

Proiect la

PROIECTAREA CU MICROPROCESOARE

MICROSISTEM CU PROCESORUL 8086

Coordonatori: **Prof. emer. dr. ing. Mircea POPA**

Ș. I. dr. ing. Sergiu NIMARĂ

Nume și prenume student: **BOȚOC Indi**

An de studii: **III licență**

Domeniul: **Calculatoare și Tehnologia Informației**

Specializarea: **Calculatoare**

0. Tema proiectului

Să se proiecteze un microsistem cu următoarea structură:

- unitate centrală cu microprocesorul 8086;
- 128 KB memorie EPROM, utilizând circuite 27C512;
- 64 KB memorie SRAM, utilizând circuite 62256;
- interfață serială, cu circuitul 8251, plasată în zona 04D0H – 04D2H sau 05D0H – 05D2H, în funcție de poziția microcomutatorului S1;
- interfață paralelă, cu circuitul 8255, plasată în zona 0250H – 0256H sau 0A50H – 0A56H, în funcție de poziția microcomutatorului S2;
- o minitastatură cu 9 contacte;
- 10 led-uri;
- un modul de afișare cu 7 segmente, cu 8 ranguri (se pot afișa maxim 8 caractere hexa simultan);
- un modul LCD, cu 2 linii a câte 16 caractere fiecare, cu o interfață la alegerea studentului.

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

Programele necesare sunt:

- rutinele de programare ale circuitelor 8251 și 8255;
- rutinele de emisie/ recepție caracter pe interfața serială;
- rutina de emisie caracter pe interfață paralelă;
- rutina de scanare a minitastaturii;
- rutina de aprindere/ stingere a unui led;
- rutina de afișare a unui caracter hexa pe un rang cu segmente.

Structura rutinelor (intrări, secvențe, ieșiri) va fi stabilită de fiecare student.

1.Descrierea Hardware

1.1. Unitatea centrală

1.1.1. Microprocesorul Intel 8086

Microprocesorul Intel 8086 este primul procesor pe 16 biți (adică registrele interne și magistrala de date externă sunt pe 16 biți). Procesorul are posibilitatea de a adresa direct 1MB de memorie, unde, datorită frecvenței ridicate a tactului de 5 MHz, se permite aducerea în avans a instrucțiunilor.

Microprocesorul 8086 este format din **două unități independente** care lucrează în paralel:

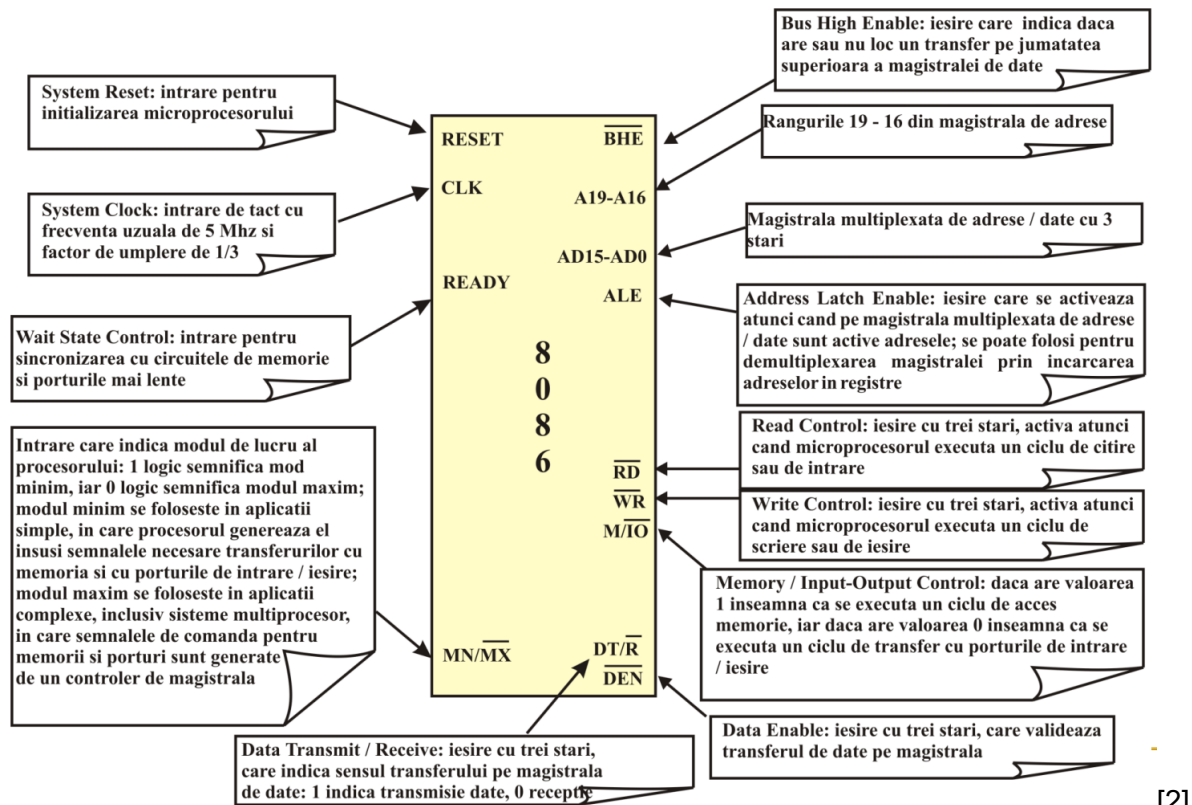
- a. EU (Execution Unit - Unitatea de Execuție):
 - este responsabilă pentru execuția instrucțiunilor.
 - componente principale:
 - ALU (Arithmetical Logic Unit): efectuează operații aritmetice (adunare, scădere etc.) și logice (AND, OR, XOR etc.).
 - Regiștri:
 - Regiștrii generali: AX, BX, CX, DX (pe 16 biți).
 - Pot fi accesați și pe 8 biți: AH/AL, BH/BL, CH/CL, DH/DL.
 - Regiștri de adresare: SP (Stack Pointer), BP (Base Pointer), SI (Source Index), DI (Destination Index).
 - Regiștri de segmentare: CS, DS, SS, ES (Code Segment, Data Segment, Stack Segment, Extra Segment).
 - Registrul de flaguri: conține indicatori (flaguri) care reflectă rezultatul operațiilor (Zero Flag, Carry Flag, Overflow Flag etc.).
 - Unitatea de decodificare: interpretează instrucțiunile și le trimite către ALU.
 - Registrul de instrucțiuni: stochează instrucțiunea curentă.
- b. BIU (Bus Interface Unit - Unitatea Interfață Magistrală):
 - Se ocupă cu interfațarea procesorului cu memoria și porturile de intrare/ieșire.
 - Funcții principale:
 - Pre-fetching: aduce instrucțiunile în avans din memorie și le stochează într-o coadă de instrucțiuni (queue) de 6 bytes.
 - Calcularea adreselor: BIU calculează adresa fizică din memorie folosind regiștrii de segment și cei de offset.
 - Gestionarea magistralei: comunică cu memoria și perifericele prin magistralele de adrese și date.

Caracteristici tehnice ale microprocesorului 8086

- a. Magistrale:
 - Magistrala de date: 16 biți (AD0-AD15) - multiplexată (folosită atât pentru date, cât și pentru adrese).
 - Magistrala de adrese: 20 biți (A0-A19) - permite adresarea directă a 1MB de memorie.
 - Magistrala de control: semnale de sincronizare și control pentru memorie și I/O.
- b. Moduri de adresare:
 - Moduri directe: adresa este specificată direct în instrucțiune.

- Moduri indirecte: adresa este calculată folosind regiștrii (ex. SI, DI, BP, SP).
 - Moduri relative: adresa este specificată relativ la un registru de segment.
- c. Frecvență de tact: funcționează la o frecvență de 5 MHz.
- d. Coada de instrucțiuni: BIU poate prelua instrucțiuni în avans (până la 6 bytes), ceea ce optimizează execuția.[1][3][4]

Terminalele microprocesorului 8086:



[2]

1.1.2. Generatorul de tact 8284A

Generatorul de tact 8284A este un circuit integrat esențial pentru funcționarea microprocesoarelor din familia Intel 8086/8088. Acesta este conceput pentru a genera semnale de tact stabile și alte semnale auxiliare necesare sincronizării microprocesorului cu componentele asociate, precum memoria și perifericele.

