

SUPERMAX HARDWARE

Technical Manual  
for the  
SIOC MODULE 0300  
October 1984

## SUPERMAX HARDWARE BESKRIVELSE.

Dette dokument beskriver den Serielle I/O Kontroller i  
SUPERMAX datamaten: SUPERMAX MODULE SIOC 0300.

Dokumentation: 1. Programmers manual.  
2. Hardware beskrivelse.  
3. Diagram, komponentliste og x-reference.  
4. Detaljeret PAL beskrivelse.

Nødvendig viden: SUPERMAX bus-system.  
Programmers guide.

Indholdsfortegnelse:	side
1. Introduktion.....	3
2. Basale hardware dele.....	3
3. Cycles.....	4
4. Bus arkitektur.....	5
5. Lager konfiguration.....	7
6. Internt lager.....	11
7. Memory mapper.....	12
8. CPU.....	15
9. Interrupt.....	17
10. Klok.....	20
11. Ready.....	21
12. Adresse dekodning.....	21
13. Parallel printer port.....	22
14. USARTs.....	22
15. Reset.....	24
16. TRAP.....	24
17. Time-out.....	25
18. Buserror.....	25
19. Error.....	25
20. Lager cycles.....	26
21. Aktive cycles.....	26
22. Passive cycles.....	27
23. PAL beskrivelse.....	27
24. Detaljeret PAL beskrivelse.....	33
25. Div. figure.....	

## 1. Introduktion.

SIOC'en er et intelligent SUPERMAX print kort. SIOC'en udgør den nødvendige forbindelse mellem MCU'er (applikations - processorer) og serielle I/O linier. Til SIOC'en kan tilsluttes 8 serielle kanaler og en parallel printer. De 6 kanaler er standard asynkrone RS-232 linier. De 2 kanaler er asynkrone eller synkrone RS-232 eller RS-422 linier. Standard for alle 8 kanaler er:

- \* Programmerbar baudrate fra 50 til 19.200 baud
- \* Karakterlængde fra 5 til 8 bits.
- \* Ulige, lige eller ingen paritet.
- \* 1, 1.5 eller 2 stop bits.

SIOC'en har et fuldstændigt interface til den fælles SUPERMAX I/O bus. SIOC'en kan således adressere 16 Mbyte i 15 enheder som er tilsluttet SUPERMAX bussen.

## 2. Basale hardware dele.

SIOC'en består af følgende basale hardware dele:

Mikro-processor. 8085AH-2.  
Dynamisk RAM lager, 4164-15 og EPROM, 2764.  
Memory mapping unit. (Adresse oversættelse)  
Interrupt del.  
I/O bus interface.  
Serielle kanaler.  
Baud rate generatorer.  
Serielt interface.  
Parallel printer port.  
Klok og reset.  
Fejl detektering.

Se oversigt diagram i afsnit 25.

De enkelte dele og sammenhængen mellem disse vil blive beskrevet nærmere i det følgende.

### 3. Cycles.

CPU'en er en 5 MHz 8085. CPU'en udfører følgende basale cycles:

1. Lager cycle.
2. I/O cycle.
3. Interrupt acknowledge cycle.(Idle)

Ad 1.

En lager cycle kan opdeles i flg. special tilfælde.

1. En access til PROM.
2. En access til det dynamiske RAM lager.
3. En access til den fælles SUPERMAX bus.

Ad 2.

En I/O cycle er en cycle hvor 8085's I/O instruktioner benyttes. (IN + OUT)

Ad 3.

En interrupt acknowledge cycle udføres som en reaktion på genkendelsen af et interrupt. Hvis interruptet er et 8085 INTR udføres acknowledge cyclen ved at INTA signalet aktiveres og 8085 læser en RST instruktion ind fra databussen. Hvis et andet interrupt signal benyttes (TRAP, RST7.5, RST6.5, RST5.5) udføres en acknowledge cycle uden aktivering af INTA, en idle cycle.

Udover forskellige CPU cycles kan kortet udføre endnu en type cycle. En anden enhed på SUPERMAX bussen kan udføre lager cycles i SIOC'ens dynamiske RAM lager.

Når en enhed udføre en lager cycle i anden enhed er førstnævnte den aktive enhed og sidstnævnte den passive enhed. Det er den aktive enhed der tager initiativ til en lager cycle og udføre denne ved at accesse den passive enheds lager.

SIOC'en kan udføre flg. cycles som passiv enhed:

1. Byte read and write.
2. Word read and write.
3. Read modify write. (Byte)

SIOC'en kan udføre flere cycles parallelt. I/O og interrupt cycles kan udføres samtidigt med at SIOC'en er en passiv enhed i en bus access. Ved lager cycles udføres en cycle adgangen.

Initiering af CPU cycles:

8085 udfører en cycle når ALE er lav. En til en cycle hørende adresse er tilgængelig før cyclen startes ved at ALE går lav.

CPU cycles adskilles af ALE pulser. Når ALE går fra høj til lav startes en cycle. Når ALE går fra lav til høj afsluttes en cycle. Status om en cycle er tilgængelig før ALE går lav. Ligeledes er 8085 adressen tilgængelig før ALE går lav. En cycle startes ved at status og adressen dekodes mens ALE er høj. Når ALE går lav er adressen og status dekodet og passende styresignaler kan aktiveres.

Den nærmere håndtering af cycles beskrives i afsnit 20 til 22.

#### 4. Bus arkitektur.

8085's multipleksede bus kaldes ADC(15:8) og DBC(7:0).

ADC og DBC splittes til følgende bus'er:

1. En lagerbus kaldet ADM og DBM.
2. En adresse bus AIO.
3. En I/O data bus til div. I/O dele. DIO.
4. En I/O data bus til LSI komponenter. DIOL.

Ad 1. ADM(15:8) er en simpel buffering af ADC(15:8), E1. ADM(7:0) fås ved at demultiplekse 8085's adresse/databus DBC(7:0) vha. registeret E2. DBM(7:0) er en simpel buffering af DBC(7:0). E3.

Ad 2. AIO(7:0) er en simpel buffering af ADC(15:8). N1.

Ad 3. DIO(7:0) er en simpel buffering af DBC(7:0). N2.

Ad 4. DIOL(7:0) er en simpel buffering af DBC(7:0). N3.

Se oversigt diagram i afsnit 25.

Lagerbus'en (ADM og DBM) giver forbindelsen til og fra SUPERMAX fællesbus'en. Se oversigt diagram i afsnit 25. Adressen fra fællesbus, AD(15:1) latches i registerne D2 og D4. Adressen til fællesbus'en drives af D1 og D3, AD(9:1). Data fra fællesbussen drives af D7 og D8, DB(15:0). Data til fællesbussen drives af registeret D6, DB(7:0) og driveren D5, DB(15:8). AD(33:10) drives af memory mapperen. Se afsnit 7.

ADM(0) er speciel idet SUPERMAX bussen ikke indeholder dette signal. ADM(0) multiplekseres mellem to signaler med multiplekseren E5. Ved interne cycles drives ADM(0) af AOC, E2-12, og ved passive cycles drives ADM(0) af AOP fra PAL S50.

## 5. Lager konfiguration.

Det interne lager består af 64kx9bit dynamisk RAM (4164-15) og en 8 kbyte EPROM (2764-3).

EPROM'en deles i to logiske dele. Den ene halvdel er placeret i 8085 adresserum fra adresse 0x0000 til 0x0fff. Den anden halvdel er placeret fra adresse 0xf000 til 0xffff. EPROM'en i adresse 0x0000 til 0x0fff selekteres med 8085 signalet SOD. Efter reset er EPROM'en i dette adresse område selekteret.

Adresse området fra 0x1000 til 0xbfff er altid RAM. Adresseringen af området 0xc000 til 0xffff afhænger af indholdet af et "address map register, AMR".

AMR(7:0), Address Mapping Register.

Access mode: I/O write.

I/O address : 0xBB

Upon power up reset: AMR(7:0) = 0

AMR(0) : /EDEB, Enable DEBugger.

AMR(0) = 0 : The EPROM is enabled in address area 0xe000 to 0xffff.

AMR(0) = 1 : The RAM is enabled in address area 0xe000 to 0xffff.

AMR(1) : ENBO, Enable Bus Out.

AMR(1) = 0 : The RAM is enabled from adr. 0xc000 to 0xe000 or 0xffff depending on AMR(0).

AMR(1) = 1 : Any memory access in area 0xc000 to 0xe000 or 0xffff will be performed as an I/O bus transfer.

AMR(2) : ENBI, Enable Bus In.

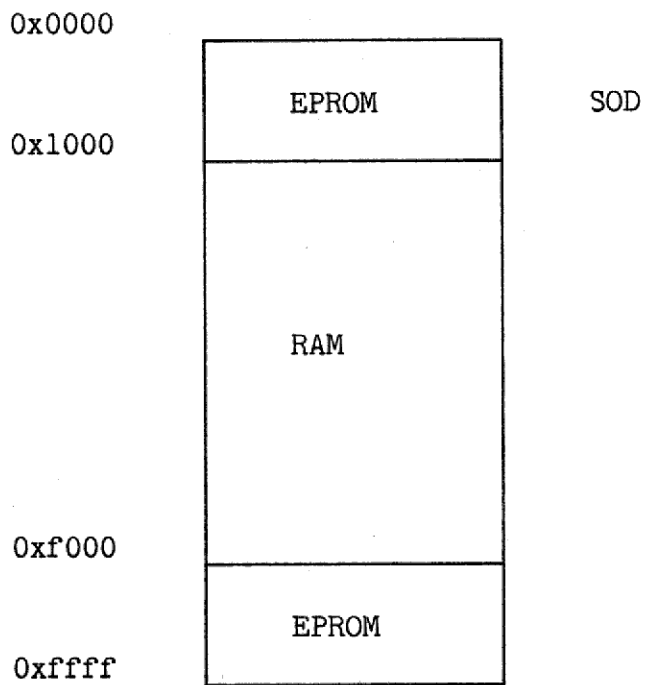
AMR(2) = 0 : An access to the SIOC module from the I/O bus is not answered.

AMR(2) = 1 : All locations in the 64 kbyte RAM memory are available from the I/O bus.

AMR(7:3) are not used.

Memory layout after reset:

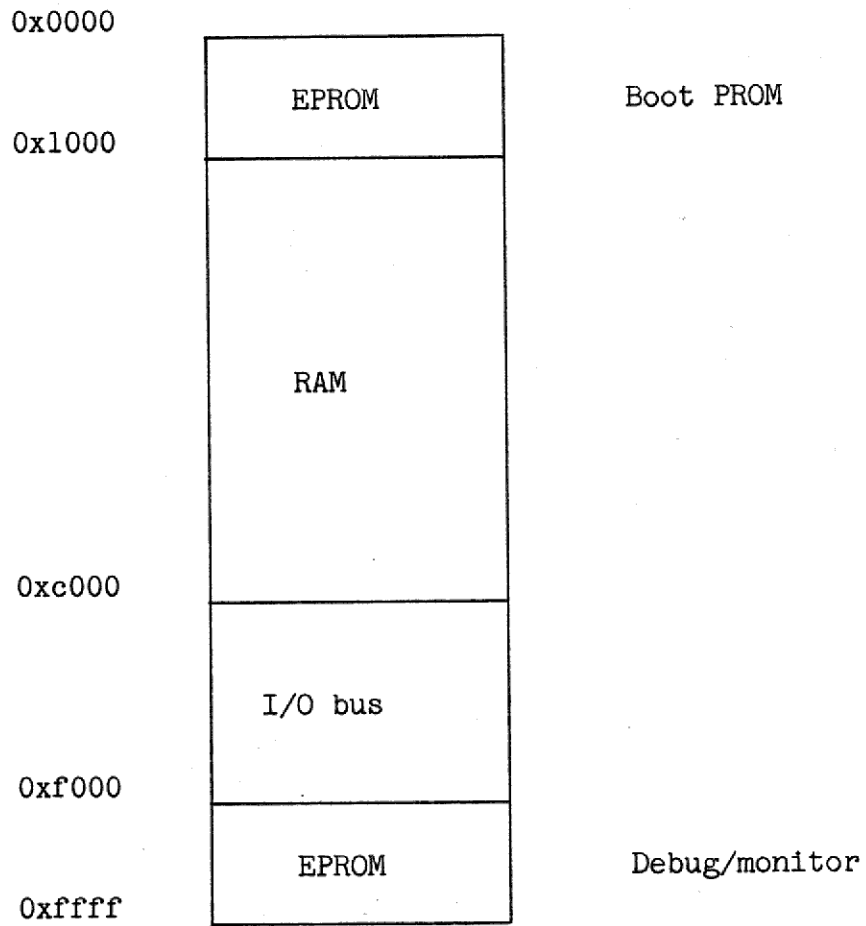
AMR(2:0) = 000





Memory layout for program development.

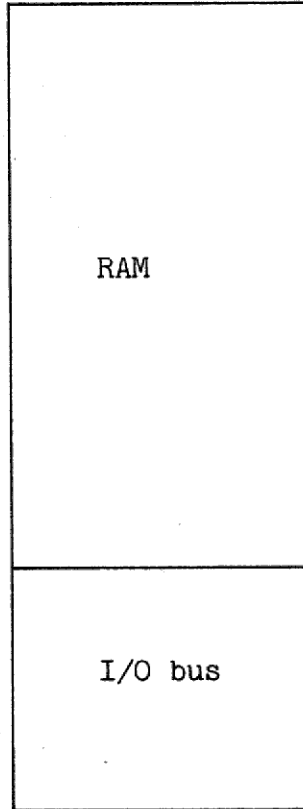
AMR(2:0) = 110



Memory layout for normal use.

AMR(2:0) = 111

0x0000



0xc000

0xffff

## 6. Internt lager.

### DRAM:

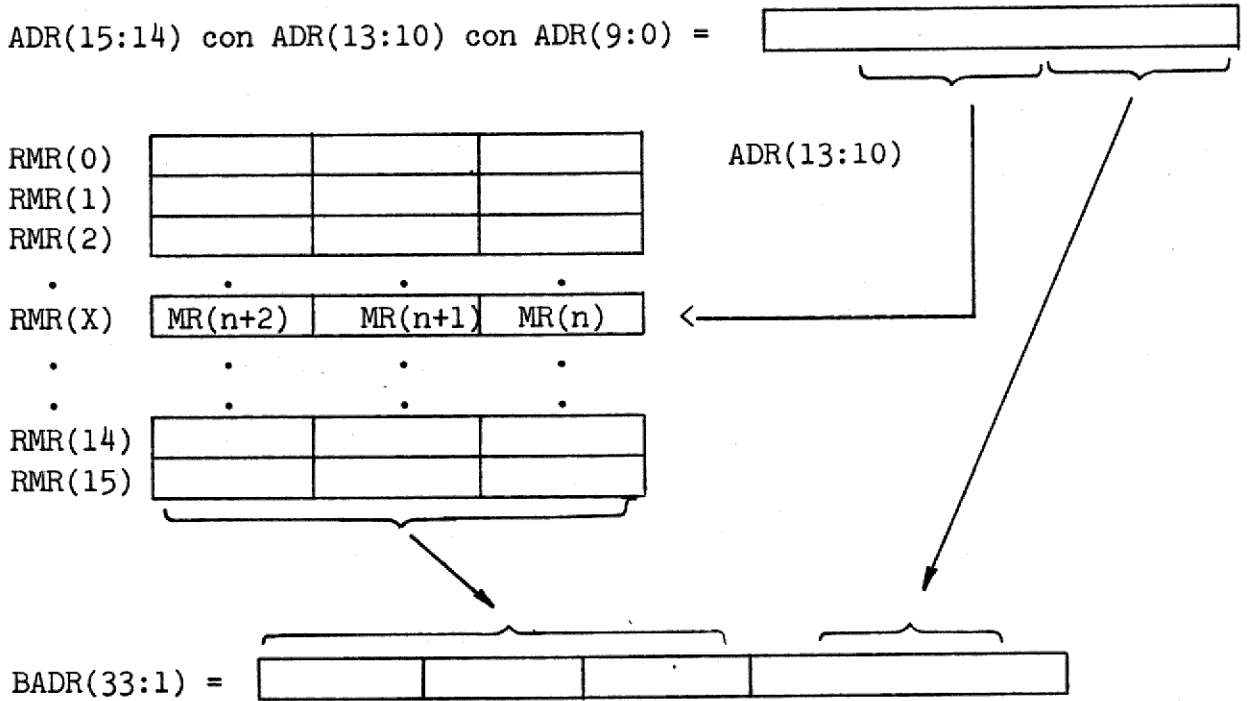
Det dynamiske RAM lager er opbygget af 9 stk 4164-15, 9x64k. Lageret er 8 bit bredt + 1 paritet bit. Lager adressen til 4164 er en 8 bit multiplekset adresse. Den multipleksede adresse drives af driver C1, C2 og C3. RAS adressen er ADM(15:8) og CAS adressen er ADM(7:0). Den sidste del af adressen er refresh tælleren. 4164 har adskilt data ind og ud. Data til lageret er DBM(7:0). Data ud fra lageret latches og buffres med latchen C4. Ved skrivning i lageret genererer C5 en paritets bit. Ved læs i lageret checkes data for paritets fejl. Paritets dekodeeren/enkodeeren (LS280) C6 genererer paritet for de 9 læste databit. Pariteten checkes af FF L2. L2 er med en nand gate koblet som en JK-FF. RW signalet til lageret, paritetscheck signalet og FF'ens indhold samples med lagerstyringssignalet LDM. Hvis der er fejl sættes signalet PE. Hvis PE er sat lukkes gaten L1 således at SIOC'en ikke kan give DTACK til den aktive enhed i en busoverførsel. RW signalet til lageret sættes af CAS med FF L3 da RWM signalet kan ændre værdi på bagkanten af en cycle.

