**Universitatea Politehnica Timișoara**

**Facultatea de Automatica si Calculatoare**

Proiectarea Microsistemelor Digitale

Microsistem cu microprocesorul 8086

Ciocani Daniel

Anul III, grupa 2.1

2017-2018

Tema proiectului

Să se proiecteze un microsistem cu următoarea structură:

- unitate centrală cu microprocesorul 8086;

- 256 KB memorie EPROM, utilizand circuite 27C1024;

- 64 KB memorie SRAM, utilizînd circuite 62256;

- interfaţă serială, cu circuitul 8251, plasată în zona 02A0H – 02A2H sau 0AA0H - 0AA2H, în funcţie de poziţia microcomutatorului S1;

- interfaţă paralelă, cu circuitul 8255, plasată în zona 04D0H–04D6H sau 0CD0H - 0CD6H, în funcţie de poziţia microcomutatorului S2;

- o minitastatură cu 9 contacte;

- 18 led-uri;

- un modul de afişare cu segmente, cu 8 ranguri.

Toate programele în limbaj de asamblare vor fi concepute sub formă de

subrutine. Programele necesare sunt:

- rutinele de programare ale circuitelor 8251 şi 8255;

- rutinele de emisie/ recepţie caracter pe interfaţa serială;

- rutina de emisie caracter pe interfaţă paralelă;

- rutina de scanare a minitastaturii;

- rutina de aprindere/ stingere a unui led;

-rutina de afişare a unui caracter hexa pe un rang cu segmente.

Structura rutinelor (intrări, secvenţe, ieşiri) va fi stabilită de fiecare student.

**Descrierea Hardware**

**1.Unitatea Centrala**

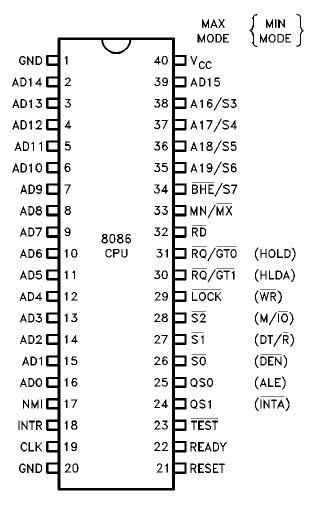
Are in componenta urmatoarele elemente :

1. **Microprocesorul Intel 8086**

Se caracterizeaza prin faptul ca registrele interne si magistrala de date externa sunt pe 16 biti, avand posibilitatea de a adresa direct 1Mo de memorie.

Magistrala de date si cea de adrese sunt multiplexate, iar o parte din terminalele de comanda au rol dublu ceea ce a permis incapsularea circuitului intr-o capsula cu doar 40 terminale. O alta caracteristica a microprocesorului este faptul ca asigura o viteza marita de lucru atat datorita frecventei tactului (5MHz) cat si unei structuri interne bazata pe conceptul de suprapunere care permite aducerea din memorie, in avans, a instructiunilor, in timpul unor cicluri fara accese la magistrala. Pentru aceasta aplicatie se foloseste microprocesorul 8086 in modul minim, in care microprocesorul genereaza el insusi semnalele necesare

transferurilor cu memoria si cu porturile de intrare/iesire.

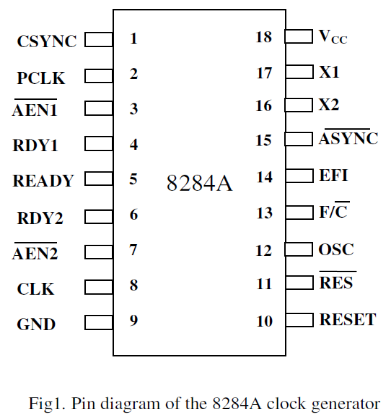


1. **un circuit 8284A (generator de tact)**

Circuitul 8284A este folosit pentru a genera tactul procesorului dar si

pentru a genera tactul circuitelor specializate in lucrul cu interfete (tactul acestor

circuite este egal cu jumatate din valoarea tactului microprocesorului 8086).



