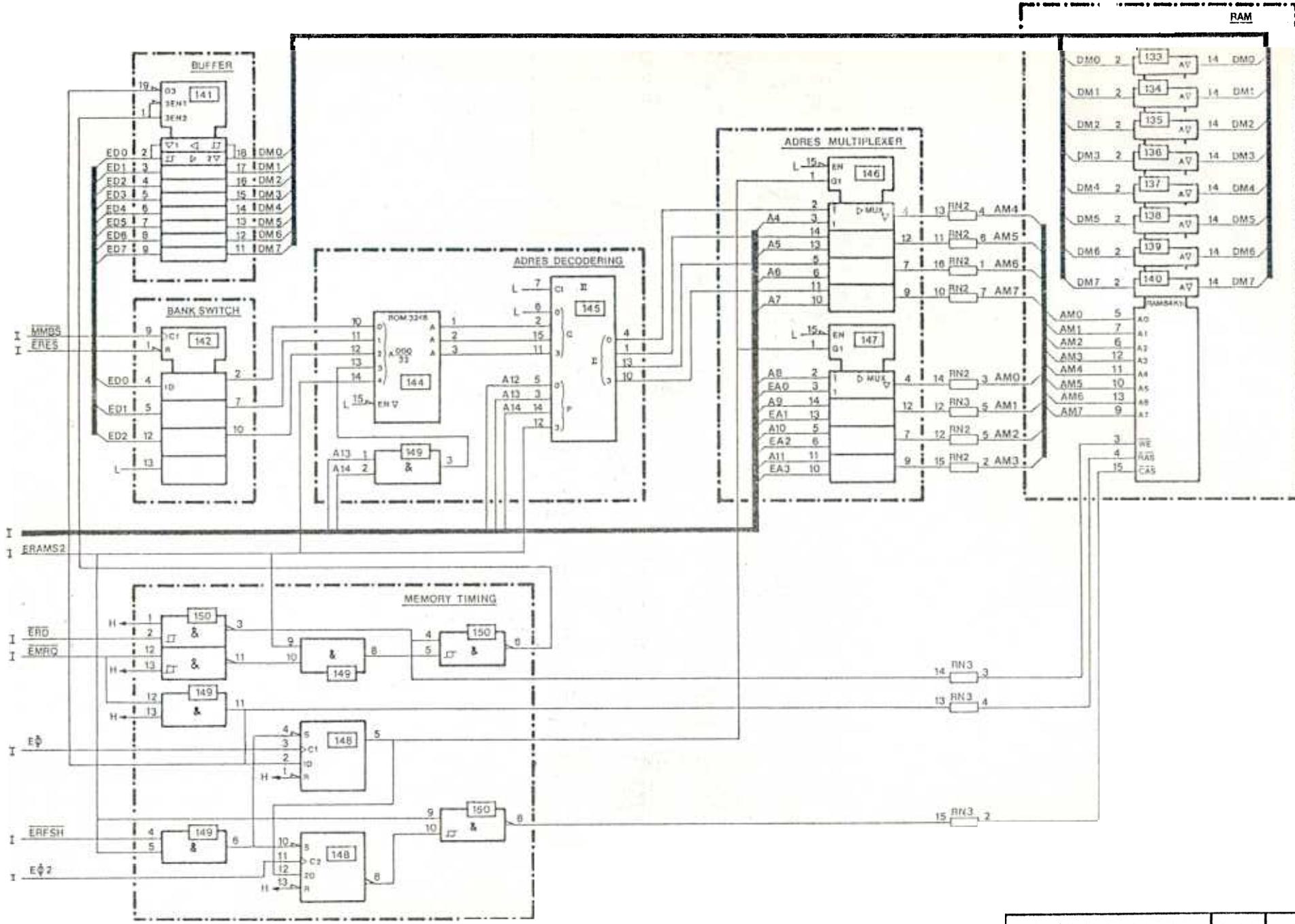
&H9C Selectieregister RTC *)
&H9D Dataregister RTC