### EPROM:

EPROM, E4, benyttes som boot prom og som lager for testprogrammer. Eprom'en er en 2764-3, 8kx8. EPROM'en adresseres af ADM(12:0). Data er tilsluttet DBM(7:0). Select genereres af PAL S10.

7. Memory mapper.

Når SIOC'en er den aktive enhed i en I/O bus access udvides den 16 bit 8085 adresse vha. en memory mapper. Med memory mapperen kan 8085 adressere alt lager tilknyttet SUPERMAX I/O bussen. Når AMR er sat op og ADC(15:14)=11 (binært) og 8085 udføre en lagercycle vil cyclen blive udført som en cycle til I/O bussen. SIOC'en udfører da en aktiv bus cycle. ADC(13:10) bliver brugt til at selektere 1 ud af 16 indgange i memory mapperen. Hver indgang til memory mapperen kaldes et resulterende memory map register(RMR). Hvert RMR består af 3 map registre, MR. Hvert MR er 8 bit bredt.



Den resulterende adresse er I/O bus adressen: BADR(33:1) som består af RMR adressen plus de 10 mindst betydende bit af 8085's adresse.

BADR(33:0) = RMR(X) con ADR(9:0)

BADR(33:0) = MR(n+2)(7:0) con MR(n+1)(7:0) con MR(n)(7:0) con ADR(9:0)

MR(47:0)(7:0), Map Register. 48 map registers, each 8 bits wide.

Access mode: I/O write and I/O read.

Upon power up reset: Contents undefined.

I/O addresses:

	MR(n+2)	MR(n+1)	MR(n)
RMR(0)	0xC2	0xC1	0xC0
RMR(1)	0xC6	0xC5	0xC4
RMR(2)	0xCA	0xC9	0xC8
RMR(3)	0xCE	0xCD	0xCC
RMR(4)	0xD2	0xD1	0xD0
RMR(5)	0xD6	0xD5	0xD4
RMR(6)	0xDA	0xD9	0xD8
RMR(7)	0xDE	0xDD	0xDC
RMR(8)	0xE2	0xE1	0xE0
RMR(9)	0xE6	0xE5	0xE4
RMR(10)	0xEA	0xE9	0xE8
RMR(11)	0xEE	0xEd	0xEC
RMR(12)	0xF2	0xF1	0xF0
RMR(13)	0xF6	0xF5	0xF4
RMR(14)	0xFA	0xF9	0xF8
RMR(15)	0xFE	0xFD	0xFC

Et eksempel:

```

MVI A,30
OUT 0C2      ;SETUP MR(2)
MVI A,89
OUT 0C1      ;SETUP MR(1)
MVI A,23
OUT 0C0      ;SETUP MR(0)
LDA C056     ;Læs en byte fra en enhed på I/O bussen.

```

I/O bus adressen er :

00 1100 0010 0010 0100 1000 1100 0101 0110 = 0x0c2248c56  
uu uuaa aaaa ssss

Unit number: 0x3  
Address space number: 0x2  
segment number: 0x2  
address: 0x48c56

Mapperen er opbygget af 6 statiske RAMs og 6 bufere. Lageret er, set fra 8085's side, organiseret som 3 blokke hver bestående af 16 bytes. De statiske RAM lagre her adskilt data ind og data ud og indeholder 16 ord a 4 bit. Adressen til lageret er AIO(5:2). Data ind til lagrene er tilsluttet DIO(7:0). Data ud fra lagrerne er for hver RAM kredsløst ført til to buffere. Den ene buffer driver DIO(7:0), således at indholdet af lagrerne kan læses, og den anden buffer driver SUPERMAX IO bussen når SIOC'en er den aktive part i en busaccess. Som skrive signal til RAM'erne benyttes IOW. Chip-select til RAM'erne genereres af PAL S10. CS genereres når der læses eller skrives programmeret i kredsene (IO instruktioner) og når der er access til SUPERMAX bussen.

## 8. CPU.

CPUen er en 5 MHz 8085. Nedenstående er en kort beskrivelse af de enkelte kontrol signaler. Adresse og data bus er beskrevet i afsnit 4.

### Input:

#### Klok.

8085 drives af en 10 MHz klok vha. to invertere, H1.

#### Reset.

Reset til 8085 genereres fra SUPERMAX IO bussen eller fra fejlhåndterings kredsløbet internt på kortet.

#### Hold.

Hold/holda funktionen benyttes ikke. Hold er ført til stel.

#### Ready.

Ready genereres af flere kilder. Omtales senere.

#### Interrupt.

Alle interrupt pinde undtagen RST5.5 benyttes. Se afsnit 9.

### Output:

#### HOLDA.

Benyttes ikke.

#### S0.

Benyttes ikke.

#### S1.

Benyttes til tidlig dekodning af læse/skrive signal, PAL S30.

#### SOD.

Benyttes til enable/disable af boot prom.

#### SID.

SOD kan læses på SID.

**RESET.**

Reset benyttes til at give reset til div. logik på kortet.

**CLK.**

CLK out, 5MHz, benyttes som input til G2 som giver flere basale klok signaler. Benyttes også til READY timing.

**ALE.**

Basalt styresignal som benyttes flere steder. Buffer H1 giver ALE i en sand og inverteret version.

**IOM.**

Basalt styresignal som benyttes flere steder.

**RD, WR.**

De to generelle skrive/læse signaler gates sammen for at give passende signal til PAL S30. RD bufferes af H3.

**INTA.**

Benyttes af interrupt delen og til styring af databus buffere vha. PAL S30.



## 9. Interrupt.

Der skelnes mellem to typer interrupt, interne og eksterne interrupts. Interrupt genereret internt på SIOC'en er interrupt fra USART'er etc. Eksterne interrupts er interrupts som aktiveres ved at andre enheder på SUPERMAX bussen skriver i udvalgte dele af SIOC'ens lager.

8085 har flg. interrupt indgange.

TRAP : Benyttes af fejl-overvågningskredsløb. Se afsnit 16.

RST7.5 : Forbundet til den programmerbare timer G5. Benyttes som realtime klok.

RST6.5 : Service port interrupt. Forbundet til USART J1, Rxrdy pin.

RST5.5 : Ikke brugt.

INTR : Generelt interrupt. I interrupt acknowledge cyclen henter 8085 en instruktion på databussen. SIOC'en benytter RST7-0 instruktioner. INTR benyttes af både interne og eksterne interrupts.

Interne interrupts og INTR:

Rxrdy fra de 8 USART'er (R1, R4, R7, R10, R11, R14, R17 og R18) er forbundet til 8 gates som benyttes til at maske de 8 signaler ud med. Masken skrives i registeret A2. Udgangene af de 8 gates er OC. og er forbundet til STRAP sokkel A6. Vha. A6 kan de 8 maskede Rxrdy interrupts strappes til et eller flere af de 8 RST niveauer.

Txrdy fra de 8 USART'er (R1, R4, R7, R10, R11, R14, R17 og R18) er forbundet til 8 gates som benyttes til at maske de 8 signaler ud med. Masken skrives i registeret A1. Udgangene af de 8 gates er OC. og er forbundet til STRAP sokkel A7. Vha. A7 kan de 8 maskede Txrdy interrupts strappes til et eller flere af de 8 RST niveauer.

Ved brug af RS-422 interface for de serielle kanaler 6 og 7 kan der genereres særlige interrupts. Se afsnit 9. Disse interrupts (IND1, IND2, IND3 og IND4) kan strappes til et eller flere af de 8 RST niveauer vha. strap sokkel A8.

Alle kilder til de 8 RST interrupts er ført til en maske således at et hvert af de 8 niveauer kan maskes ud vha. maskeregisteret A3.

De 8 maskede niveauer løber til en transparent latch, A4, Udgangen af latchen går ind i en 8-3 prioritets enkoder, A9. Hvis blot en af indgangene til A9 er aktiv aktiverer A9 pin 15 som er forbundet til 8085's INTR indgang. Når 8085 udfører interrupt acknowledge cyclen aktiverer 8085 INTA. INTA signalet lukker latchen A4 således at indgangene på prioritets enkoderen er stabile. Derved haves også et stabilt output fra enkoderen. Enkoderens udgange bestemmer hvilken RST instruktion der gates ud på databussen DIO(7:0) gennem bufferen A5.

### Externe interrupts og INTR.

Externe interrupts genereres når en enhed på SUPERMAX bussen skriver at passende sted i SIOC'ens lager. Lagerområdet der benyttes til interrupt er 0x0080 til 0x00ff. Området er delt op i 8 blokke a 16 bytes. Skrivning i en blok giver anledning til et interrupt på et RST niveau.

0x0080	RST0
0x0090	RST1
0x00a0	RST2
0x00b0	RST3
0x00c0	RST4
0x00d0	RST5
0x00e0	RST6
0x00f0	RST7
0x0100	

En skrive cycle kan være en af flg. typer cycles:

1. Word write.
2. Byte write.
3. Read-modify-write.

Hardwaren er implementeret på flg. måde:

Lager adressen ADM(15:7) og skrive signalet dekodes med OC-gates B1 og B2. Når SIOC'en svarer den aktive part i en buscycle med DTACK fås et aktivt signal på den adresserbare latch B4 som afhængigt af sin dataindgang sætter/clearer en af sine 8 FF. De 8 FF i B4 adressere vha. multiplekseren B3. Når SIOC'en er den passive part i en busaccess (CONI aktiv) adresseres B4 af ADM(6:4), desuden sættes data indgangen på B4 høj. Når SIOC'en kører lokalt, adresseres B4 af AIO(2:0) og dataindgangen sættes lav. Når SIOC'en aktiverer CSINT, input til B5-05, aktiveres den adresserbare latch og den adresserede FF cleares. CSINT genereres af PAL S10. RST0 FF cleares med instruktionen OUT 00, RST1 FF cleares med instruktionen OUT 01 etc. Udgangen fra hver af de 8 FF er ført til oc drivere som hver aktiverer et af de 8 RST niveauer.

#### 10. Klok.

Alle klokker genereres ud fra en 20 MHz klok, CP20. CP20 benyttes til styring af sekvensmaskiner og div. logik. CP20 deles ned til en 10 Mhz klok, CP10, som benyttes som input til 8085. 8085 deler denne klok ned til 5 MHz, CP5. CP5 er input til tælleren G2 som giver flg. klokker:

CP2.5: Input til alle USARTS.

CP1.5: Input til alle baudrate tællere.

CP0.8: Timing af refresh til det dynamiske RAM lager.

Desuden genereres sende/modtage klokker for service porten. Kan strappes til 9600, 4800, 2400 eller 1200 baud. Ved programmering af service porten kan disse frekvenser yderligere deles med fire.

Sende/modtage klokke til USART'er genereres af tre programmerbare tællere, G3, G4 og G5. Et klok-signal fra G5 benyttes til programmerbart timer-interrupt signal til 8085. (RST7.5)

#### 11. Ready.

Ready til 8085 drives af gate B7. Som input til B7 haves 4 signaler. FF B8 indsætter et wait state i hver 8085 cycle, uafhængigt af cycle typen. FBREADY aktiveres når SIOC er den aktive part i en SUPERMAX IO bus cycle. READYP aktiveres af PAL S70 når SIOC'en er den passive i en IO-bus cycle. Skifteregisteret B10 indsætter et antal wait-cycles enten ved refresh, ved at RFSH aktiveres, eller ved at PAL S70 aktiverer AWAIT. AWAIT bliver aktiveret når SIOC'en er passiv i en IO bus cycle.

#### 12. Adresse dekodning.

IO adressedekodning foretages primært af PAL S20, 01. 01 dekode styresignaler og AIO(7:3) til 7 chip selects. Pin 19,18 og 17 giver enable signaler til tre 3 til 8 dekodere som giver de endelige chip select signaler.

OEP er select af unit PAL.

CSSU er select af service port.

CSFS er select af error status registeret.

CSL er select af LSI databus, DIOL(7:0).

PAL S10, 02, dekode adresser og styresignaler. De 7 udgange benyttes til flg.: CSM1, CSM2 og CSM3 er select af SRAM i memory mapperen. Select gives både ved programmeret læsning/skrivning og når SIOC'en er aktiv i en bus overførelse.

CEPROM er select af EPROM'en.

FBRQ er request signal der aktiveres når SIOC'en er aktiv part i en IO bus cycle.

CSINT er clear-signal af eksterne interrupts.

DIRAM aktiveres når der ikke skal laves en lager cycle i DRAM, men i f.eks EPROM.

### 13. Parallel printer port.

Porten er en simpel output port til en centronics kompatibel printer. Data skrives i registeret P1 med select signalet CSC. CSC aktiverer også one-shot P3 som ved udløb trigger den anden one-shot som der ved giver en puls til printeren. Pulsen kaldes DATASTROBE, DS. DS kan strappes til at være aktiv høj eller lav. CSC clearer også FF P4, hvis udgang giver et ready signal som gated med busy fra printeren kan polles af programmellenet. ACK signalet aktiveres af printeren når data er taget. ACK sætter FF P4 som indikerer at printeren er klar til et nyt tegn. Signalet BUSY aktiveres typisk ved lineskift, paper out etc. BSY og ACK kan strappes til at være aktiv høje eller lave. Data bussen til printeren drives af en OC driver, P2. Både BSY fra printeren og BSY gated med indholdet af FF P4 kan læses i registeret N6. Alle interface signaler er ført til det 50 polede fladkabel stik.

### 14. USARTS.

De 8 serielle linier består hver af en USART, 8251A, og linie drivere/receivere. De 8 linier omtales som kanaler, kanal 0 til kanal 7. Kanal 0 til 5 er asynkrone RS-232 kanaler som hver får baudrate fra de programmerbare timere. TxClk og RxClk er forbundet sammen for hver kanal. Som linie drivere benyttes 75188 og som receivere benyttes 75189A. Begge typer monteres altid i sokler. Receiverens tærskelspænding kan evt. ændres ved at montere en modstand mellem receiverens control pin og VCC. Denne modstand er normalt ikke monteret. Kanal 6 og 7 har nogle ekstra egenskaber. Kanalerne kan køre både asynkront og synkront ligesom man kan vælge mellem RS-422 og RS-232 interface. Valg mellem RS-422 og RS-232 er programmerbart.

Valg af ekstern/intern klok skal strappes. De to kanaler er opbygget på samme måde.

#### Udgangs signaler fra USART.

Txd driver både RS-232 TXD og RS-422 T(a) og T(b).

DTR driver både RS-232 DTR og RS-422 C(a) og C(b).

RTS driver RS-232 RTS.

#### Indgangs signaler til USART.

Rxd drives af multiplekser (S1, T1) enten RS-232 RXD eller RS-422 R(a) og R(b).

DSR drives af multiplekser (S1, T1) enten RS-232 DSR eller RS-422 I(a) og I(b).

Txclk drives af multiplekser (S1, T1) enten RS-232 TXCLK eller RS-422 S(a) og S(b).

Rxclk drives af multiplekser (S1, T1) enten RS-232 RXCLK eller RS-422 S(a) og S(b).

De tre interface klok signaler TXCLK, RXCLK og S drives altid af en af de programmerbare timere. Valget mellem ekstern/intern klok vælges med straps: SS1, SS2, ST1 og ST2. Hvis intern klok vælges vil SIOC'en samtidigt drive klok signaler i interfacen.

CTS drives af multiplekser (S7, T7) enten RS-232 CTS eller RS-422 I(a) og I(b) eller lagt til sit aktiv lave niveau.

Der er to særlige interrupts fra hver af de to kanaler. Interrupts sættes når RS-422 signalet I ændrer niveau. Interruptene sættes ved at I's flanker sætter FF S2 eller T2. FF S2 og T2 driver OC buffere som er ført til strap sokkel A8 således at interruptene kan forbindes til et hvilket som helst RST niveau. Indholdet af registeret N6 vælger mellem

RS-232/RS-422, CTS lav eller fra interface samt enable/disable af interrupt FF'ene.

#### 15. Reset.

8085 får reset fra tre kilder.

1. Switch monteret på sokkel Q3, FF Q2.
2. Reset fra bus aktiv, M7 og Q8.
3. Err. signalet aktivt pga. detektion af en fejl. Reset puls aktiveres af one-shot Q1.

8085 reset out giver reset til al logik på kortet. Pga. muligheden for detektion af fejl er der enkelte dele der kun resettes af signalet RESETIN, der er identisk med IO bus reset.

#### 16. Trap.

Trap interruptet til 8085 sættes af 5 kilder.

1. Switch monteret på sokkel Q3, FF Q2.
2. Ved timeout, FF Q4.
3. Ved modtagelse af en bus error når SIOC'en er aktiv i en buscycle. FF Q7.
4. Ved aktivering af PE (paritets fejl) og SIOC'en er passiv. Buserror aktiveres til aktive. FF Q5.
5. Ved aktivering af ERR signalet gives TRAP i første instruktion efter reset fra one-shot Q1. FF Q4 kan sættes programmeret.