1. 2 circuite 74LS245 ( amplificare/separare a magistralelor de date)

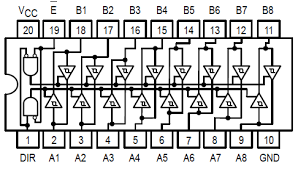
Circuitiul 74LS245 este folosit pentru amplificarea respectiv separarea

magistralelor bidirectionale si este alcatuit din 8 perechi de porti cu 3 stari

bidirectionale cu o intrare de validare pentru toate portile (/G) activa la 0 logic si

o intrare de stabilire a directiei de transfer. Daca intrarea de validare este pe 1

logic portile se afla in a treia stare.



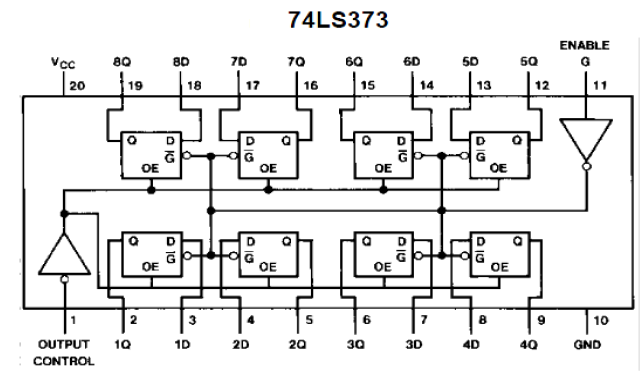
1. 3 circuite 74LS373 (demultiplexarea liniilor de adresa)

Circuitul 74LS373 este folosit pentru demultiplexarea liniilor de

adrese.Este un registru cu iesiri cu 3 stari, alcatuit din 8 bistabile. Are o intrare

de validare pentru toate iesirile /OC(daca este pe 1 logic bistabilele trec in a 3-a

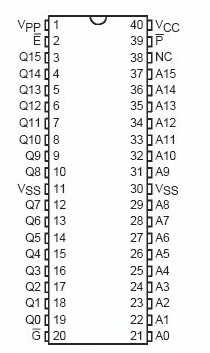
stare) si o intrare pentru incarcarea bistabilelor G activa la 1 logic.



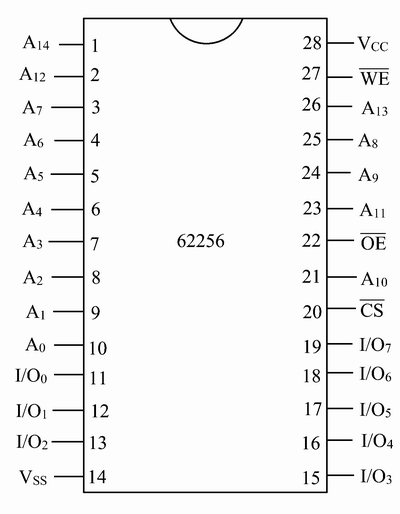
**2.Memorii**

- **Conectarea memoriilor**:

Avem nevoie de 4 circuite de tip EPROM 27C1024, care au o capacitate de 64KB, si 2 circuite de tip SRAM, cu o capacitate de 32KB.



* EPROM 27C1024
* Capacitate : 64KB



* SRAM 62256
* Capacitate : 32KB

**- Decodificarea memoriilor**

Pentru decodificarea memoriilor am folosit urmatoarea harta :

C0000H – FFFFFH - > 4 circuite EPROM 27C1024, avand capacitatea de 4x 64KB

20000H – 2FFFFH - > 2 circuite SRAM 62256, avand capacitatea de 2x 32KB

Notam : C1 - memoria EPROM cu circuite 27C1024

C2 - memoria SRAM cu circuite 62256

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A19 | A18 | A17 | A16 | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | C |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | C1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | C2 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Folosind decodificarea completa :

SelC1 = A19 · A18

SelC2 = Ā19 · Ā18 · A17 · Ā16

**3.Interfata seriala si paralela**

Interfata seriala 8251 :

- este produs de Intel

- inntrarile si iesirile sunt compatibile TTL.

- conversia TTL – EIA e realizata de cip-ul MAX232

- transmisie full-duplex cu buffer dublu

Interfata paralela 8255:

- este produs de Intel

- este compatibil TTL

- 24 de pini IO programabili

- se poate programa in 3 moduri de functionare

Harta memoriei pentru interfata serial si paralela este :

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A15 | A14 | A13 | A12 | A11 | A10 | A9 | A8 | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 |
| IS | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| IP | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 |

IS -> Interfata seriala

IP -> Interfata paralela

Pentru selectie am ales bitii A11 – A8 .