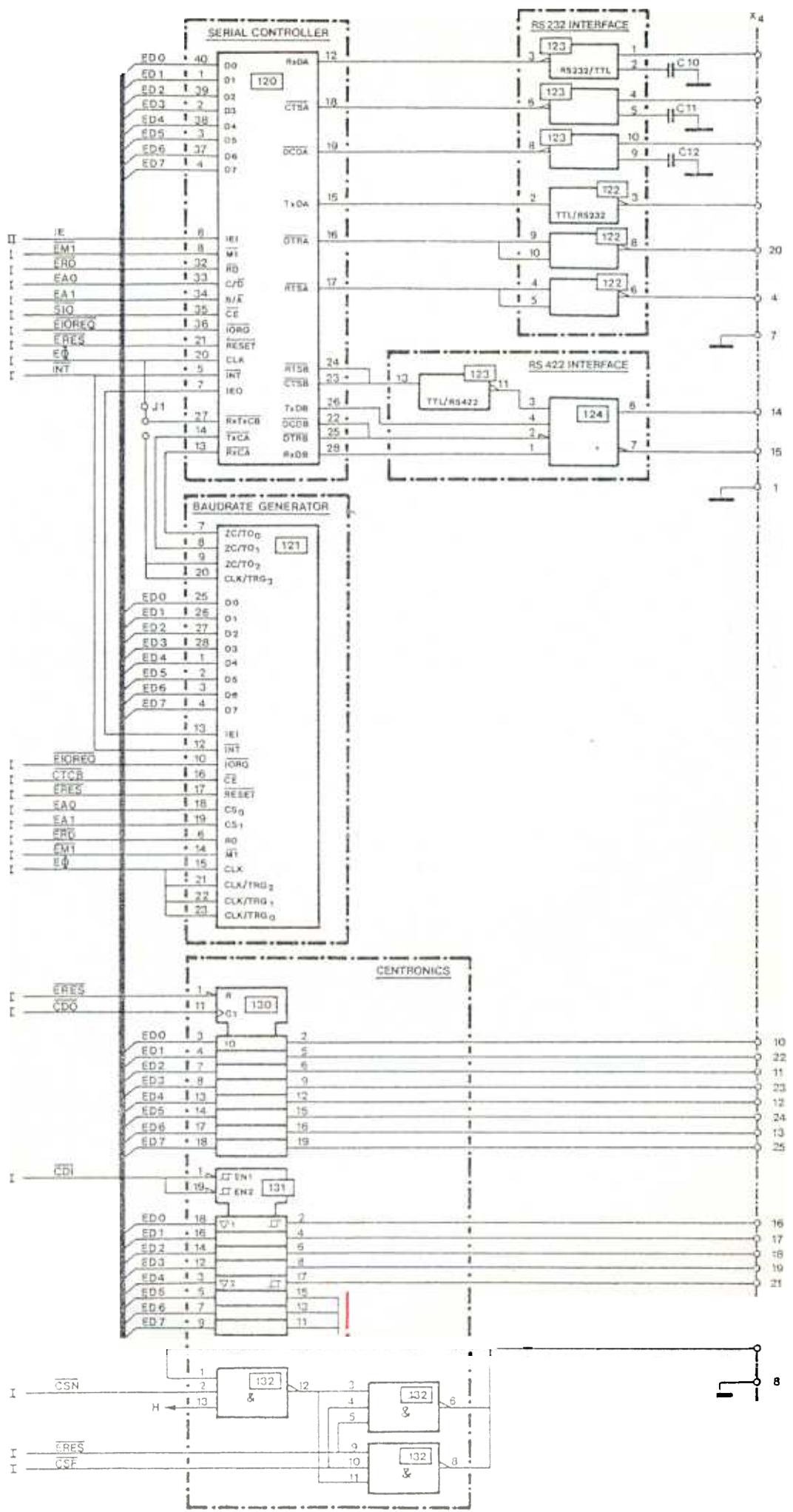
*) write only
**) read only

N.B. Hoewel niet alle poorten tussen &H80 en &H9F gebruikt zijn, is het niet mogelijk de niet gebruikte input-poorten in dit gebied voor eigen applicaties aan te wenden. Dit wordt veroorzaakt door het feit dat de databus-transceiver toch ge-enabeld wordt.