### 17. Time out.

Alle cycles overvåges af et time out kredsløb. Når en enhed er aktiv er det den aktives time out kredsløb der overvåger cyclen og evt. afbryder den. Da alle cycles initieres af 8085 benyttes ALE signalet som trig signal til time out kredsløb. Kredsløbet er en simpel retrigbar monostabil multivibrator som sætter FF Q4 hvis den løber ud og dermed detektere en TOUT. Indholdet af FF Q4 kan læses i registeret N6.

### 18. Buserror.

Den passive enhed svarer en access med DTACK eller BUSERROR. Hvis buserror modtages af SIOC'en genereres et TRAP interrupt af FF Q7. Hvis FF Q7 er enabled af signalet på clear indgangen, EN/CLBUSI, kan gyldig status læses i registeret N6.

SIOC passiv i en bus access.

Hvis den aktive læser i SIOC'ens lager og en paritetsfejl detekteres aktiveres buserror signalet i bussen. Hvis FF Q5 er enabled af clear signalet EN/CLBUSO aktiveres TRAP signalet og status kan læses i N6.

### 19. Error.

FF Q4 starter veldefineret op efter reset. Q4 aktiverer signalet ERROR i bussen. Q4 kan deaktiveres af programmet vha. signalerne ER og SERROR fra registeret N4. Hvis en paritets fejl detekteres sættes Q4 og ERROR er dermed aktiveret. ERROR kan fjernes af bus signalet ERRORRESET.

## 20. Lager cycles.

8085 udfører en cycle når ALE er lav. En til en cycle hørende adresse er tilgængelig før cyclen startes ved at ALE går lav. CPU cycles adskilles af ALE pulser. Når ALE går fra høj til lav startes en cycle. Når ALE går fra lav til høj afsluttes en cycle. Status om en cycle er tilgængelig før ALE går lav. Ligeledes er 8085 adressen tilgængelig før ALE går lav. En cycle startes ved at status og adressen dekodes mens ALE er høj, i PAL S10. Når ALE går lav er adressen og status dekodet og passende styresignaler kan aktiveres, af PAL S30. Hvis cyclen er en access til det dynamiske lager aktiveres signalet MEMA. MEMA synkroniseres til CP20 og er input til PAL S70. S70 arbitrerer mellem de aktive enheder der vil have adgang til lageret. PAL S70 starter sekvensmaskinen PAL S40 som udfører selve lager cyclen.

## 21. Aktive cycles.

PAL S10 dekoder løbende 8085's adresse. Hvis address mapping registeret er sat rigtigt op og adressen angiver en access til SUPERMAX IO bus sker flg: PAL S10 aktiverer DIRAM og stopper dermed en lager access til det lokale dynamiske lager. PAL S10 enabler de statiske RAM lagrer i memory mapperen. PAL S10 aktiverer FBRQ signalet. Signalet samples i FF M4 når en 8085 cycle starter. Sample signalet er /ALE. Udgangen af FF M4 angiver at der ønskes udført en access til IO bussen. Signalet stopper 8085 ved at aktivere ready signalet. Signalet sætter datastrobes afhængigt af A0, FF M5. Signalet synkroniseres til IO bus klokken og sætter request til prioriterings PAL'en M1. En prioritering finder sted på vanlig vis og når SIOC'en kan udføre sin dataoverførsel aktiverer PAL M1 ASB. ASB bufres med gatene M6. Signalet kaldes CONO. CONO sætter FF M3 hvis udgang benyttes til enable af div. adresse/data/control signaler. CONO sætter også signalet ACCESS som først deaktiveres af næste ALE puls. ACCESS er således aktiv fra CONO aktiveres til 8085 cyclen er udført. Cyclen afsluttes af et af tre signaler. Hvis dataoverførslen er gået godt aktiverer den passive enhed DTACK. Hvis der er fejl i overførslen aktiverer den passive buserror. Hvis dtack eller buserror ikke aktiveres aktiverer

SIOC'en selv et time out signal, TOUT. Alle tre signaler går ind på gate M8 hvis udgang clearer FF M5. M5's udgang fjerner request til PAL M1 og dataoverførslen er afsluttet. Samtidigt fjernes vent til 8085. Hvis cyclen er en læsning vil udgangen af M5, signalet DT (DTACK), latches data i en af latchene D7, D8 vha. PAL S60, LECI og UECI. SIOC'en afslutter dataoverførslen på IO bussen lige så snart data er tilgængelige, mens 8085 først tager data og afslutter cyclen senere.

## 22. Passive cycles.

En enhed er aktiv og ønsker at adressere SIOC'en. Den aktive gater unit adressen ud på IO bussen på en busklok. Adressen dekoder af unit pal 11. Hvis unit nummeret er ok. sættes FF I4 på næste busklok. Signalet kaldes FBAC. FBAC synkroniseres og er input til PAL S70. PAL S70 aktiverer signalet CONI som enabler bus drivere etc. PAL S70 starter de to PAL'er S40 og S50. S40 udfører selve lager cyclen. S50 bestemmer typen af lager cycle og giver DTACK til den aktive enhed. En passiv cycle afsluttes når den aktive deaktiverer IO bus signalet ASB. Mulige passive cycles:

1. Word read og write.
2. Byte read og write.
3. Read-modify-write.

## 23. Beskrivelse af PAL'er.

### PAL S10, 02.

CSM1 : Select af memory mapping unit. Aktiv både ved accesser til IO bus og ved skrivning/læsning i mapperen.

CSM2 :

CSM3 :

CEPROM : Chip select til EPROM.

DIRAM : Aktiv når lager access ikke er til det dynamiske lager.

FBRQ : Aktiv når lager cyclen er en access til SUPERMAX IO

bus.

CSINT : Clear af eksterne interrupts.

PAL S20, O1.

IO adresse dekodning.

CSSU : Chip select til service port USART.

CSL : Styring af LSI databus driver.

CSFC : Chip select til register, N6.

OEP : Chip selelet til Unit no. PAL.

CSC : Select af baudrate generatorer. Dekodes yderligere af LS138, 05.

CSO : Select af div. registerer. Dekodes yderligere af LS138, 04.

CSU : Select af USARTs. Dekodes yderligere af LS138, 03.

PAL S30, H7.

Dekodning af 8085 styresignaler til lagerstyringssignaler og bus buffer styresignaler.

IOR : Read signal til IO enheder.

IOW : Skrive signal til IO enheder.

RW : Skrive læse signal til lager.

MEMA : Aktiv når der er access til det dynamiske lager.

DBG3 : Styring af databus buffer E3.

DBG : Styring af databus buffere N2 og N3.

ADG : Styring adresse drivere E1 og E2.

DIR5 : Styring af databus buffer N2 og N3 's retning.

PAL S40, K2.

Styring af det dynamiske lager.

RAS : Inverteret RAS puls til det dynamiske lager.

CAS : CAS puls til det dynamiske lager.

MSEL : Signal til multipleksning af lageradressen.

ACK : Indikere at en lagercycle er igangsat.

LDM : Latch af data fra lageret.  
 RFSH : Styling af refresh adresse.

Sekvensmaskinen kan udføre tre typer lager cycles.

1. Refresh cycle. (RAS only refresh)
2. Normal lager cycle, dvs. RAS, ROWEN, CAS etc.
3. Page mode lager cycle, som er en normal lager cycle hvor CAS deaktiveres og aktiveres uden at RAS deaktiveres.

Sekvensmaskinen starter op i tilstand T17. Efter reset hoppes til tilstand T0. I T0 ventes på start af lagercycle eller start af refresh cycle. Hvis refresh cycle startes hoppes til T9. Fra T13 hoppes til T14. T14 og T15 giver lageret passende precharge. Fra T0 startes en lager cycle hvis START går aktiv og refresh request ikke er aktiv. I tilstand T16 ventes på status linierne S0 og S1. Hvis cyclen er en word eller RMW hoppes til T7 hvorved lageret udfører en pagemode cycle. Fra T16 hoppes til T14 hvis lager accessen er færdig. Start signalet aktiveres af PAL S70. Status signalerne S0 og S1 aktiveres af PAL S50.

#### PAL S50, K3.

Styling af passive cycles i det dynamiske lager.

DTACK: Dtack til aktiv enhed når SIOC'en er passiv.  
 S1 : Angiver lagercycle typen.  
 S0 : do.  
 UDG : Styling af databus buffer fra IO bus.  
 LDG : do.  
 LDO : Latch af data ved læsning af word.  
 A0 : Adresse ADM(0) ved passive cycles.

Sekvensmaskinen styrer tre former for passive cycles:

1. Word cycles.
2. Byte cycle, UDS.
3. Byte cycle, LDS.

Specialtilfælde af punkt 2. og 3. er read-modify-write cycles.

Sekvensmaskinen starter op i tilstand T15. Efter reset hoppes til tilstand T0. I T0 ventes på start signal. Når start aktiveres hoppes til T1, T5 eller T9 afhængigt af de to datastrobe signaler UDS og LDS. Ved hop til T1 eller T5 udføres byte operation. Ved hop til T9 udføres word operation.

#### Byte operation:

I T1 startes det dynamisk lager af PAL S70. I T1 sættes buffer styringssignalerne LDG, UDG og A0. Når lageret er godt igang modtages signalet ACK fra PAL S40 og der hoppes til tilstand T2. I T2 aktiveres DTACK mens styresignalerne fastholdes. I T2 ventes på aktivering/deaktivering af ASB og datastrobe. Hvis cyclen er færdig og afsluttes med aktivering af ASB hoppes til tilstand T14. Hvis datastrobe signalet deaktiveres hoppes til T3. I T3 kan cyclen afsluttes af ASB eller fortsættes ved at datastrobe går aktiv igen. Hvis datastrobe aktiveres hoppes til T4 hvorved der udføres en RMW. I tilstand T14 sættes S0 og S1 således at PAL S40 vender tilbage til tilstand T0 og er klar til en ny lager cycle.

#### Word operation:

I T9 startes det dynamiske lager af PAL S70. I T9 sættes buffer styringssignaler LDG, UDG og A0. Når lageret er godt igang modtages ACK signalet fra PAL S40 og der hoppes til tilstand T10. I tilstandene T10, T11 og T12 ændres adressen A0 og data fra det dynamiske lager latches i registeret D6. Data latches uafhængigt af om cyclen er en læse eller skrive cycle. I T12 starter PAL S40 på en page mode lager cycle. Når lageret er færdig med den cycle modtages ACK signalet og DTACK aktiveres i T13. I T13 ventes på afslutning af cyclen. Cyclen afsluttes af ASB.

#### PAL S60, K4.

Styring af div. buffere mod SUPERMAX IO bus.

RWM : RW signal til lager.  
 LDGI : Enable af databusbufer fra IO bus til DBM.  
 UDGI : Do.  
 LDGO : Enable af databus buffer fra DBM til IO bus.  
 UDGO : Do.  
 LECI : Latch af data ind fra IO bus til DBM.  
 UECI : Do.

### PAL S70, K1.

Arbitrering af cycles.

STARTM : Start af PAL S40.  
 STARTB : Start af PAL S50  
 CONI : Indikerer at en passiv cycle udføres. Gate af buffere.  
 READY : Vent signal til 8085.  
 AWAIT : Vent signal til 8085.

Sekvensmaskinen prioriterer og styrer tre typer cycles:

1. 8085 cycles i det dynamiske lager.
2. Passive cycles i det dynamiske lager.
3. DMA cycles i det dynamiske lager.

Sekvensmaskinen starter op i tilstand T0. Tilstandene T13, T14, T15, T8, T9 og T10 benyttes ikke i SIOC'en. Fra T0 hoppes til T6 hvis MEMA signalet aktiveres mens FBAC er inaktiv. MEMA aktiveres af PAL S30 når 8085 ønsker access til det dynamiske RAM lager. I T6 startes PAL S40 med signalet STARTM. PAL S40 aktiverer ACK signalet når lager cyclen er startet. Fra T6 hoppes til T7 hvor der ventes indtil 8085 deaktiverer MEMA og dermed er færdig med brug af lageret og lagerbussen. Fra T7 hoppes til T0.

Fra T0 hoppes der til T1 hvis FBAC er aktiv og CONO er inaktiv. At CONO er inaktiv sikrer at SIOC'en ikke både er aktiv og passiv i en bus cycle. I T1 aktiveres CONI og der

gives vent til 8058. CONI åbner for div. adresse/data bus drivere. CONI er aktiv i hele den passive cycle. I T2 er styresignaler og adressen stabile og PAL S50 startes af signalet STARTB. I T3 startes PAL S40 med signalet STARTM. Når lageret er startet hoppes til T4 hvor der ventes på at cyclen afsluttes. Cyclen er afsluttet når requestet FBAC er inaktivt. I T11 venter sekvensmaskinen på at lageret bliver helt færdigt inden en ny lager cycle evt. startes. Pga. page mode cycles i lageret står lageret længe med RAS og lageret skal have tid til precharge inden en ny cycle.

Udover ovennævnte PAL'er benytter SIOC'en en prioriterings PAL og en Unit no. PAL. Prioriterings-PAL'en er komponent M1. Unit no. PAL er komponent I1.



Components for SIOC board.

## \*ICs

NAME	DESC.	TYPE	POSITION
A1 (20 )	74 , 52 : 1	;;74ALS574	A 7,4
A2 (20 )	74 , 64 : 1	;;74ALS574	A 7,5
A3 (20 )	74 , 76 : 1	;;74ALS576	A 7,6
A4 (20 )	50 , 52 : 1	;;74ALS573	A 8,4
A5 (20 )	50 , 76 : 1	;;74LS541	A 8,6
A6 (16 )	4 , 28 : 1	;;STRAP	A 2,10
A7 (16 )	4 , 16 : 1	;;STRAP	A 1,10
A8 (16 )	26 , 16 : 1	;;STRAP	A 1,9
A9 (16 )	50 , 64 : 1	;;74LS148	A 5,8
A10 (14 )	4 , 52 : 1	;;74LS03	A 4,10
A11 (14 )	26 , 52 : 1	;;74LS03	A 4,9
A12 (14 )	4 , 40 : 1	;;74LS03	A 3,10
A13 (14 )	26 , 40 : 1	;;74LS03	A 3,9
A14 (14 )	26 , 28 : 1	;;74LS32	A 2,9
A15 (14 )	50 , 40 : 1	;;74LS32	A 3,8
B1 (14 )	146 , 52 : 1	;;74LS05	A 4,4
B2 (14 )	146 , 64 : 1	;;74LS05	A 5,4
B3 (16 )	74 , 88 : 1	;;74LS257	A 7,7
B4 (16 )	98 , 88 : 1	;;74LS259	A 7,6
B5 (14 )	98 , 76 : 1	;;74LS03	A 6,6
B6 (14 )	98 , 64 : 1	;;74LS05	A 5,6
B7 (14 )	146 , 88 : 1	;;74LS21	A 7,4
B8 (14 )	146 , 76 : 1	;;74LS74	A 6,4
B9 (14 )	170 , 98 : 1	;;74S74	B 8,3
B10 (16 )	122 , 76 : 1	;;74LS175	A 6,5
C1 (20 )	194 , 88 : 1	;;74ALS573	A 7,2
C2 (20 )	218 , 88 : 1	;;74ALS573	A 7,1
C3 (20 )	170 , 88 : 1	;;74ALS573	A 7,3
C4 (20 )	194 , 26 : 1	;;74ALS573	B 2,2
C5 (14 )	194 , 38 : 1	;;74LS280	B 3,2
C6 (14 )	194 , 50 : 1	;;74LS280	B 4,2

C7	(14 )	170 ,	76 : 1	;;74LS393	A 6,3
C8	(16 )	216 ,	10 : 1	;;4164-15	B 1,1
C9	(16 )	216 ,	18 : 1	;;4164-15	B 2,1
C10	(16 )	216 ,	26 : 1	;;4164-15	B 2,1
C11	(16 )	216 ,	34 : 1	;;4164-15	B 3,1
C12	(16 )	216 ,	42 : 1	;;4164-15	B 3,1
C13	(16 )	216 ,	50 : 1	;;4164-15	B 4,1
C14	(16 )	216 ,	58 : 1	;;4164-15	B 5,1
C15	(16 )	216 ,	66 : 1	;;4164-15	B 6,1
C16	(16 )	216 ,	74 : 1	;;4164-15	B 6,1
C17	(14 )	170 ,	64 : 1	;;74S00	A 5,3
D1	(20 )	218 ,	64 : 1	;;74LS541	A 5,1
D2	(20 )	218 ,	76 : 1	;;74ALS574	A 6,1
D3	(20 )	194 ,	64 : 1	;;74LS541	A 5,2
D4	(20 )	194 ,	76 : 1	;;74ALS574	A 6,2
D5	(20 )	122 ,	16 : 1	;;74LS541	A 1,5
D6	(20 )	146 ,	16 : 1	;;74ALS573	A 1,4
D7	(20 )	122 ,	28 : 1	;;74ALS573	A 2,5
D8	(20 )	146 ,	28 : 1	;;74ALS573	A 2,4
E1	(20 )	194 ,	14 : 1	;;74LS541	B 1,2
E2	(20 )	146 ,	100 : 1	;;74ALS573	A 8,4
E3	(20 )	146 ,	112 : 1	;;74LS645	A 9,4
E4	(28 )	180 ,	106 : 1	;;2764	A 8,2
E5	(14 )	194 ,	62 : 1	;;74LS00	B 5,2
E6	(14 )	216 ,	110 : 1	;;74LS32	B 9,1
F1	(16 )	170 ,	40 : 1	;;74LS189	A 3,3
F2	(16 )	170 ,	52 : 1	;;74LS189	A 4,3
F3	(16 )	194 ,	40 : 1	;;74LS189	A 3,2
F4	(16 )	194 ,	52 : 1	;;74LS189	A 4,2
F5	(16 )	218 ,	40 : 1	;;74LS189	A 3,1
F6	(16 )	218 ,	52 : 1	;;74LS189	A 4,1
F7	(20 )	170 ,	28 : 1	;;74LS540	A 2,3
F8	(20 )	194 ,	28 : 1	;;74LS540	A 2,2
F9	(20 )	218 ,	28 : 1	;;74LS540	A 2,1
F10	(20 )	170 ,	16 : 1	;;74ALS580	A 1,3
F11	(20 )	194 ,	16 : 1	;;74ALS580	A 1,2
F12	(20 )	218 ,	16 : 1	;;74ALS580	A 1,1