SelIS = Ā11 · Ā10 · A9 · Ā8

SelIP = Ā11 · A10 · Ā 9 · Ā8

**4.Programarea circuitelor**

SERIALC equ 02A2h

SERIALD equ 02A0h

PARARCC equ 04D6h PARAA equ 04D0h PARAB equ 04D2h PARAC equ 04D4h

CHAR\_0 equ 0C0h CHAR\_1 equ 0F9h CHAR\_2 equ 0A2h CHAR\_3 equ 0B0h CHAR\_4 equ 099h CHAR\_5 equ 092h CHAR\_6 equ 082h CHAR\_7 equ 0F8h CHAR\_8 equ 080h

; int 20h

; Rutina de programare a interfetei seriale 8251

; 8 biti, fara paritate, factor x16, rata 9600bps

; /RTS = 1, comanda receptie, comanda transmisie, /DTR = 1

INT\_20:

mov DX, 01002h

mov AL, 014h

out DX, AL

mov DX, 01000h

mov AL, 41h

out DX, AL

mov DX, SERIALC

mov AL, 0CEh

out DX, AL

mov AL, 015h

out DX, AL

IRET

;int 21h

;rutina de transmitere a caracterului din CL

INT\_21:

MOV DX, SERIALC

IN AL,DX ; citire si testare rang TxRDY din cuvantul de stare

RCR AL,1

JNC INT\_21

MOV AL,CL ; se preia data din registrul CL

MOV DX, SERIALD

OUT DX,AL

IRET

;int 22h

;rutina de receptie a caracterului in CL

INT\_22:

MOV DX, SERIALC

IN AL,DX ; citire si testare rang RxRDY din cuvântul de stare

RCR AL,2

JNC INT\_22

MOV DX,SERIALD

IN AL,DX ; se preia data de la 8251

MOV CL,AL ; se depune data in registrul CL

IRET

;int 23h

;rutina de programare a interfetei paralele 8255

; grup A - mod 0

; port A - iesire

; port C superior - iesire

; grup B - mod 0

; port B - iesire

; port C inferior - intrare

INT\_23:

MOV DX, PARARCC

MOV AL,81H

OUT DX,AL

IRET

;int 24h

;rutina de emisie a unui caracter din CL

INT\_24:

MOV DX,PARAC

IN AL,DX

RCR AL,1

JNC INT\_24

MOV AL,CL ; se preia caracterul din registrul CL

MOV DX,PARAA ; si e pun pe portul A care este cu memorare

OUT DX,AL

MOV AL,01H ; se trimite un semnal de strobe pe portul B

MOV DX,PARAB

OUT DX,AL ; /STB = 1

AND AL,00H

OUT DX,AL ; /STB = 0

OR AL,01H

OUT DX,AL ; /STB = 1

IRET

;int 25h

;rutina de scanare a tastaturii

;tasta apasata se stocheaza in CL

; 0 1 2

; 3 4 5

; 6 7 8

INT\_25:

;se pune 0 pe prima coloana si se verifica tastele 0 3 6

MOV AL, 0FEh

MOV DX,840H OUT DX,AL

;se verifica pe rand tastele

MOV 08C0H, DX

IN AL, DX ;citeste prima coloana

MOV CL, 00h ;seteaza ca si activa tasta 0

AND AL, 80H ;verifica daca tasta activa e 0

JNZ INT\_25\_END ;daca e se iese din intrerupere, daca nu se

;scaneaza tastele urmatoare

MOV CL, 03h ;seteaza ca si activa tasta 3

AND AL, 80H ;verifica daca tasta activa e 3

JNZ INT\_25\_END ;daca e se iese din intrerupere, daca nu se

;scaneaza tastele urmatoare

MOV CL, 06h ;seteaza ca si activa tasta 6

AND AL, 60H ;verifica daca tasta activa e 6

JNZ INT\_25\_END ;daca e se iese din intrerupere, daca nu se

;scaneaza tastele urmatoare

;se pune 0 pe coloana a doua si se verifica tastele 1 4 7

MOV AL, 0FDh

MOV DX,840H

OUT DX,AL

;se verifica pe rand tastele

MOV DX, 08C0H

IN AL, DX ;citeste a doua coloana

;TASTA 1

MOV CL, 01h

AND AL, 80H

JNZ INT\_25\_END

;TASTA 4

MOV CL, 05h

AND AL, 40H

JNZ INT\_25\_END

;TASTA 7

MOV CL, 07h

AND AL, 40H

JNZ INT\_25\_END

;se pune 0 pe coloana a treia si se verifica tastele 2 5 8

MOV AL, 0FBh

MOV DX,840H OUT DX,AL

;se verifica pe rand tastele

MOV DX, 08C0H

IN AL, DX ;citeste a doua coloana

;TASTA 2

MOV CL, 02h

AND AL, 80H

JNZ INT\_25\_END

;TASTA 5

MOV CL, 05h

AND AL, 40H

JNZ INT\_25\_END

;TASTA 8

MOV CL, 08h

AND AL, 20H

JNZ INT\_25\_END

Rutina de aprindere a unui LED, in CL se afla numarul LED-ului

INT\_26:

MOV AX,01h;

SHL AX,CL

MOV DX,0CC0H

OUT DX, AL

MOV DX,0C40H

MOV AL, AH

OUT DX, AL

IRET

Rutina de afisare a unui carater hexa pe un anumit rang (CL caracterul ce va fi afisat, CH - rangul)

INT\_27:

CMP CL,00h

JE PRINT\_0

CMP CL,01h

JE PRINT\_1

CMP CL,02h

JE PRINT\_2

CMP CL,03h

JE PRINT\_3

CMP CL,04h

JE PRINT\_4

CMP CL,05h

JE PRINT\_5

CMP CL,06h

JE PRINT\_6

CMP CL,07h

JE PRINT\_7

CMP CL,08h

JE PRINT 8

PRINT\_0: MOV AL, CHAR\_0

JMP PRINT

PRINT\_1: MOV AL, CHAR\_1

JMP PRINT

PRINT\_2: MOV AL, CHAR\_2

JMP PRINT

PRINT\_3: MOV AL, CHAR\_3

JMP PRINT

PRINT\_4: MOV AL, CHAR\_4

JMP PRINT

PRINT\_5: MOV AL, CHAR\_5

JMP PRINT

PRINT\_6: MOV AL, CHAR\_6

JMP PRINT

PRINT\_7: MOV AL, CHAR\_7

JMP PRINT

PRINT\_8: MOV AL, CHAR\_8

JMP PRINT

PRINT:

CMP CH,00h

JE PRINT\_r0

CMP CH,01h

JE PRINT\_r1

CMP CH,02h

JE PRINT\_r2

CMP CH,03h

JE PRINT\_r3

PRINT\_r0:

MOV DX,04C0H

OUT DX,AL

JMP INT\_27\_END

PRINT\_r1:

MOV DX,0440H

OUT DX,AL

JMP INT\_27\_END

PRINT\_r2:

MOV DX,00C0H

OUT DX,AL

JMP INT\_27\_END

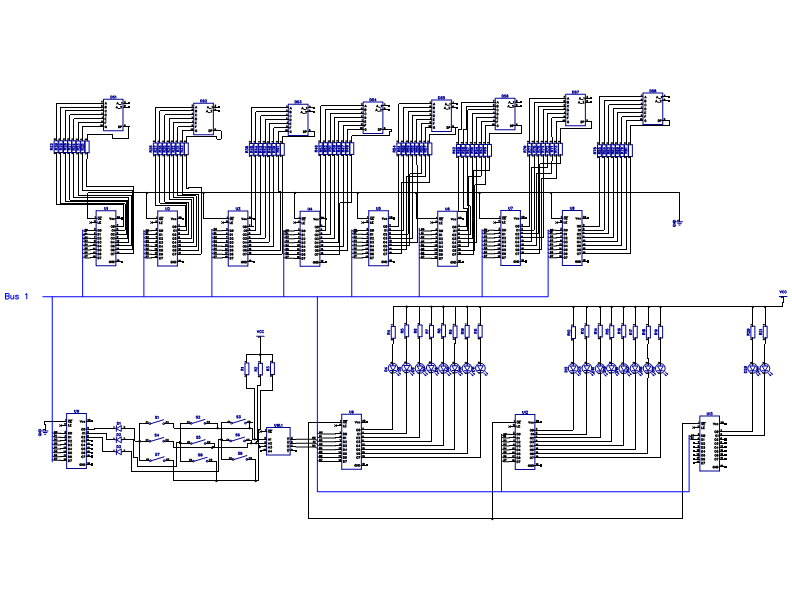
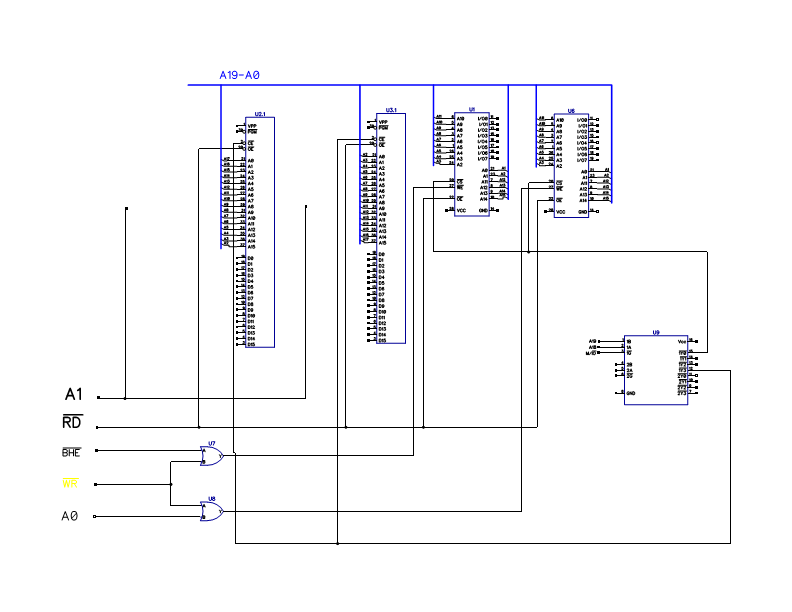
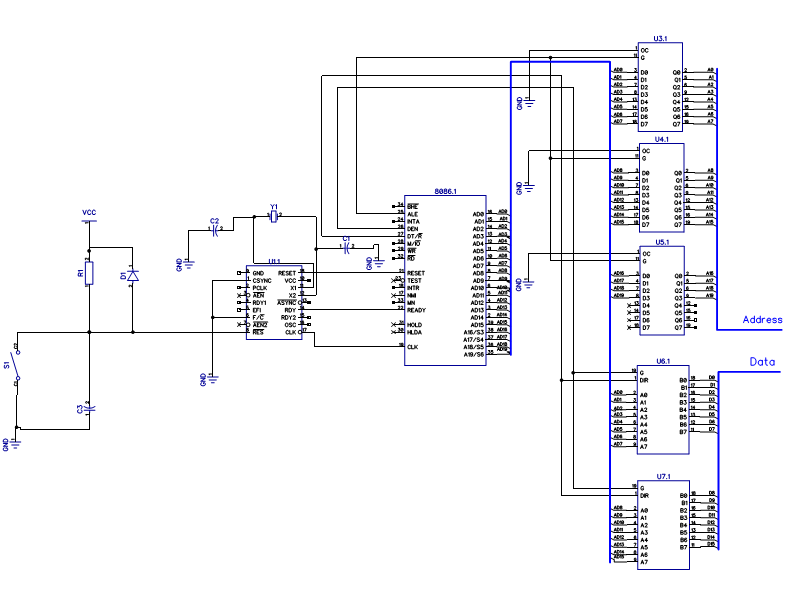
PRINT\_r3:

MOV DX,0040H

OUT DX,AL

JMP INT\_27\_END

INT\_27\_END: IRET



**Bibliografie**

1. POPA, Mircea – “Proiectarea Microsistemelor Digitale”, editura Orizonturi Universitare, Timisoara 2003;

2. POPA, Mircea – “Sisteme cu microprocesoare”, editura Orizonturi Universitare, Timisoara 2003;

3. https://sites.google.com/site/labpmd/