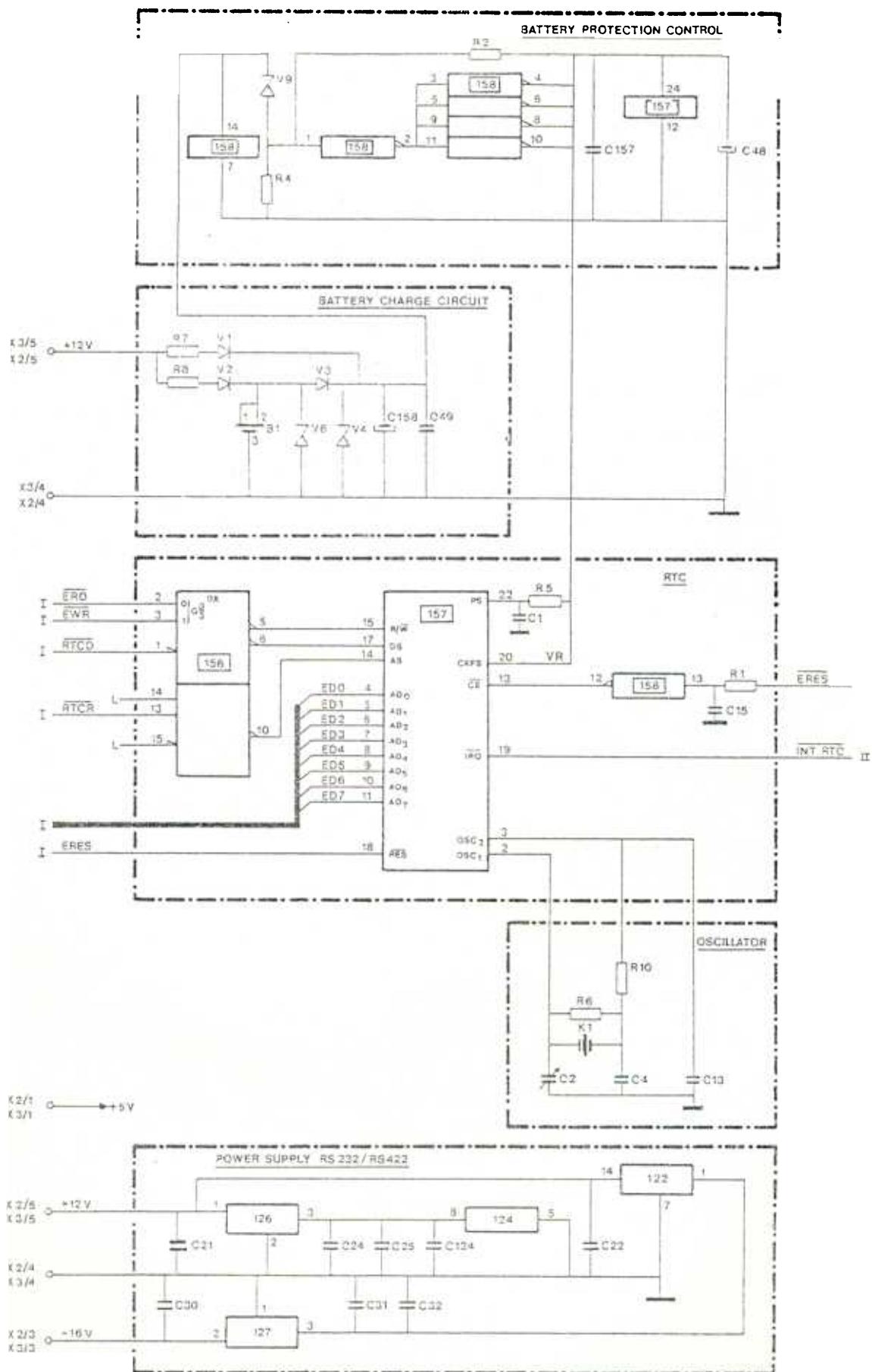


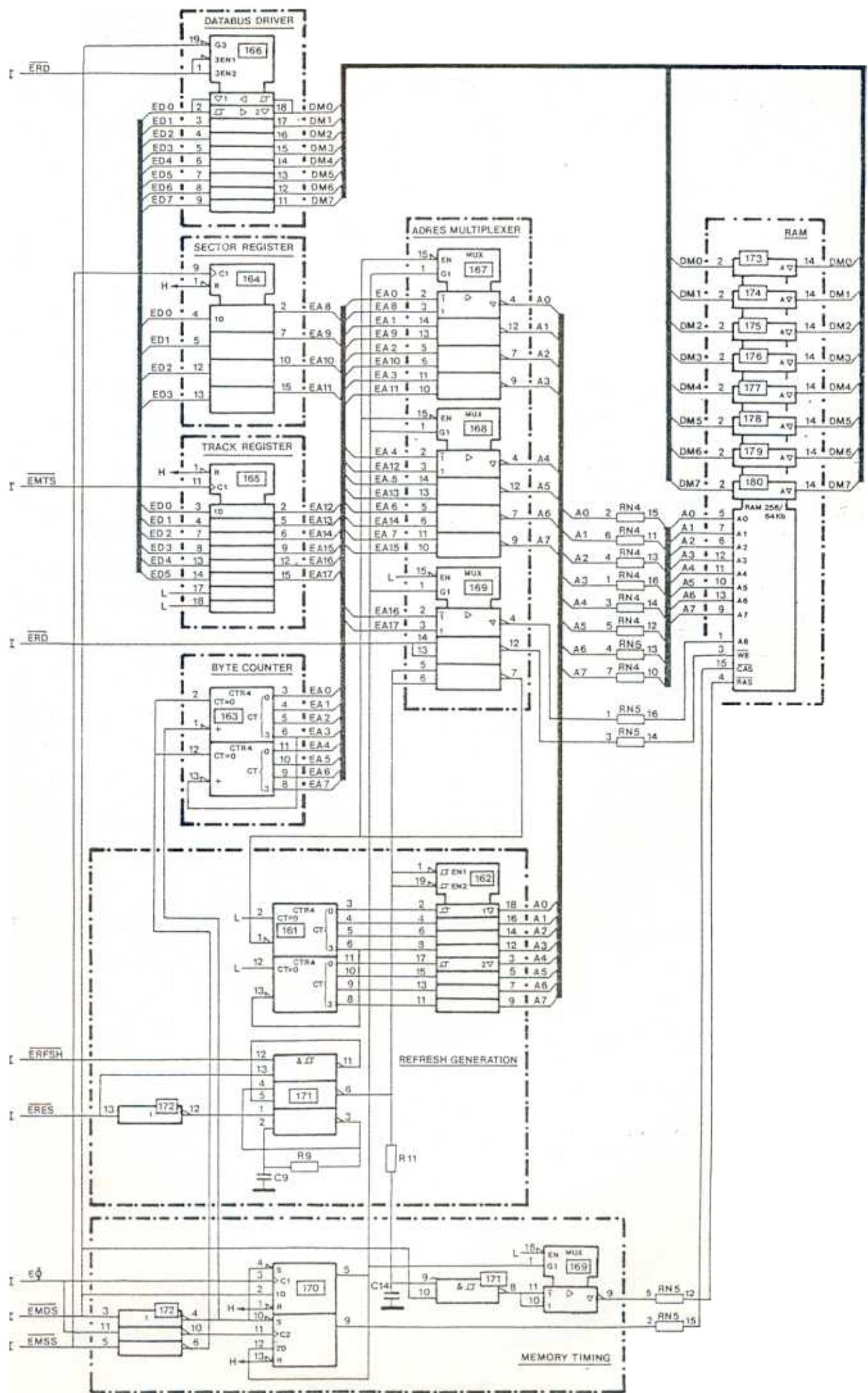






M2200 RS232 RS422 CENTRONICS
PRINCIPLESCHHEMA IV





STUKLIJSTEN MULTIFUNKTIEKAART M2200

IC 100	-	74LS139	IC 141	-	74LS245
IC 101	-	74LS138	IC 142	-	74LS175 13→14,15
IC 102	-	74LS138	IC 144	-	82S123
IC 104	-	74LS32	IC 145	-	74LS283
IC 105	-	74LS367	IC 146	-	74LS258
IC 106	-	74LS27	IC 147	-	74LS258
IC 107	-	82S123	IC 148	-	74LS74
IC 108	-	74LS132	IC 149	-	74LS08
IC 109	-	7407 13→12	IC 150	-	74LS132
IC 110	-	74LS175	IC 151	-	74LS244
IC 111	-	74LS32	IC 155	-	74LS138
IC 112	-	UPD765	IC 156	-	74LS139
IC 113	-	Z8430-CTC	IC 157	-	HD146818
IC 114	-	74LS157	IC 158	-	HEF4069
IC 115	-	FDC9229	IC 160	-	74LS138
IC 116	-	74LS368	IC 161	-	74LS393
IC 117	-	7406 13→12	IC 162	-	74LS244
IC 118	-	74LS123 HALF	IC 163	-	74LS393
IC 119	-	74LS74 "	IC 164	-	74LS175
IC 120	-	Z8440-SIO	IC 165	-	74LS273
IC 121	-	Z8430-CTC	IC 166	-	74LS245