G1	(14 )	216 ,	98 : 1	;;7404	B 8,1
G2	(14 )	216 ,	86 : 1	;;74LS393	B 7,1
G3	(24 )	42 ,	88 : 1	;;8253	A 7,8
G4	(24 )	42 ,	112 : 1	;;8253	A 9,8
G5	(24 )	42 ,	14 : 1	;;8253	B 1,8
G6	(14 )	146 ,	98 : 1	;;74ALS74	B 8,4
H1	(14 )	146 ,	50 : 1	;;74S04	B 4,4
H2	(14 )	122 ,	50 : 1	;;74S04	B 4,5
H3	(14 )	122 ,	38 : 1	;;74S08	B 3,5
H4	(14 )	122 ,	74 : 1	;;74LS08	B 6,5
H5	(14 )	146 ,	62 : 1	;;74LS74	B 5,4
H6	(40 )	122 ,	14 : 1	;;8085	B 1,4
H7	(20 )	146 ,	38 : 1	;;16L8-S30	B 3,4
H8	(14 )	122 ,	62 : 1	;;74LS32	B 5,5
I1	(20 )	122 ,	52 : 1	;;16L8-SUXX	A 5,5
I2	(20 )	170 ,	26 : 1	;;74ALS574	B 2,3
I3	(20 )	146 ,	40 : 1	;;74ALS244	A 3,4
I4	(14 )	98 ,	52 : 1	;;74S74	A 4,6
I5	(14 )	122 ,	64 : 1	;;74LS00	A 5,5
J1	(28 )	138 ,	14 : 1	;;8251	C 2,4
J2	(14 )	170 ,	14 : 1	;;75188	C 2,3
J3	(14 )	146 ,	30 : 1	;;75189	C 3,4
J4	(16 )	170 ,	24 : 1	;;STRAP	C 3,3
K1	(20 )	170 ,	50 : 1	;;16R8-S70	B 4,3
K2	(20 )	170 ,	62 : 1	;;16R8-S40	B 5,3
K3	(20 )	170 ,	38 : 1	;;16R8-S50	B 3,3
K4	(20 )	170 ,	14 : 1	;;16L8-S60	B 1,3
K5	(14 )	194 ,	74 : 1	;;74S04	B 6,2
L1	(14 )	170 ,	74 : 1	;;74S00	B 6,3
L2	(14 )	170 ,	86 : 1	;;74S74	B 7,3
L3	(14 )	194 ,	86 : 1	;;74S74	B 7,2
M1	(20 )	98 ,	16 : 1	;;16R8-SPXX	A 1,6
M2	(20 )	98 ,	28 : 1	;;74LS641-1	A 2,6

M3	(14)	98	,	40	:	1	;;74LS74	A	3,6
M4	(14)	122	,	40	:	1	;;74S74	A	3,5
M5	(14)	74	,	40	:	1	;;74LS74	A	3,7
M6	(14)	74	,	28	:	1	;;74ALS37	A	2,7
M7	(14)	74	,	16	:	1	;;74LS14	A	1,7
M8	(14)	50	,	16	:	1	;;74LS21	A	1,8
M9	(14)	50	,	28	:	1	;;74LS32	A	2,8
N1	(20)	98	,	26	:	1	;;74LS541	B	2,6
N2	(20)	122	,	100	:	1	;;74LS645	A	8,5
N3	(20)	122	,	112	:	1	;;74LS645	A	9,5
N4	(20)	122	,	110	:	1	;;74LS273	B	9,5
N5	(20)	122	,	88	:	1	;;74LS273	A	7,5
N6	(20)	98	,	100	:	1	;;74LS541	A	8,6
N7	(20)	98	,	112	:	1	;;74ALS574	A	9,6
O1	(20)	98	,	38	:	1	;;16L8-S20	B	3,6
O2	(20)	98	,	14	:	1	;;16L8-S10	B	1,6
O3	(16)	50	,	38	:	1	;;74LS138	B	3,8
O4	(16)	74	,	100	:	1	;;74LS138	A	8,7
O5	(16)	74	,	112	:	1	;;74LS138	A	9,7
P1	(20)	122	,	4	:	1	;;74ALS574	C	1,5
P2	(20)	98	,	4	:	1	;;74LS641-1	C	1,6
P3	(16)	118	,	26	:	1	;;74LS123	C	3,5
P4	(14)	146	,	110	:	1	;;74LS74	B	9,4
P5	(14)	118	,	14	:	1	;;74LS14	C	2,5
P6	(14)	146	,	4	:	1	;;7438	C	1,4
Q1	(16)	194	,	24	:	1	;;74LS123	C	3,2
Q2	(14)	218	,	24	:	1	;;74LS00	C	3,1
Q3	(16)	218	,	34	:	1	;;STRAP	C	4,1
Q4	(14)	218	,	14	:	1	;;74LS74	C	2,1
Q5	(14)	194	,	14	:	1	;;74LS74	C	2,2
Q6	(14)	194	,	4	:	1	;;74LS74	C	1,2
Q7	(14)	218	,	4	:	1	;;74LS74	C	1,1
Q8	(20)	170	,	110	:	1	;;74LS642	B	9,3
Q9	(20)	170	,	4	:	1	;;74LS642	C	1,3
Q10	(14)	146	,	74	:	1	;;74LS00	B	6,4
Q11	(14)	194	,	110	:	1	;;74LS03	B	9,2

R1	(28 )	4 ,	64 : 1	;;8251	A	5,10
R2	(14 )	4 ,	110 : 1	;;75188	B	9,10
R3	(14 )	4 ,	98 : 1	;;75189	B	8,10
R4	(28 )	4 ,	80 : 1	;;8251	A	7,10
R5	(14 )	26 ,	110 : 1	;;75188	B	9,9
R6	(14 )	26 ,	98 : 1	;;75189	B	8,9
R7	(28 )	4 ,	96 : 1	;;8251	A	8,10
R8	(14 )	50 ,	110 : 1	;;75188	B	9,8
R9	(14 )	50 ,	98 : 1	;;75189	B	8,8
R10	(28 )	4 ,	112 : 1	;;8251	A	9,10
R11	(28 )	4 ,	4 : 1	;;8251	B	1,10
R12	(14 )	26 ,	86 : 1	;;75188	B	7,9
R13	(14 )	26 ,	74 : 1	;;75189	B	6,9
R14	(28 )	4 ,	20 : 1	;;8251	B	2,10
R15	(14 )	4 ,	86 : 1	;;75188	B	7,10
R16	(14 )	4 ,	74 : 1	;;75189	B	6,10
R17	(28 )	4 ,	36 : 1	;;8251	B	3,10
R18	(28 )	4 ,	52 : 1	;;8251	B	4,10
S1	(16 )	50 ,	50 : 1	;;74LS157	B	4,8
S2	(14 )	74 ,	26 : 1	;;74LS74	B	2,7
S3	(14 )	74 ,	110 : 1	;;75188	B	9,7
S4	(14 )	74 ,	98 : 1	;;75189	B	8,7
S5	(16 )	74 ,	86 : 1	;;26LS31	B	7,7
S6	(16 )	74 ,	74 : 1	;;26LS32	B	6,7
S7	(14 )	98 ,	50 : 1	;;74LS10	B	4,6
S8	(14 )	74 ,	14 : 1	;;74LS05	B	1,7
S9	(14 )	74 ,	50 : 1	;;74LS14	B	4,7
T1	(16 )	98 ,	62 : 1	;;74LS157	B	5,6
T2	(14 )	74 ,	38 : 1	;;74LS74	B	3,7
T3	(14 )	98 ,	98 : 1	;;75189	B	8,6
T4	(14 )	98 ,	110 : 1	;;75188	B	9,6
T5	(16 )	98 ,	74 : 1	;;26LS32	B	6,6
T6	(16 )	98 ,	86 : 1	;;26LS31	B	7,6
T7	(14 )	74 ,	62 : 1	;;74LS10	B	5,7
XT	(M2 )	204 ,	104 : 0	;;20 M HZ	B	8,2

\*RESISTORS

NAME	DESC.		TYPE	POSITION
RB1 (M4 )	186 ,	106 : 3	;;1K	B
RB2 (M4 )	146 ,	60 : 1	;;680	A
RB3 (M4 )	98 ,	74 : 1	;;1K	A
RB4 (M4 )	146 ,	62 : 1	;;1K	A
RC1 (M4 )	178 ,	62 : 3	;;1K	A
RC2 (M4 )	194 ,	72 : 1	;;33	B
RC3 (M4 )	194 ,	70 : 1	;;33	B
RC4 (M4 )	194 ,	84 : 1	;;33	B
RC5 (M4 )	220 ,	100 : 2	;;33	A
RC6 (M4 )	222 ,	100 : 2	;;33	A
RC7 (M4 )	224 ,	100 : 2	;;33	A
RC8 (M4 )	226 ,	100 : 2	;;33	A
RC9 (M4 )	228 ,	100 : 2	;;33	A
RC10(M4 )	230 ,	100 : 2	;;33	A
RC11(M4 )	232 ,	100 : 2	;;33	A
RC12(M4 )	234 ,	100 : 2	;;33	A
RE1 (M4 )	190 ,	120 : 3	;;1K	A
RG1 (M4 )	54 ,	28 : 1	;;1K	B
RG2 (M4 )	208 ,	106 : 0	;;270	B
RG3 (M4 )	210 ,	106 : 0	;;270	B
RH1 (M4 )	122 ,	36 : 1	;;470	B
RH2 (M4 )	132 ,	12 : 3	;;470	B
RH3 (M4 )	132 ,	28 : 1	;;4K7	B
RH4 (M4 )	122 ,	46 : 1	;;4K7	B
RH5 (M4 )	122 ,	48 : 1	;;1K	B
RH6 (M4 )	122 ,	28 : 1	;;4K7	B
RH7 (M4 )	122 ,	58 : 1	;;1K	B
RH8 (M4 )	146 ,	118 : 1	;;1K	B
RI1 (M4 )	146 ,	72 : 1	;;1K	A
RI2 (M4 )	58 ,	14 : 3	;;1K	A
RI3 (M4 )	164 ,	48 : 3	;;1K	A
RI4 (M4 )	146 ,	48 : 1	;;1K	A
RI5 (M4 )	106 ,	50 : 3	;;1K	A
RJ1 (M4 )	4 ,	16 : 1	;;150	C
RJ2 (M4 )	4 ,	14 : 1	;;150	C
RJ3 (M4 )	142 ,	42 : 1	;;4K7	C
RJ4 (M4 )	142 ,	40 : 1	;;4K7	C

RJ5 (M4 )	142 ,	38 : 1	;;4K7	C
RL1 (M4 )	170 ,	94 : 1	;;1K	B
RM1 (M4 )	74 ,	38 : 1	;;1K	A
RP1 (M4 )	118 ,	24 : 1	;;2K2	C
RP2 (M4 )	118 ,	34 : 1	;;3K3	C
RQ1 (M4 )	156 ,	118 : 1	;;1K	B
RQ2 (M4 )	194 ,	32 : 1	;;22K	C
RQ3 (M4 )	194 ,	22 : 1	;;22K	C
RQ4 (M4 )	218 ,	32 : 1	;;1K	C
RQ5 (M4 )	218 ,	22 : 1	;;1K	C
RQ6 (M4 )	214 ,	40 : 0	;;1K	C
RQ7 (M4 )	214 ,	30 : 0	;;1K	C
RQ8 (M4 )	124 ,	36 : 2	;;150	C
RQ9 (M4 )	128 ,	36 : 2	;;150	C
RQ10(M4 )	132 ,	36 : 2	;;150	C
RQ11(M4 )	140 ,	36 : 2	;;150	C
RQ12(M4 )	136 ,	36 : 2	;;150	C
RS1 (M4 )	60 ,	10 : 1	;;3K9	C
RS2 (M4 )	60 ,	12 : 1	;;270	C
RS3 (M4 )	60 ,	2 : 1	;;3K9	C
RS4 (M4 )	60 ,	4 : 1	;;3K9	C
RS5 (M4 )	60 ,	6 : 1	;;270	C
RS6 (M4 )	60 ,	8 : 1	;;3K9	C
RS7 (M4 )	72 ,	4 : 1	;;3K9	C
RS8 (M4 )	72 ,	6 : 1	;;270	C
RS9 (M4 )	72 ,	8 : 1	;;3K9	C
RT1 (M4 )	74 ,	106 : 1	;;4K7	B
RT2 (M4 )	84 ,	4 : 1	;;3K9	C
RT3 (M4 )	84 ,	6 : 1	;;270	C
RT4 (M4 )	84 ,	8 : 1	;;3K9	C
RT5 (M4 )	84 ,	10 : 1	;;3K9	C
RT6 (M4 )	84 ,	12 : 1	;;270	C
RT7 (M4 )	84 ,	2 : 1	;;3K9	C
RT8 (M4 )	72 ,	10 : 1	;;3K9	C
RT9 (M4 )	72 ,	12 : 1	;;270	C
RT10(M4 )	72 ,	2 : 1	;;3K9	C

\*SIL

NAME	DESC.		TYPE	POSITION
------	-------	--	------	----------

SA1 (SI10)	2 ,	36 : 1	;;1KSIL	A
SC1 (SI10)	218 ,	96 : 1	;;1KSIL	A
SC2 (SI10)	194 ,	24 : 1	;;3K3SIL	B
SE1 (SI10)	122 ,	36 : 1	;;1KSIL	A
SE2 (SI10)	194 ,	86 : 1	;;1KSIL	A
SE3 (SI10)	218 ,	86 : 1	;;1KSIL	A
SN1 (SI10)	74 ,	50 : 1	;;1KSIL	A
SN2 (SI10)	122 ,	98 : 1	;;3K3SIL	A
SN3 (SI10)	98 ,	24 : 1	;;3K3SIL	B
SP1 (SI10)	98 ,	12 : 1	;;1KSIL	C
SR1 (SI10)	50 ,	78 : 1	;;4K7SIL	B
SR2 (SI10)	50 ,	76 : 1	;;1NFSIL	B
SR3 (SI10)	50 ,	74 : 1	;;SIL	B
SR4 (SI10)	50 ,	86 : 1	;;4K7SIL	B
SR5 (SI10)	50 ,	84 : 1	;;1NFSIL	B
SR6 (SI10)	50 ,	82 : 1	;;SIL	B
SR7 (SI10)	50 ,	94 : 1	;;4K7SIL	B
SR8 (SI10)	50 ,	92 : 1	;;1NFSIL	B
SR9 (SI10)	50 ,	90 : 1	;;SIL	B

## \*CONDENSATOR

NAME	DESC.		TYPE	POSITION
C12A(M1 )	22 ,	88 : 0	;;100nf	B
C12B(M1 )	22 ,	92 : 0	;;100nf	B
C12C(M1 )	22 ,	112 : 0	;;100nf	B
C12D(M1 )	22 ,	116 : 0	;;100nf	B
C12E(M1 )	68 ,	116 : 0	;;100nf	B
C12F(M1 )	68 ,	112 : 0	;;100nf	B
CG1 (M2 )	212 ,	104 : 0	;;270pf	B
CT1 (M2 )	88 ,	106 : 3	;;1nf	B
CJ1 (M2 )	156 ,	38 : 3	;;1nf	C
CJ2 (M2 )	156 ,	40 : 3	;;1nf	C
CJ3 (M2 )	156 ,	42 : 3	;;1nf	C
CP1 (M2 )	118 ,	22 : 1	;;1nf	C
CP2 (M2 )	128 ,	34 : 1	;;1nf	C
CQ1 (M2 )	208 ,	32 : 3	;;100nf	C
CQ2 (M2 )	208 ,	22 : 3	;;100nf	C



## \*JUMPER

NAME	DESC.		TYPE	POSITION
SB1 (SIL3)	170 ,	106 : 1	;;JUMPER	B
SG1 (SIL4)	222 ,	94 : 1	;;JUMPER	B
SG2 (SIL4)	222 ,	96 : 1	;;JUMPER	B
SP2 (SIL3)	136 ,	32 : 1	;;JUMPER	C
SP3 (SIL3)	134 ,	24 : 0	;;JUMPER	C
SP4 (SIL3)	134 ,	18 : 0	;;JUMPER	C
SS1 (SIL3)	12 ,	82 : 1	;;JUMPER	B
SS2 (SIL3)	4 ,	82 : 1	;;JUMPER	B
SS3 (SIL3)	82 ,	82 : 1	;;JUMPER	B
SS4 (SIL3)	74 ,	82 : 1	;;JUMPER	B
ST1 (SIL3)	98 ,	120 : 1	;;JUMPER	B
ST2 (SIL3)	106 ,	120 : 1	;;JUMPER	B
ST3 (SIL3)	106 ,	82 : 1	;;JUMPER	B
ST4 (SIL3)	98 ,	82 : 1	;;JUMPER	B

## \*CONNECTORS

NAME	DESC.		TYPE	POSITION
3M20(3M20)	172 ,	38 : 1	;;CONNECTOR	C
3M50(3M50)	50 ,	16 : 1	;;CONNECTOR	C
3M60(3M60)	50 ,	38 : 1	;;CONNECTOR	C

## \*DIODER

NAME	DESC.		TYPE	POSITION
DJ1 (M4 )	4 ,	6 : 1	;;1N4007	C
DJ2 (M4 )	4 ,	10 : 1	;;1N4007	C
DJ3 (M4 )	190 ,	34 : 0	;;1N4148	C
DJ4 (M4 )	180 ,	34 : 1	;;1N4148	C
DJ5 (M4 )	170 ,	34 : 1	;;1N4148	C
DQ1 (P1 )	124 ,	46 : 1	;;LED	C
DQ2 (P1 )	128 ,	46 : 1	;;LED	C
DQ3 (P1 )	132 ,	46 : 1	;;LED	C

DQ4 (P1 )	136 ,	46 : 1	;;LED	C
DQ5 (P1 )	140 ,	46 : 1	;;LED	C

PAL16L8  
 SIOCPAL-1.0 (SIOC 0300 02)  
 ADRESSEDEKODNING  
 DDE-821028  
 A15 A14 A13 A12 A09 A08 /IORW /IOM /SOD GND  
 EB /CSI /ED /BR /DI /CE /M3 /M2 /M1 VCC

IF (VCC) CSI = A15 + A14 + A13 + A12 + /IORW  
 ;;Clear interrupt

IF (VCC) BR = A15 \* A14 \* EB \* /IOM \* /ED  
 + A15 \* A14 \* /A13 \* A12 \* EB \* /IOM  
 + A15 \* A14 \* A13 \* /A12 \* EB \* /IOM  
 + A15 \* A14 \* /A13 \* /A12 \* EB \* /IOM  
 ;;Request til bus

IF (VCC) DI = A15 \* A14 \* A13 \* A12 \* /IOM \* ED  
 + /A15 \* /A14 \* /A13 \* /A12 \* /IOM \* /SOD  
 + A15 \* A14 \* /A13 \* A12 \* /IOM \* EB  
 + A15 \* A14 \* A13 \* /A12 \* /IOM \* EB  
 + A15 \* A14 \* /A13 \* /A12 \* /IOM \* EB

IF (VCC) CE = /A15 \* /A14 \* /A13 \* /A12 \* /IOM \* /SOD;CE BOOT  
 + A15 \* A14 \* A13 \* A12 \* /IOM \* ED ;CE DEBUG

IF (VCC) M3 = A15 \* A14 \* A09 \* /A08 \* IOM \* IORW;IO ADR  
 + A15 \* A14 \* /A13 \* /A12 \* /IOM \* EB ;C2 HEX  
 + A15 \* A14 \* /A13 \* A12 \* /IOM \* EB  
 + A15 \* A14 \* A13 \* /A12 \* /IOM \* EB  
 + A15 \* A14 \* /IOM \* /ED \* EB

IF (VCC) M2 = A15 \* A14 \* /A09 \* A08 \* IOM \* IORW;IO ADR  
 + A15 \* A14 \* /A13 \* /A12 \* /IOM \* EB ;C1 HEX  
 + A15 \* A14 \* /A13 \* A12 \* /IOM \* EB  
 + A15 \* A14 \* A13 \* /A12 \* /IOM \* EB  
 + A15 \* A14 \* /IOM \* /ED \* EB

IF (VCC) M1 = A15 \* A14 \* /A09 \* /A08 \* IOM \* IORW;IO ADR

```

+ A15 * A14 * /A13 * /A12 * /IOM * EB ;CO HEX
+ A15 * A14 * /A13 * A12 * /IOM * EB
+ A15 * A14 * A13 * /A12 * /IOM * EB
+ A15 * A14 * /IOM * /ED * EB

```

## FUNCTION TABLE:

```

A15 A14 A13 A12 A09 A08 /IORW /IOM /SOD EB /ED
CSI BR DI CE M3 M2 M1

```

```

-----
LLLLXX LXX XX LXXX XXX CSINT
HLLLXX LXX XX HXXX XXX
LHLLXX LXX XX HXXX XXX
LLHLXX LXX XX HXXX XXX
LLLHXX LXX XX HXXX XXX
LLLLXX HXX XX HXXX XXX
HHHXX XHX HH XHXX XXX BR
LHHXX XHX HH XLXX XXX
HLHXX XHX HH XLXX XXX
HHHXX XLX HH XLXX XXX
HHHXX XHX LH XLXX XXX
HHHXX XHX HL XLXX XXX
HHLXX XHX HL XHXX XXX
HHLXX XHX HL XHXX XXX
HHLXX XHX HL XHXX XXX
HHHXX XHX HL XLXX XXX
LHLXX XHX HL XLXX XXX
HLLXX XHX HL XLXX XXX
HLLXX XLX HL XLXX XXX
HLLXX XHX LL XLXX XXX
LHLXX XHX HL XLXX XXX
HLLXX XHX HL XLXX XXX
HHLXX XLX HL XLXX XXX
HHLXX XHX LL XLXX XXX
LHLXX XHX HL XLXX XXX
HLHLXX XHX HL XLXX XXX
HHHLXX XLX HL XLXX XXX
HHHLXX XHX LL XLXX XXX
HHLXX XHL HL XXHX XXX DI
HHLXX XHL HL XXHX XXX
HHHLXX XHL HL XXHX XXX

```

HHHHXX	XHL	HL	XXHX	XXX
LHLLXX	XHL	HL	XXLX	XXX
HLLLXX	XHL	HL	XXLX	XXX
HHLLXX	XL	HL	XXLX	XXX
HHLLXX	XHL	LL	XXLX	XXX
LHLHXX	XHL	HL	XXLX	XXX
HLLHXX	XHL	HL	XXLX	XXX
HHLHXX	XL	HL	XXLX	XXX
HHLHXX	XHL	LL	XXLX	XXX
LHHLXX	XHL	HL	XXLX	XXX
HLHLXX	XHL	HL	XXLX	XXX
HHHLXX	XL	HL	XXLX	XXX
LHHLXX	XHL	LL	XXLX	XXX
LHHHXX	XHL	HL	XXLX	XXX
HLHHXX	XHL	HL	XXLX	XXX
HHHHXX	XL	HL	XXLX	XXX
HHHHXX	XHL	HH	XXLX	XXX
LLLLXX	XHH	LH	XXHX	XXX
HLLLXX	XHH	LH	XXLX	XXX
LHLLXX	XHH	LH	XXLX	XXX
LLHLXX	XHH	LH	XXLX	XXX
LLLHXX	XHH	LH	XXLX	XXX
LLLLXX	XLH	LH	XXLX	XXX
LLLLXX	XHL	LH	XXLX	XXX
HHHHXX	XHL	XL	XXXH	XXX CE
HLLLXX	XHH	XH	XXXL	XXX
LHLLXX	XHH	XH	XXXL	XXX
LLHLXX	XHH	XH	XXXL	XXX
LLLHXX	XHH	XH	XXXL	XXX
LLLLXX	XLH	XH	XXXL	XXX
LLLLXX	XHL	XH	XXXL	XXX
HHHHXX	XHL	XL	XXXH	XXX
LHHHXX	XHL	XL	XXXL	XXX
HLHHXX	XHL	XL	XXXL	XXX
HHLHXX	XHL	XL	XXXL	XXX
HHHLXX	XHL	XL	XXXL	XXX
HHHHXX	XL	XL	XXXL	XXX
HHHHXX	XHL	XH	XXXL	XXX
HHLLXX	HXX	HL	XXXX	HHH M3 M2 M1
HHLHXX	HXX	HL	XXXX	HHH

HHHLXX	HXX	HL	XXXX	HHH
HHHHXX	HXX	HL	XXXX	LLL
LHLLEX	HXX	HL	XXXX	LLL
HLLEX	HXX	HL	XXXX	LLL
HHLLEX	HLX	HL	XXXX	LLL
HHLLEX	HXX	LL	XXXX	LLL
HXXLL	LLX	LL	XXXX	LLH
HXXLH	LLX	LL	XXXX	LHL
HXXHL	LLX	LL	XXXX	HLL
HXXHH	LLX	LL	XXXX	LLL
LHXXLL	LLX	LL	XXXX	LLL
HLXXLL	LLX	LL	XXXX	LLL
HXXLL	HLX	LL	XXXX	LLL
HXXLL	LHX	LL	XXXX	LLL
LHXXLH	LLX	LL	XXXX	LLL
HLXXLH	LLX	LL	XXXX	LLL
HXXLH	HLX	LL	XXXX	LLL
HXXLH	LHX	LL	XXXX	LLL
LHXXHL	LLX	LL	XXXX	LLL
HLXXHL	LLX	LL	XXXX	LLL
HXXHL	HLX	LL	XXXX	LLL
HXXHL	LHX	LL	XXXX	LLL
HHHHH	XHX	HH	XXXX	HHH
LHHHH	XHX	HH	XXXX	LLL
HLHHH	XHX	HH	XXXX	LLL
HHHHH	XLX	HH	XXXX	LLL
HHHHH	XHX	LH	XXXX	LLL
HHHHH	XHX	HL	XXXX	LLL

---

DESCRIPTION

PAL16L8

SIOCPAL-2.0 (0300 01)

IO ADRESSEDEKODNING

DDE-830311BS

A7	A6	A5	A4	A3	NC	NC	/IOM	/IOW	GND
/IOR	/CSSU	/CSL	/CSFC	/OEP	/INTA	/CSC	/CSO	/CSU	VCC

IF ( VCC ) CSSU = /A7 \* A6 \* A5 \* /A4 \* /A3 \* IOM  
;PORT 60-67

IF ( VCC ) CSL = /A7 \* A6 \* A5 \* /A4 \* /A3 \* IOM \* /INTA  
+ A7 \* /A6 \* /A5 \* /A4 \* IOM \* /INTA  
+/A7 \* A6 \* /A5 \* A4 \* IOM \* /INTA  
; ACCESS TIL LSI IO-DATABUS

IF ( VCC ) CSFC = /A7 \* A6 \* A5 \* A4 \* /A3 \* IOR  
;PORT 70-77

IF ( VCC ) OEP = /A7 \* A6 \* A5 \* /A4 \* A3 \* IOR  
;PORT 68-6F

IF ( VCC ) CSC = A7 \* /A6 \* /A5 \* /A4 \* IOM  
;PORT 80-8F

IF ( VCC ) CSO = A7 \* /A6 \* A5 \* A4 \* A3 \* IOW  
;PORT B8-BF

IF ( VCC ) CSU = /A7 \* A6 \* /A5 \* A4 \* IOM  
;PORT 50-5F

FUNCTION TABLE

A7	A6	A5	A4	A3	/IOM	/IOW	/IOR	/INTA				
					CSSU	CSL	CSFC	OEP	CSC	CSO	CSU	

---

LHLL	LXXH	XHX	XX	XX	CSL							
HHLL	LXXH	XLX	XX	XX								
LLHL	LXXH	XLX	XX	XX								
LHHL	LXXH	XLX	XX	XX								
LHHL	LXXH	XLX	XX	XX								
LHLL	HXXH	XLX	XX	XX								
LHLL	LXXL	XLX	XX	XX								

HLLLH LXXH	XHX XX XX	
LLLLH LXXH	XLX XX XX	
HLLLH LXXH	XLX XX XX	
HLHLH LXXH	XLX XX XX	
HLLLH HXXH	XLX XX XX	
HLLHH HXXH	XLX XX XX	
HLLLH LXXL	XLX XX XX	
LHLHH LXXH	XHX XX XX	
HHLHH LXXH	XLX XX XX	
LLLHH LXXH	XLX XX XX	
LHHHH LXXH	XLX XX XX	
LHLLH LXXH	XLX XX XX	
LHLHH HXXH	XLX XX XX	
LHLHH LXXL	XLX XX XX	
LHHLL LXXH	HHL LL LL	CSSU
HHLLL LXXX	LXX XX XX	
LLHLL LXXX	LXX XX XX	
LHLLL LXXX	LXX XX XX	
LHHHL LXXX	LXX XX XX	
LHHLH LXXX	LXX XX XX	24
LHHLL HXXX	LXX XX XX	
LHHHL XXLH	LLH LL LL	CSFC
HHHHL XXLX	XXL XX XX	
LLHHL XXLX	XXL XX XX	
LHLHL XXLX	XXL XX XX	
LHHLL XXLX	XXL XX XX	
LHHHH XXLX	XXL XX XX	
LHHHL XXHX	XXL XX XX	
LHHLH XXLH	LLL HL LL	OEP
HHHLH XXLX	XXX LX XX	
LLHLH XXLX	XXX LX XX	
LHLLH XXLX	XXX LX XX	
LHHHH XXLX	XXX LX XX	
LHHLL XXLX	XXX LX XX	
LHHLH XXHX	XXX LX XX	
HLLLX LXXH	LHL LH LL	CSC
LLLLX LXXX	XXX XL XX	
HLLLX LXXX	XXX XL XX	
HLHLX LXXX	XXX XL XX	
HLLHX LXXX	XXX XL XX	



HLLLX HXXX	XXX XL XX
HLHHH XLXH	LLL LL HL CSO
LLHHH XLXX	XXX XX LX
HHHHH XLXX	XXX XX LX
HLLHH XLXX	XXX XX LX
HLHLH XLXX	XXX XX LX
HLHHL XLXX	XXX XX LX
HLHHH XHXX	XXX XX LX
LHLHX LXXH	LHL LL LH CSU
HHLHX LXXX	XXX XX XL
LLLHX LXXX	XXX XX XL
LHHHX LXXX	XXX XX XL
LHLLX LXXX	XXX XX XL
LHLHX HXXX	XXX XX XL

---

DESCRIPTION

PAL16L8

SIOCPAL-3.1 (SIOC 0300 H7)

8085 STYRESIGNALER

DDE-850807bs

/DADIS /DIRAM /CONI S1 /IOM /ALE S0 RW1 /RW2 GND  
/INTA /DIR5 /ADG /DBG5 /DBG3 /RW /MEMA /IOW /IOR VCC

IF ( /DADIS ) IOR = IOM \* RW2 \* /ALE \* S1

IF ( /DADIS ) IOW = IOM \* RW2 \* /ALE \* /S1

IF ( VCC ) RW = ALE + S1 + DADIS + IOM

IF ( VCC ) MEMA = /IOM \* RW1 \* /ALE \* /DIRAM \* /DADIS  
+ IOM \* S1\*/S0\* RW1 \* /ALE \* /DIRAM \* /DADIS  
+ IOM \*/S1\* S0\* RW1 \* /ALE \* /DIRAM \* /DADIS  
+ IOM \* S1\* S0\* RW1 \* /ALE \* /DIRAM \* /DADIS  
\* INTA

IF ( VCC ) DBG3 = /IOM \* RW2 \* /ALE \* /CONI \* /DADIS

IF ( VCC ) DBG5 = IOM \* RW2 \* /ALE \* /DADIS ;IORW  
+ INTA ;INTA

IF ( VCC ) ADG = /IOM \* /ALE \* /CONI \* /DADIS

IF ( VCC ) DIR5 = IOM \* RW2 \* /ALE \* S1 ;IOR  
+ INTA ;INTA

FUNCTION TABLE

/DADIS /DIRAM /CONI S1 /IOM /ALE S0 /RW2 RW1 /INTA  
DIR5 ADG DBG5 DBG3 RW MEMA IOW IOR

-----  
HHX H LH HXH L XX XX XH XX  
LHX H LH HXH L XX XX XL XX  
HLX H LH HXH L XX XX XL XX  
HHX H LL HXH L XX XX XL XX  
HHX H LH HXL L XX XX XL XX  
HHX H LH HXH H XX XX XL XX  
HHX L LH LXH L XX XX XL XX  
HHX L LH HXH H XX XX XH XX  
LHX L LH HXH H XX XX XL XX  
HLX L LH HXH H XX XX XL XX  
HHX L LL HXH H XX XX XL XX



PAL16L8  
 SIOCPAL-3.0 (SIOC 0300 H7)  
 8085 STYRESIGNALER  
 DDE-821029  
 /DADIS /DIRAM /CONI S1 /IOM /ALE /RW3 RW1 /RW2 GND  
 /INTA /DIR5 /ADG /DBG5 /DBG3 /RW /MEMA /IOW /IOR VCC