IC 122	-	75188	IC 167	-	74LS258
IC 123	-	75189	IC 168	-	74LS258
IC 124	-	75176	IC 169	-	74LS258
IC 126	-	LM7805	IC 170	-	74LS74
IC 127	-	LM7912	IC 171	-	74LS132
IC 129	-	74LS245	IC 172	-	74LA04
IC 130	-	74LS273	IC 173	-	HM50256/HM4864
IC 131	-	74LS244 βIT 5-6-7	IC 174	-	HM50256/HM4864
IC 132	-	74LS10	IC 175	-	HM50256/HM4864
IC 133	-	HM4864	IC 176	-	HM50256/HM4864
IC 134	-	HM4864	IC 177	-	HM50256/HM4864
IC 135	-	HM4864	IC 178	-	HM50256/HM4864
IC 136	-	HM4864	IC 179	-	HM50256/HM4864
IC 137	-	HM4864	IC 180	-	HM50256/HM4864
IC 138	-	HM4864			
IC 139	-	HM4864			
IC 140	-	HM4864			

STUKLIJSTEN MULTIFUNKTIEKAART M2200

C1	-	1nF	C24	-	100nF
C2	-	40pF trimmer	C25	-	1uF 16V
C3	-	1nF	C30	-	100nF
C4	-	33pF	C31	-	100nF
C5	-	1nF	C32	-	100nF
C6	-	100pF	C40	-	47uF 16V
C7	-	1nF	C41	-	47uF 16V
C8	-	47uF	C42	-	47uF 16V
C9	-	10nF	C43	-	47uF 16V
C10	-	10nF	C44	-	47uF 16V
C11	-	10nF	C45	-	47uF 16V
C12	-	10nF	C46	-	47uF 16V
C13	-	33pF	C47	-	47uF 16V