IF ( /DADIS ) IOR = IOM \* RW2 \* /ALE \* S1  
 IF ( /DADIS ) IOW = IOM \* RW2 \* /ALE \* /S1  
 IF ( VCC ) RW = ALE + S1 + DADIS  
 IF ( VCC ) MEMA = /IOM \* RW1 \* /ALE \* /DIRAM \* /DADIS  
 IF ( VCC ) DBG3 = /IOM \* RW2 \* /ALE \* /CONI \* /DADIS  
 IF ( VCC ) DBG5 = IOM \* RW3 \* /ALE \* /DADIS ;IORW  
                   + INTA ;INTA  
 IF ( VCC ) ADG = /IOM \* /ALE \* /CONI \* /DADIS  
 IF ( VCC ) DIR5 = IOM \* RW2 \* /ALE \* S1 ;IOR  
                   + INTA ;INTA

FUNCTION TABLE:

	/DADIS	/DIRAM	/CONI	S1	/IOM	/ALE	/RW3	/RW2	RW1	/INTA
			DIR5	ADG	DBG5	DBG3	RW	MEMA	IOW	IOR
XXX	H	LH	XLX	H	HX	XX	XX	XX	DIR5	
XXX	L	LH	XLX	H	LX	XX	XX	XX		
XXX	H	HH	XLX	H	LX	XX	XX	XX		
XXX	H	LL	XLX	H	LX	XX	XX	XX		
XXX	H	LH	XHX	H	LX	XX	XX	XX		
XXX	X	XX	XHX	L	HX	XX	XX	XX		
XXX	X	XX	XHX	H	LX	XX	XX	XX		
HXH	X	HH	XXX	X	XH	XX	XX	XX	ADG	
LXH	X	HH	XXX	X	XL	XX	XX	XX		
HXL	X	HH	XXX	X	XL	XX	XX	XX		

HXH X LH XXX X XL XX XX XX  
 HXH X HL XXX X XL XX XX XX  
 HXX X LH LXX H XX HX XX XX DBG5  
 LXX X LH LXX H XX LX XX XX  
 HXX X HH LXX H XX LX XX XX  
 HXX X LL LXX H XX LX XX XX  
 HXX X LH HXX H XX LX XX XX  
 LXX X XX XXX L XX HX XX XX  
 LXX X XX XXX H XX LX XX XX  
 HXH X HH XLX X XX XH XX XX DBG3  
 LXH X HH XLX X XX XL XX XX  
 HXL X HH XLX X XX XL XX XX  
 HXH X LH XLX X XX XL XX XX  
 HXH X HL XLX X XX XL XX XX  
 HXH X HH XHX X XX XL XX XX  
 HHX X HH XXH X XX XX XH XX MEMA  
 LHX X HH XXH X XX XX XL XX  
 HLX X HH XXH X XX XX XL XX  
 HHX X LH XXH X XX XX XL XX  
 HHX X HL XXH X XX XX XL XX  
 HHX X HH XXL X XX XX XL XX  
 HXX L XH XXX X XX XX LX XX RW  
 LXX L XH XXX X XX XX HX XX  
 HXX H XH XXX X XX XX HX XX  
 HXX L XL XXX X XX XX HX XX  
 XXX L LH XLX X XX XX XX HL IOW  
 XXX H LH XLX X XX XX XX LX  
 XXX L HH XLX X XX XX XX LX  
 XXX L LL XLX X XX XX XX LX  
 XXX L LH XHX X XX XX XX LX  
 XXX H LH XLX X XX XX XX LH IOR  
 XXX L LH XLX X XX XX XX XL  
 XXX H HH XLX X XX XX XX XL  
 XXX H LL XLX X XX XX XX XL  
 XXX H LH XHX X XX XX XX XL

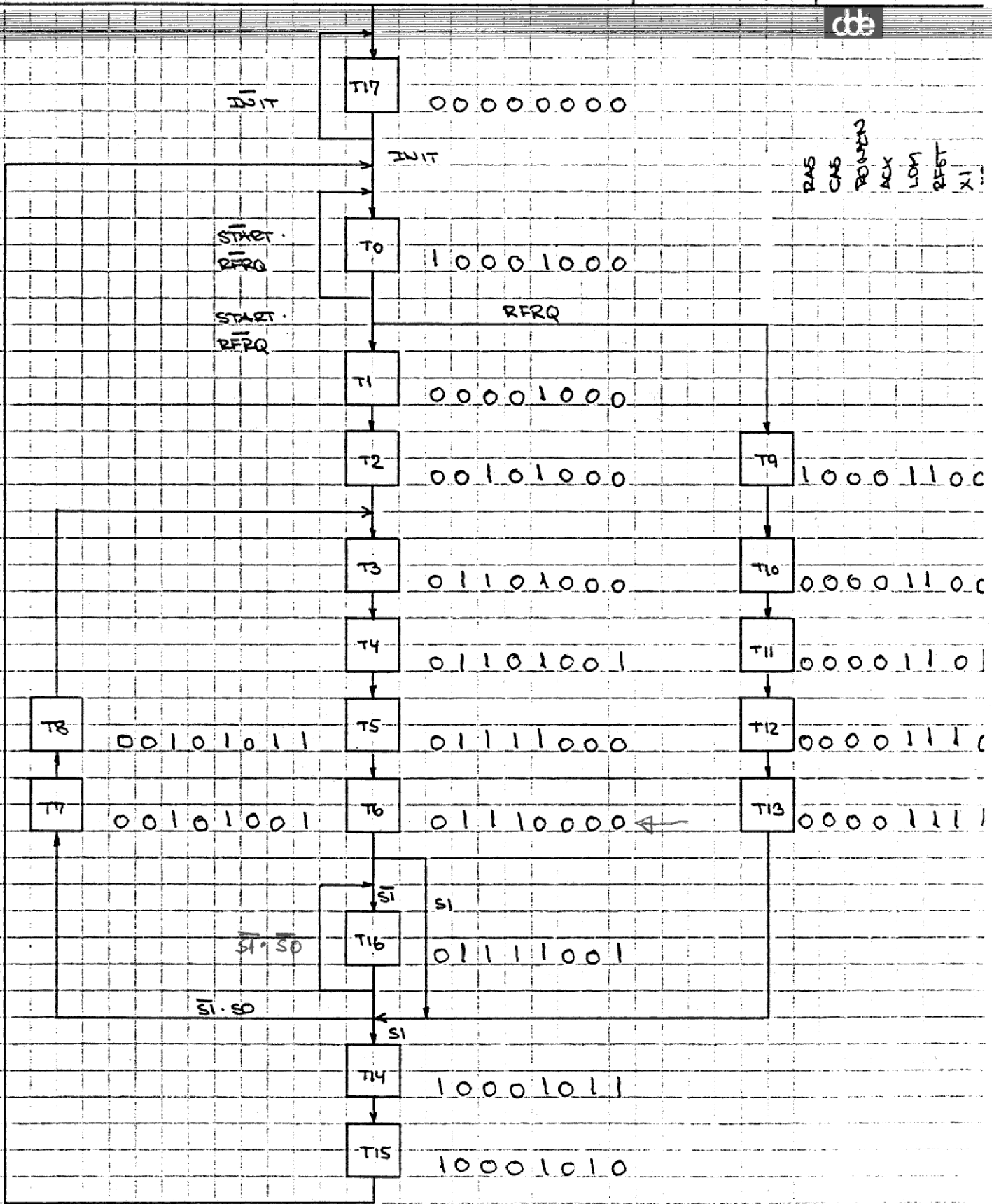
---

DESCRIPTION

SPAL 40, SIOC 0300



RAS  
CAS  
RDYEN  
ACK  
LOM  
RFST  
XI



PAL16R8  
 SIOCPAL-4.0 (O300 K2)  
 LAGERSTYRING  
 DDE-821108BS  
 CP20 /START /RFRQ NC /S0 /INIT /S1 NC NC GND  
 GND /Q0 /Q1 /Q2 /Q3 /Q4 /Q5 /Q6 /Q7 VCC

```

Q7 := S1          * /Q7* Q6* Q5* Q4*/Q3*/Q2*/Q1*/Q0 ;T6
                  + /Q7*/Q6*/Q5*/Q4* Q3* Q2* Q1* Q0 ;T13
                  + Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1 ;T14
                  ;T15
+ /START* /RFRQ*Q7*/Q6*/Q5*/Q4* Q3*/Q2*/Q1*/Q0 ;T0
+ RFRQ * Q7*/Q6*/Q5*/Q4* Q3*/Q2*/Q1*/Q0 ;T0
+ INIT
+ S1          * /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1* Q0 ;T16
                  + /Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0 ;T17

Q6 :=          /Q7*/Q6* Q5*/Q4* Q3*/Q2*/Q1*/Q0*/INIT;T2
+ /Q7* Q6* Q5*/Q4* Q3*/Q2*/Q1*/Q0*/INIT;T3
+ /Q7* Q6* Q5*/Q4* Q3*/Q2*/Q1* Q0*/INIT;T4
+ /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1*/Q0*/INIT;T5
+ /S1          * /Q7* Q6* Q5* Q4*/Q3*/Q2*/Q1*/Q0*/INIT;T6
+ /Q7*/Q6* Q5*/Q4* Q3*/Q2* Q1* Q0*/INIT;T8
+ /S1* /S0    * /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1* Q0*/INIT;T16

Q5 :=          /Q7*/Q6*/Q5*/Q4* Q3*/Q2*/Q1*/Q0*/INIT;T1
+ /Q7*/Q6* Q5*/Q4* Q3*/Q2*/Q1*/Q0*/INIT;T2
+ /Q7* Q6* Q5*/Q4* Q3*/Q2*/Q1*/Q0*/INIT;T3
+ /Q7* Q6* Q5*/Q4* Q3*/Q2*/Q1* Q0*/INIT;T4
+ /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1*/Q0*/INIT;T5
+ /S1          * /Q7* Q6* Q5* Q4*/Q3*/Q2*/Q1*/Q0*/INIT;T6
+ /Q7*/Q6* Q5*/Q4* Q3*/Q2* Q0*/INIT;T7
                  ;T8
+ /S1          * /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1* Q0*/INIT;T16

Q4 :=          /Q7* Q6* Q5*/Q4* Q3*/Q2*/Q1* Q0*/INIT;T4
+ /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1*/Q0*/INIT;T5
+ /S1          * /Q7* Q6* Q5* Q4*/Q3*/Q2*/Q1*/Q0*/INIT;T6
+ /S1* /S0    * /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1* Q0*/INIT;T16
  
```

```

Q2 :=   RFRQ      *   Q7*/Q6*/Q5*/Q4* Q3*/Q2*/Q1*/Q0*/INIT;T0
          +   Q7*/Q6*/Q5*/Q4* Q3* Q2*/Q1*/Q0*/INIT;T9
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*/Q1*/Q0*/INIT;T10
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*/Q1*  Q0*/INIT;T11
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*  Q1*/Q0*/INIT;T12

Q1 :=           Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1* Q0*/INIT;T14
          +   /Q7*/Q6*  Q5*/Q4* Q3*/Q2*/Q1* Q0*/INIT;T7
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*/Q1* Q0*/INIT;T11
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*  Q1*/Q0*/INIT;T12
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*  Q1* Q0*/INIT;T13
          + S1      *   /Q7* Q6* Q5* Q4*/Q3*/Q2*/Q1*/Q0*/INIT;T6
          + S1      *   /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1* Q0*/INIT;T16

Q0 :=           /Q7* Q6* Q5*/Q4* Q3*/Q2*/Q1*/Q0*/INIT;T3
          +   /Q7* Q6* Q5* Q4*/Q3*/Q2*/Q1*/Q0*/INIT;T6
          +   /Q7*/Q6*  Q5*/Q4* Q3*/Q2*/Q1* Q0*/INIT;T7
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*/Q1*/Q0*/INIT;T10
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*  Q1*/Q0*/INIT;T12
          +   /Q7*/Q6*/Q5*/Q4* Q3* Q2*  Q1* Q0*/INIT;T13
          +   /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1* Q0*/INIT;T16

Q3 :=           /Q7* Q6* Q5* Q4*/Q3*/Q2*/Q1*/Q0      ;T6
          +   /Q7*/Q6*/Q5*/Q4* Q3
          +   /Q7*/Q6*  Q5*/Q4* Q3*/Q2
          +   /Q7* Q6* Q5*/Q4* Q3*/Q2*/Q1
          +   Q7*/Q6*/Q5*/Q4* Q3
          +   INIT
          +   /Q7* Q6* Q5* Q4* Q3*/Q2*/Q1* Q0      ;T16
          +   /Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0      ;T17

```

## FUNCTION TABLE:

```
CP20 /START /RFRQ /S0 /S1 /INIT Q7 Q6 Q5 Q4 Q3 Q2 Q1 Q0
```

```
-----
CXXXXL HLLLHLLL TO
CHHXXH HLLLHLLL TO
CLHXXH LLLLHLLL T1
```

CXXXXH LLHLHLLL T2  
CXXXXH LHHLHLLL T3  
CXXXXH LHHLHLLH T4  
CXXXXH LHHHLLL T5  
CXXXXH LHHHLLL T6  
CXXXLH HLLLHLHH T14  
CXXXXH HLLLHLHL T15  
CXXXXH HLLLHLLL T0  
CXLXXH HLLLHHLL T9  
CXXXXH LLLLHLL T10  
CXXXXH LLLLHHLH T11  
CXXXXH LLLLHHHL T12  
CXXXXH LLLLHHHH T13  
CXXXXH HLLLHLHH T14  
CXXXXH HLLLHLHL T15  
CXXXXH HLLLHLLL T0  
CLHXXH LLLLHLLL T1  
CXXXXH LLHLHLLL T2  
CXXXXH LHHLHLLL T3  
CXXXXH LHHLHLLH T4  
CXXXXH LHHHLLL T5  
CXXXXH LHHHLLL T6  
CXXXXH LHHHLLH T16  
CXXHHH LHHHLLH T16  
CXXLHH LLHLHLLH T7  
CXXXXH LLHLHLHH T8  
CXXXXH LHHLHLLL T3  
CXXXXH LHHLHLLH T4  
CXXXXH LHHHLLL T5  
CXXXXH LHHHLLL T6  
CXXXXH LHHHLLH T16  
CXXXLH HLLLHLHH T14  
CXXXXH HLLLHLHL T15  
CXXXXH HLLLHLLL T0

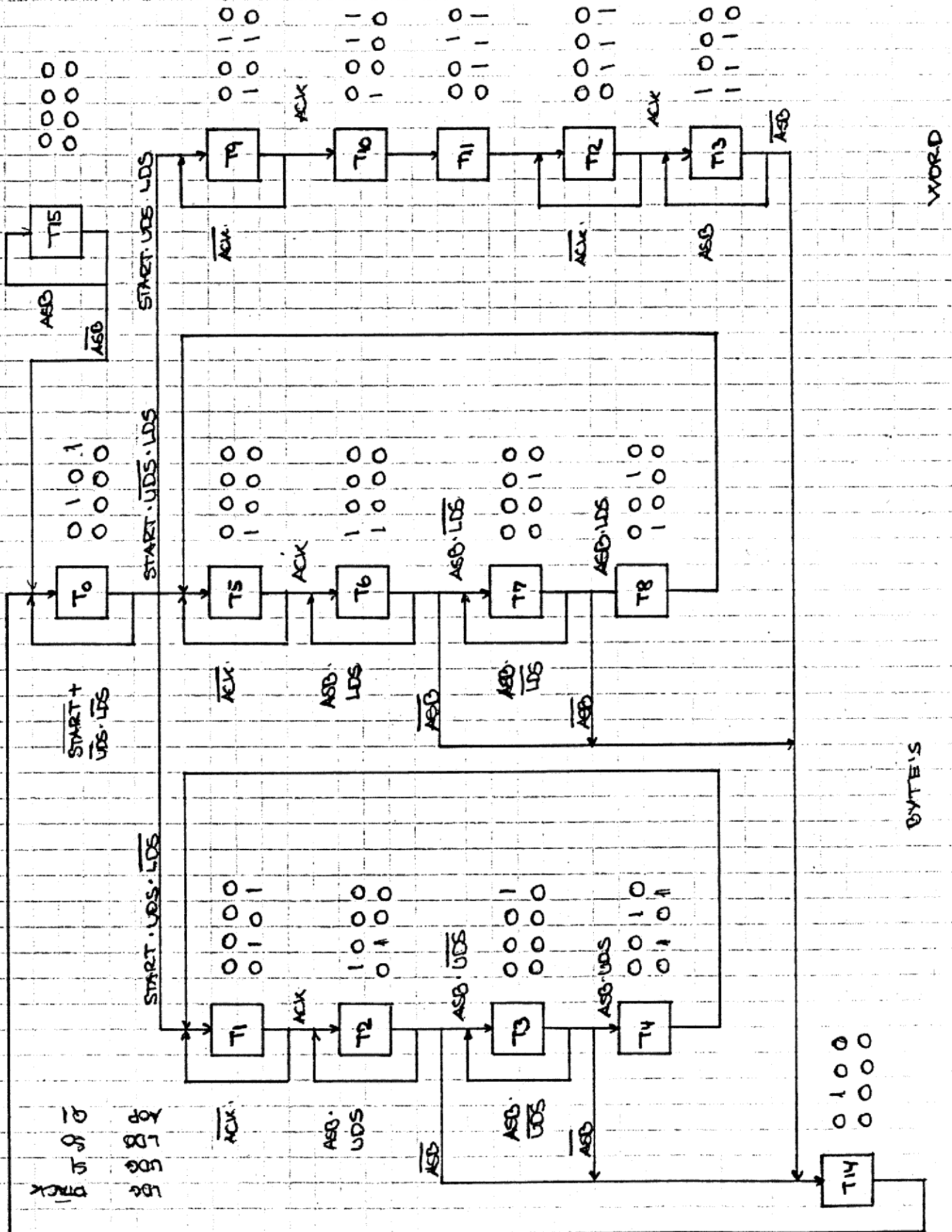
-----  
DESCRIPTION



BYTE, WORD OG RMW I ROSS LAGER.

010 C0300

SPAL 50



PAL16R8

SIOCPAL-5.0 (0300 K3)

STYRING AF BYTE, WORD OG RMW I 8085'S LAGER.

821216-BS

CLK /STB AS /UDS /LDS ACK /CL NC NC GND

GND /A /L /LD /U /Q1 /S0 /S1 /DT VCC

```

DT := ACK *
      /DT * /S1 * /S0 * /Q1 * U * /LD * /L * A * /CL ;1
+ AS * UDS *
      DT * /S1 * /S0 * /Q1 * U * /LD * /L * /A * /CL ;2
+ ACK *
      /DT * /S1 * /S0 * /Q1 * /U * LD * /L * /A * /CL ;5
+ AS * LDS *
      DT * /S1 * /S0 * /Q1 * /U * LD * /L * /A * /CL ;6
+ ACK *
      /DT * /S1 * /S0 * /Q1 * U * /LD * L * A * /CL ;12
+ AS *
      DT * /S1 * /S0 * /Q1 * U * LD * L * /A * /CL ;13
S1 := /AS *
      DT * /S1 * /S0 * /Q1 * U * /LD * /L * /A * /CL ;2
+ /AS *
      /DT * /S1 * /S0 * /U * /LD * /L * /A * /CL ;3
+ /AS * ;15
      DT * /S1 * /S0 * /Q1 * /U * LD * /L * /A * /CL ;6
+ /AS *
      /DT * /S1 * /S0 * /Q1 * /U * /LD * L * /A * /CL ;7
+ /AS *
      DT * /S1 * /S0 * /Q1 * U * LD * L * /A * /CL ;13
+ /DT * S1 * /S0 * /Q1 * /U * /LD * /L * /A * /CL ;14
+ /STB *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /UDS * /LDS *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
S0 := AS * UDS *
      /DT * /S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;3
+ AS * LDS *
      /DT * /S1 * /S0 * /Q1 * /U * /LD * L * /A * /CL ;7
+ STB * LDS * UDS *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /ACK *

```

```

      /DT * /S1 * S0 * /Q1 * /U * LD * L * /A * /CL ;9
+ ACK *
      /DT * /S1 * S0 * /Q1 * /U * LD * L * /A * /CL ;9
+ /DT * /S1 * S0 * Q1 * /U * LD * /L * /A * /CL ;10
Q1 := AS * /UDS *
      DT * /S1 * /S0 * /Q1 * U * /LD * /L * /A * /CL ;2
+ AS * /UDS *
      /DT * /S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;3
+ ACK *
      /DT * /S1 * S0 * /Q1 * /U * LD * L * /A * /CL ;9
+ /DT * S1 * /S0 * /Q1 * /U * /LD * /L * /A * /CL ;14
+ /AS *
      /DT * /S1 * /S0 * /Q1 * /U * /LD * /L * /A * /CL ;15
+ /STB *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /UDS * /LDS *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
LD := STB * /UDS * LDS *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /DT * /S1 * /Q1 * /U * LD * /L * /A * /CL ;5
      ;8
+ AS * LDS *
      DT * /S1 * /S0 * /Q1 * /U * LD * /L * /A * /CL ;6
+ AS * LDS *
      /DT * /S1 * /S0 * /Q1 * /U * /LD * L * /A * /CL ;7
+ ACK *
      /DT * /S1 * /S0 * /Q1 * U * /LD * L * A * /CL ;12
+ AS *
      DT * /S1 * /S0 * /Q1 * U * LD * L * /A * /CL ;13
+ STB * UDS * LDS *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /DT * /S1 * S0 * /Q1 * /U * LD * L * /A * /CL ;9
U := STB * UDS * /LDS *
      /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /DT * /S1 * /S0 * /Q1 * U * /LD * /L * A * /CL ;1
+ /DT * /S1 * S0 * /Q1 * U * /LD * /L * A * /CL ;4
+ AS * UDS *
      DT * /S1 * /S0 * /Q1 * U * /LD * /L * /A * /CL ;2
+ AS * UDS *
      /DT * /S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;3

```

```

+ /DT * /S1          * /Q1 * U * /LD * L * A * /CL ;11
;12
+ AS *
  DT * /S1 * /S0 * /Q1 * U * LD * L * /A * /CL ;13
+ /DT * /S1 * S0 * Q1 * /U * LD * /L * /A * /CL ;10
L := AS * /LDS *
  DT * /S1 * /S0 * /Q1 * /U * LD * /L * /A * /CL ;6
+ AS * /LDS *
  /DT * /S1 * /S0 * /Q1 * /U * /LD * L * /A * /CL ;7
+ STB * UDS * LDS *
  /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /ACK *
  /DT * /S1 * S0 * /Q1 * /U * LD * L * /A * /CL ;9
+ /DT * /S1 * S0 * Q1 * /U * LD * /L * /A * /CL ;10
+ /DT * /S1 * S0 * /Q1 * U * /LD * L * A * /CL ;11
+ /DT * /S1 * /S0 * /Q1 * U * /LD * L * A * /CL ;12
+ AS *
  DT * /S1 * /S0 * /Q1 * U * LD * L * /A * /CL ;13
A := STB * UDS * /LDS *
  /DT * S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;0
+ /ACK *
  /DT * /S1 * /S0 * /Q1 * U * /LD * /L * A * /CL ;1
+ /DT * /S1 * S0 * /Q1 * U * /LD * /L * A * /CL ;4
+ AS * UDS *
  /DT * /S1 * /S0 * Q1 * /U * /LD * /L * /A * /CL ;3
+ /DT * /S1 * S0 * Q1 * /U * LD * /L * /A * /CL ;10
+ /DT * /S1 * S0 * /Q1 * U * /LD * L * A * /CL ;11
+ /ACK *
  /DT * /S1 * /S0 * /Q1 * U * /LD * L * A * /CL ;12

```

## FUNCTION TABLE:

CLK	/STB	AS	/UDS	/LDS	ACK	/CL	DT	S1	S0	Q1	LD	U	L	A
-----														
CXXXXX	L	L	L	L	L	L	L	L	L	L	L	L	L	T15
CXXXXX	L	L	L	L	L	L	L	L	L	L	L	L	L	T15
CXHXHX	H	L	L	L	L	L	L	L	L	L	L	L	L	T15
CXLXXX	H	L	H	L	H	L	L	L	L	L	L	L	L	T0
CHXXXX	H	L	H	L	H	L	L	L	L	L	L	L	L	T0
CLXHHX	H	L	H	L	H	L	L	L	L	L	L	L	L	T0
CLXLHX	H	L	L	L	L	L	L	L	L	H	L	L	L	T1

CXXXXLHLLLLLHHT1  
CXXXXHHHLLLLHLLT2  
CXHLXXHHLLLLHLLT2  
CXHHXXHLLLLHLLLT3  
CXHHXXHLLLLHLLLT3  
CXHLXXHLLHLLHHT4  
CXXXXHLLLLLHHT1  
CXXXXHHHLLLLHLLT2  
CXLXXHLLHLLLLLT14  
CXXXXHLLHLLLT0  
CLXLHXHLLLLLHHT1  
CXXXXHHHLLLLHLLT2  
CXHHXXHLLLLHLLLT3  
CXLXXHLLHLLLLLT14  
CXXXXHLLHLLLT0  
CLXHLXHLLLLHLLT5  
CXXXXLHLLLLHLLT5  
CXXXXHHHLLLLHLLT6  
CXHLXHHLLLLHLLT6  
CXHXHXHLLLLLHHT7  
CXHXHXHLLLLLHHT7  
CXHLXHLLHLLHLLT8  
CXXXXHLLLLHLLT5  
CXXXXHHHLLLLHLLT6  
CXLXXHLLHLLLLLT14  
CXXXXHLLHLLLT0  
CLXHLXHLLLLHLLT5  
CXXXXHHHLLLLHLLT6  
CXHXHXHLLLLLHHT7  
CXLXXHLLHLLLLLT14  
CXXXXHLLHLLLT0  
CLLLXHLLHLLHHT9  
CXXXXLHLLHLLHHT9  
CXXXXHHLLHHHLLT10  
CXXXXHLLHLLHHHT11  
CXXXXHLLLLLLHHHT12  
CXXXXLHLLLLHHHT12  
CXXXXHHHLLHHHHT13  
CXHXHHHLLHHHHT13  
CXLXXHLLHLLLLLT14

CXXXXHLHLHLLLT0

-----  
DESCRIPTION

PAL16L8

SIOCPAL-6.0 (0300 K4)

FÆLLESBUS-BUFFERSTYRING OG SKRIV/LÆS TIL LAGER (LDS=A0)

DDE-821115BS

/RWC RWB /LDG /UDG CONI /DTACK /ACCESS LDS /ADIS GND  
/CEP /RM /UECI /LECI /UDGO /LDGO /UDGI /LDGI /RWM VCC

IF (VCC) RWM = CONI \* /RWB + /CONI \* /RWC

IF (VCC) LDGI = CONI \* /ADIS \* LDG \* /UDG \* /RWB  
+ ACCESS \* RWC \* LDS

IF (VCC) UDGI = CONI \* /ADIS \* /LDG \* UDG \* /RWB  
+ ACCESS \* RWC \* /LDS

IF (VCC) LDGO = CONI \* /ADIS \* LDG \* RWB  
+ ACCESS \* /RWC \* /ADIS \* LDS

IF (VCC) UDGO = CONI \* /ADIS \* UDG \* RWB  
+ ACCESS \* /RWC \* /ADIS \* /LDS

IF (VCC) LECI = ACCESS \* RWC \* LDS \* DTACK

IF (VCC) UECI = ACCESS \* RWC \* /LDS \* DTACK

IF (VCC) RM = RWC \* /CONI \* /CEP \* /ACCESS + RWB \* CONI

FUNCTION TABLE

/RWC	RWB	/LDG	/UDG	CONI	/DTACK	/ACCESS	LDS	/ADIS	/CEP
		RM	UECI	LECI	UDGO	LDGO	UDGI	LDGI	RWM

LL	XXL	XHX	XH	H	XX	XX	XX	X	RM
HL	XXL	XHX	XH	L	XX	XX	XX	X	
LL	XXH	XHX	XH	L	XX	XX	XX	X	
LL	XXL	XLX	XH	L	XX	XX	XX	X	
LL	XXL	XHX	XL	L	XX	XX	XX	X	
HH	XXH	XXX	XX	H	XX	XX	XX	X	
HL	XXH	XXX	XX	L	XX	XX	XX	X	
HH	XXL	XXX	XX	L	XX	XX	XX	X	

LX XXX LLL XX	X HX XX XX X	UECI
HX XXX LLL XX	X LX XX XX X	
LX XXX HLL XX	X LX XX XX X	
LX XXX LHL XX	X LX XX XX X	
LX XXX LLH XX	X LX XX XX X	
LX XXX LLH XX	X XH XX XX X	LECI
HX XXX LLH XX	X XL XX XX X	
LX XXX HLH XX	X XL XX XX X	
LX XXX LHH XX	X XL XX XX X	
LX XXX LLL XX	X XL XX XX X	
XH XLH XHX HX	X XX HX XX X	UDGO
XL XLH XHX HX	X XX LX XX X	
XH XHH XHX HX	X XX LX XX X	
XH XLL XHX HX	X XX LX XX X	
XH XLH XHX LX	X XX LX XX X	
HX XXL XLL HX	X XX HX XX X	
LX XXL XLL HX	X XX LX XX X	
HX XXL XHL HX	X XX LX XX X	
HX XXL XLH HX	X XX LX XX X	
HX XXL XLL LX	X XX LX XX X	
XH LXH XHX HX	X XX XH XX X	LDGO
XL LXH XHX HX	X XX XL XX X	
XH HXH XHX HX	X XX XL XX X	
XH LXL XHX HX	X XX XL XX X	
XH LXH XHX LX	X XX XL XX X	
HX XXL XLH HX	X XX XH XX X	
LX XXL XLH HX	X XX XL XX X	
HX XXL XHH HX	X XX XL XX X	
HX XXL XLL HX	X XX XL XX X	
HX XXL XLH LX	X XX XL XX X	
XL HLH XHX HX	X XX XX HX X	UDGI
XH HLH XHX HX	X XX XX LX X	
XL LLH XHX HX	X XX XX LX X	
XL HHH XHX HX	X XX XX LX X	
XL HLL XHX HX	X XX XX LX X	
XL HLH XHX LX	X XX XX LX X	
LX XXL XLL XX	X XX XX HX X	
HX XXL XLL XX	X XX XX LX X	
LX XXL XHL XX	X XX XX LX X	
LX XXL XLH XX	X XX XX LX X	

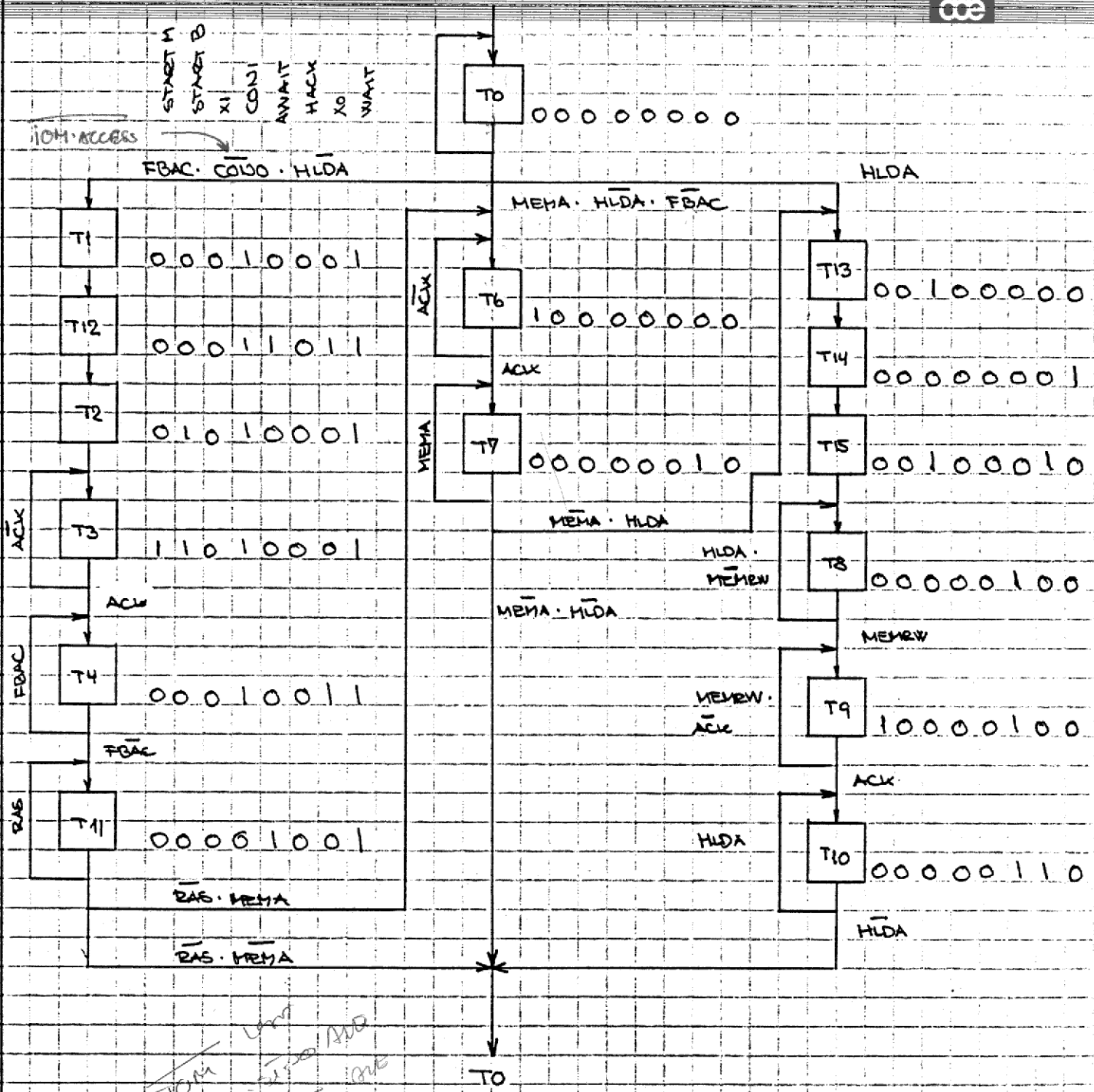


XL LHH XHX HX	X XX XX XH X LDGI
XH LHH XHX HX	X XX XX XL X
XL HHH XHX HX	X XX XX XL X
XL LLH XHX HX	X XX XX XL X
XL LHL XHX HX	X XX XX XL X
XL LHH XHX LX	X XX XX XL X
LX XXL XLH XX	X XX XX XH X
HX XXL XLH XX	X XX XX XL X
LX XXL XHH XX	X XX XX XL X
LX XXL XLL XX	X XX XX XL X
LL XXH XXX XX	X XX XX XX H RWM
LH XXH XXX XX	X XX XX XX L
LL XXL XXX XX	X XX XX XX L
HH XXL XXX XX	X XX XX XX H
LH XXL XXX XX	X XX XX XX L
HH XXH XXX XX	X XX XX XX L

---

DESCRIPTION

Initialer/dato	Side
Revideret	Projekt



MEMA =  
 + IOM  
 + IOM - 500  
 + RAS 50 30

PAL16R8

SIOCPAL-7.0 (SIOC 0300 K1)

ARBITRERING MELLEM FBUS/CPU/DMA.