C14	-	100pF	C48	-	1uF 16V
C15	-	1nF	C49	-	1uF 16V
C21	-	100nF			
C22	-	100nF			
C100	-	C102	-	100nF	
C104	-	C121	-	100nF	
C123			-	100nF	
C124			-	100nF	
C129	-	C142	-	100nF	
C144	-	C151	-	100nF	
C155	-	C158	-	100nF	
C160	-	C180	-	100nF	

STUKLIJSTEN MULTIFUNCTIEKAART M2200

R 1	-	4K7	R 11	-	1K
R 2	-	5M6	R 38	-	150E
R 4	-	240K	R 39	-	150E
R 5	-	4K7	R 40	-	150E
R 6	-	5M6	R 41	-	150E
R 7	-	1K	R 42	-	100K
R 8	-	220E	R 43	-	240E
R 9	-	1K	R 45	-	240E
R 10	-	150K			

RN 2	-	8x33E	RN 5	-	8x33E
RN 3	-	8x33E	RN 1	-	15x4K7
RN 4	-	8x33E			

V 1	-	1N4148	V 4	-	BZX79 5V1
V 2	-	1N4148	V 5	-	1N4148
V 3	-	AA129	V 6	-	BZX79 4V1
			V 9	-	BZX79 3V0

K 1 - X-TAL 32768 Hz K 2 - X-TAL 8MHz

B 1 - 3.6V

X 1 - 40 POLIGE CONNECTOR AMP 4-826711-0
X 2 - 5 POLIGE CONNECTOR AMP 0-164713-5
X 3 - 5 POLIGE CONNECTOR AMP 0-164713-5
X 4 - 25 POLIGE SUBD CONNECTOR AMP 0-164494-1

Voor X1 busbankabel AMP 1-161142-9

Voor X2 voedingssnoer AMP 9-161134-0