DDE-821103BS

CLK /MEMA CONO /ACK /FBAC /I HLDA /MEMRW RAS GND  
 GND /Q0 /Q1 /Q2 /Q3 /Q4 /Q5 /Q6 /Q7 VCC

```

Q7 :=          FBAC      * Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T4
  +          ACK        */Q7* Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T3
  +          ACK        */Q7* Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T6
  +          MEMA       * Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T7
  +          ACK        */Q7* Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T9
  +          HLDA       * Q7*/Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T10
  +                                     /Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T1
  +                                     /Q7*/Q6*/Q5*/Q4*/Q3*/Q2* Q1*/Q0*/I ;T14

Q6 :=          /Q7*/Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T2
  +          /ACK       */Q7* Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T3
  +          MEMA*/RAS  */Q7*/Q6*/Q5*/Q4*/Q3*/Q2* Q1* Q0*/I ;T11
  + MEMA */HLDA * /FBAC */Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T0
  +          /ACK       */Q7* Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T6
  +          MEMRW      */Q7*/Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T8
  +          /ACK * MEMRW */Q7* Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T9

Q5 :=          Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1* Q0*/I ;T12
  +          /Q7*/Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T2
  +          /ACK       */Q7* Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T3

Q4 :=          Q7*/Q6*/Q5*/Q4*/Q3* Q2*/Q1*/Q0*/I ;T15
  +          /MEMRW * HLDA */Q7*/Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T8
  +          MEMRW      */Q7*/Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T8
  +          /ACK * MEMRW */Q7* Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T9
  +          ACK        */Q7* Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T9
  +          HLDA       * Q7*/Q6*/Q5*  Q4*/Q3*/Q2*/Q1*/Q0*/I ;T10

Q3 := FBAC * /CONO */HLDA */Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T0
  +          /Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T1
  +          /Q7*/Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T2

```



```

+ /Q7* Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T3
+ FBAC * Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T4
+ Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1* Q0*/I ;T12

Q0 := RAS */Q7*/Q6*/Q5*/Q4*/Q3*/Q2* Q1* Q0*/I ;T11
+ /Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T1
+ /FBAC * Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T4

Q1 := FBAC * /CONO */HLDA */Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T0
+ /Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T1
+ /Q7*/Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T2
+ /Q7* Q6* Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T3
+ Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1*/Q0*/I ;T4
+ RAS * /Q7*/Q6*/Q5*/Q4*/Q3*/Q2* Q1* Q0*/I ;T11
+ Q7*/Q6*/Q5*/Q4* Q3*/Q2* Q1* Q0*/I ;T12
+ /Q7*/Q6*/Q5*/Q4*/Q3* Q2*/Q1*/Q0*/I ;T13

Q2 := /Q7*/Q6*/Q5*/Q4*/Q3*/Q2* Q1*/Q0*/I ;T14
+ HLDA * /Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T0
+ HLDA * /MEMA * Q7*/Q6*/Q5*/Q4*/Q3*/Q2*/Q1*/Q0*/I ;T7
    
```

FUNCTION TABLE:

CLK /MEMA CONO /ACK /FBAC HLDA /MEMRW RAS /I  
 Q6 Q5 Q3 Q0 Q4 Q7 Q1 Q2

```

-----
CXXXXXXXXL LLLLLLLL T0
CXHXXXXXXXXH LLLLLLLL T0
CXLXLLXXH LLHLLLHL T1
CXXXXXXXXH LLHHLHHL T12
CXXXXXXXXH LHLLLHL T2
CXXXXXXXXH HHHLLLHL T3
CXXHXXXXH HHHLLLHL T3
CXXLXXXXH LLHLLHHL T4
CXXXLXXXH LLHLLHHL T4
CXXHXXXXH LLLHLLHL T11
CXXXXXXXXH LLLHLLHL T11
CHXXXXXXXXL LLLLLLLL T0
CLXXHLXXH HLLLLLLL T6
    
```

CXXHXXXXH HLLLLLLL T6  
CXXLXXXXH LLLLHL T7  
CLXXXXXH LLLLHL T7  
CHXXLXXXH LLLLLLL T0  
CXXXXHXXH LLLLLLLH T13  
CXXXXXXH LLLLHL T14  
CHXXLXXXH LLLLHLH T15  
CXXXXXXH LLLHL T8  
CXXXXHXXH LLLHL T8  
CXXXXLXH HLLHL T9  
CXXHXXLXH HLLHL T9  
CXXLXXXXH LLLHHL T10  
CXXXXHXXH LLLHHL T10  
CXXXXLXXH LLLLLLL T0  
CXLXLXXH LLHLHL T1  
CXXXXXXH LHHLHL T12  
CXXXXXXH LHLLHL T2  
CXXXXXXH HHLLHL T3  
CXXLXXXXH LLHLHL T4  
CXXHXXXXH LLHLHL T11  
CLXXXXLH HLLLLLL T6  
CXXLXXXXH LLLLHL T7  
CHXXHXXH LLLLLLLH T13  
CXXXXXXH LLLLHL T14  
CXXXXXXH LLLLHLH T15  
CXXXXXXH LLLHL T8  
CXXXXLXH HLLHL T9  
CXXLXXXXH LLLHHL T10  
CXXXXLXXH LLLLLLL T0

-----  
DESCRIPTION

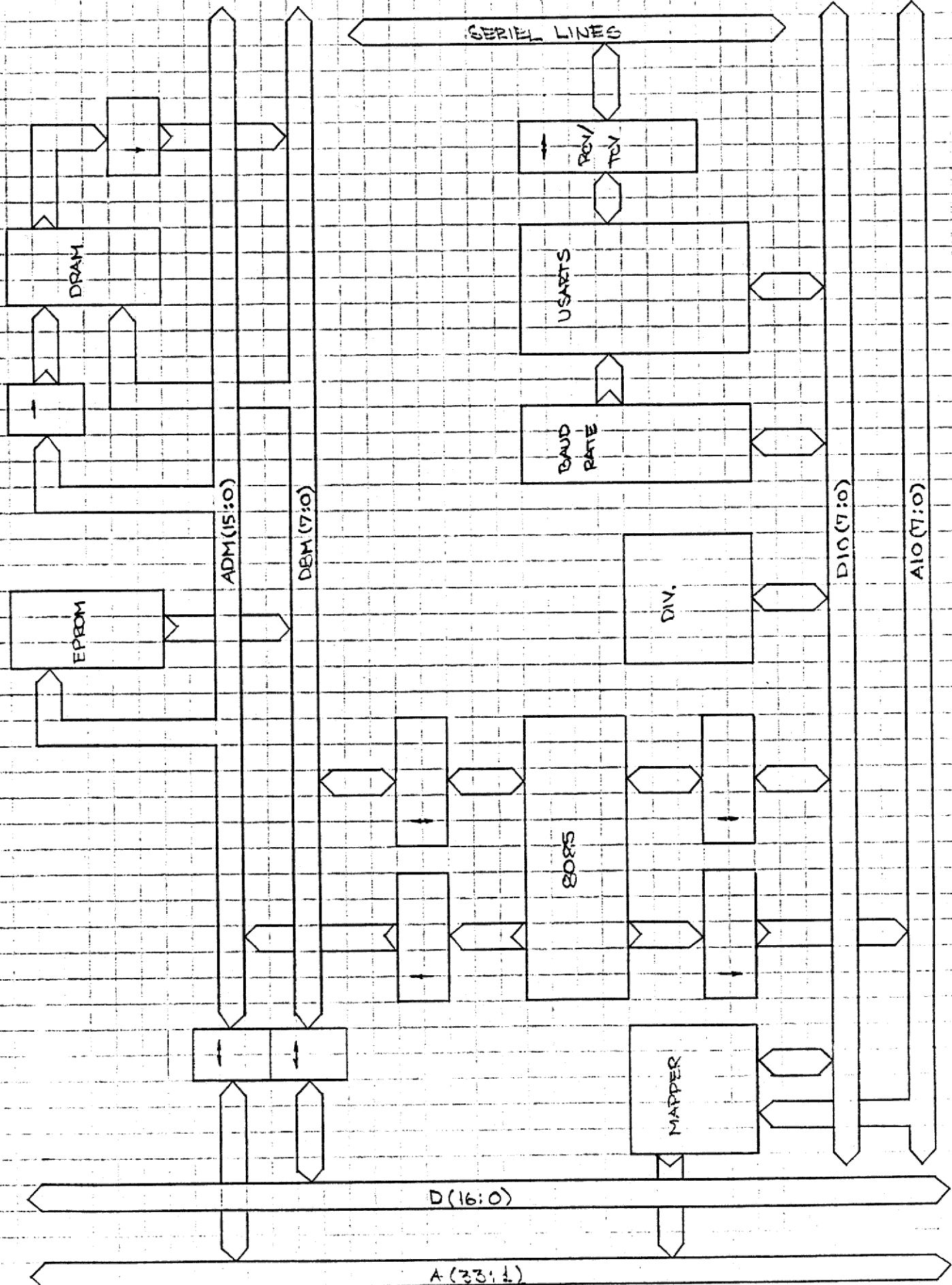
S10C 0300

Initialer/dato

Side

Revideret

Projekt



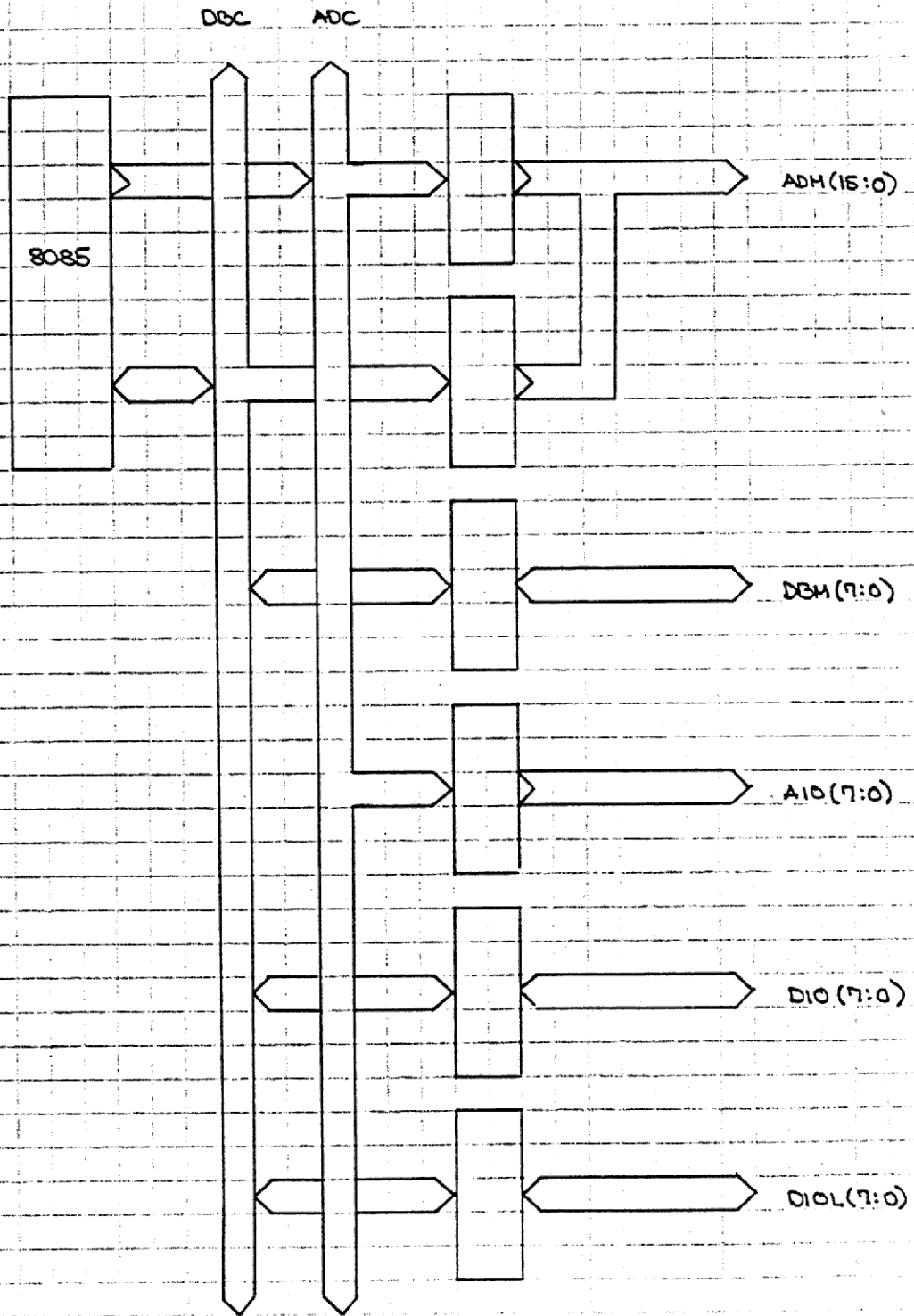
SI0C 0300

Initialer/dato

Side

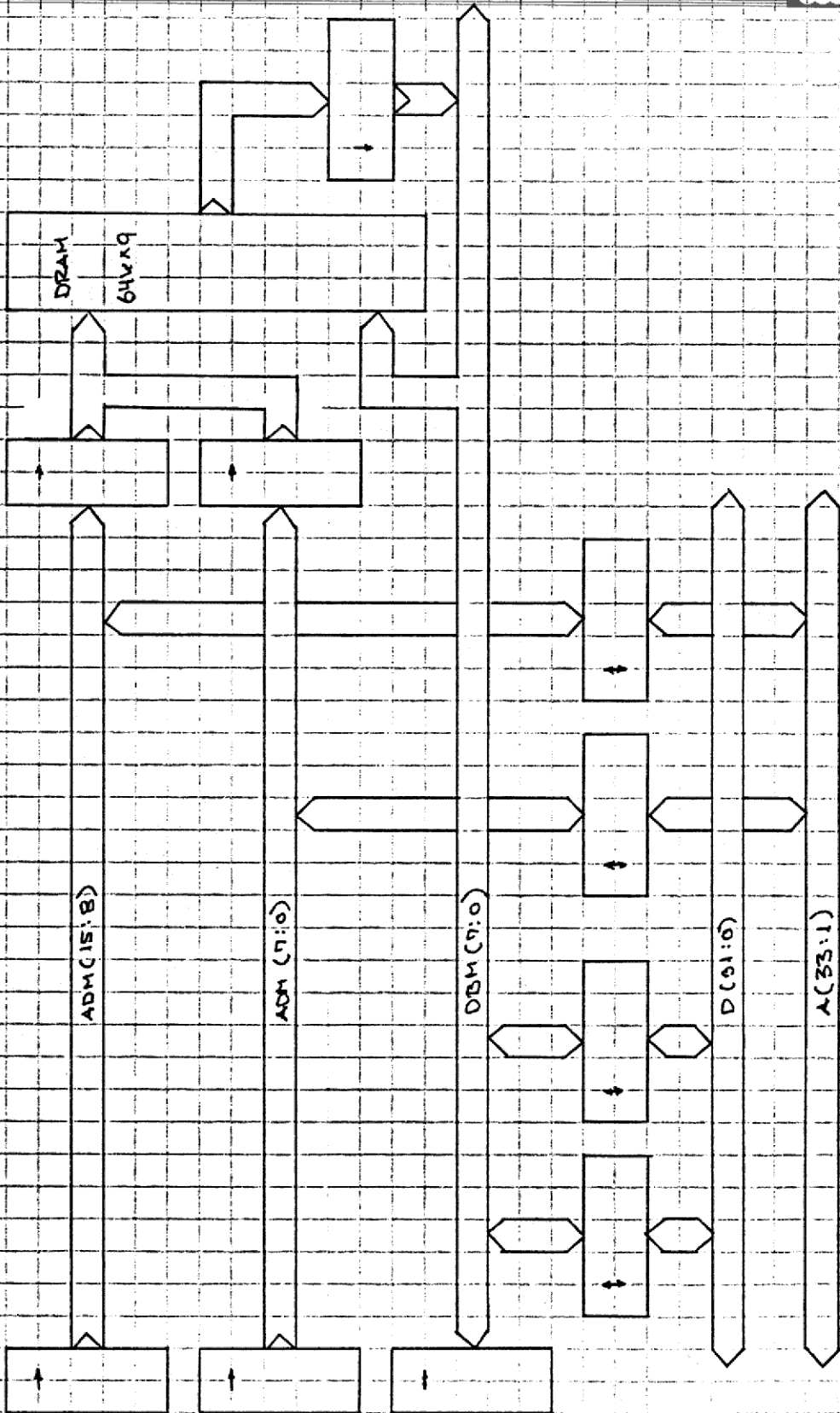
Revlderet

Projekt



SIOC 0300

Initialer/dato	Side
Revideret	Projekt



8085