Acesta primește un semnal de intrare de la un cristal de cuarț sau un oscilator extern și produce un semnal de tact stabil utilizat de microprocesor. Semnalul CLK este esențial pentru sincronizarea tuturor operațiunilor procesorului. Circuitul generează semnale precum READY, RESET și SYNC, care permit microprocesorului să comunice și să se sincronizeze cu perifericele. De asemenea, 8284A include un circuit care generează un semnal RESET stabil, necesar pentru inițializarea și resetarea microprocesorului. Generatorul gestionează semnalul READY, care permite procesorului să aștepte dacă un periferic nu este pregătit să răspundă.

Generatorul dispune de mai multe intrări, inclusiv Crystal Input (X1, X2), conectate la un cristal de cuarț pentru generarea semnalului de bază, Clock Input (EFI), utilizat ca alternativă la cristal, și RESET, pentru inițializarea sistemului. Ieșirile includ CLK, semnalul de tact pentru microprocesor, READY, semnalul pentru sincronizarea cu perifericele, RESET,

pentru resetarea microprocesorului, și PCLK, semnalul de tact pentru periferice. Este compatibil cu microprocesoarele 8086/8088, care funcționează la frecvențe de ordinul megahertzilor.

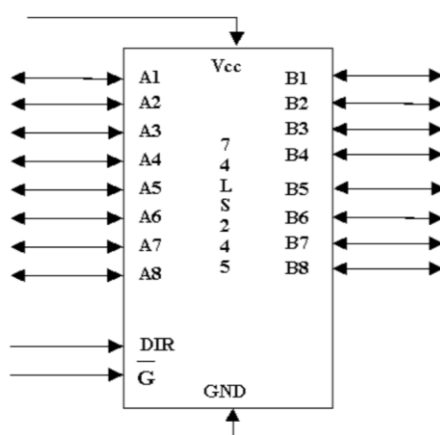
1.1.3. Circuitul 74x245

Circuitul 74x245 este un dispozitiv utilizat pentru transferul bidirecțional de date între două magistrale digitale. Acesta este cunoscut ca un transceiver bidirecțional cu trei stări ("tri-state"), ceea ce înseamnă că poate opera în mod de intrare, ieșire sau deconectare completă, în funcție de semnalele de control.

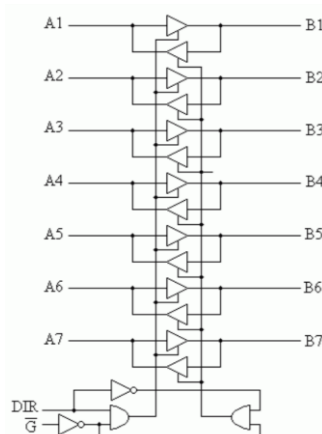
Acesta include opt buffer-e (amplificatoare de izolare) care oferă capacitatea de a amplifica semnalele digitale, reducând degradarea acestora pe magistrale lungi. Circuitul conține, de asemenea, un semnal de activare ("Output Enable", OE), care poate plasa toate ieșirile într-o stare de împreună ("tri-state"), permițând altor dispozitive să utilizeze magistrala respectivă. Această caracteristică este esențială în sistemele cu mai multe dispozitive care partajează aceeași magistrală.

Circuitul 74x245 dispune de următoarele caracteristici tehnice:

- Transfer bidirecțional de date: Permite schimbul de date între două magistrale, controlat de semnalul DIR.
- Ieșiri cu trei stări: Starea deconectată permite conectarea mai multor dispozitive pe aceeași magistrală, evitând conflictele de semnal.
- Opt buffer-e independente: Fiecare buffer poate fi utilizat pentru a izola semnalele și a preveni interferențele.
- Tensiune de alimentare: Funcționează de obicei la 5V (pentru seria clasică 74xx).
- Compatibilitate logică: Este compatibil cu alte circuite din familia 74xx, inclusiv variantele HC, HCT, LS, ALS, etc.
- Capacitate de curent: Poate furniza sau absorbi un curent suficient pentru a acționa mai multe intrări logice conectate la ieșirile sale. [1][3]



Configurația terminalelor



Schema internă

/G	DIR	A8 – A1	B8 – B1
0	0	B8 – B1	Intrări
0	1	Intrări	A8 – A1
1	X	A 3 – a stare	A 3 – a stare

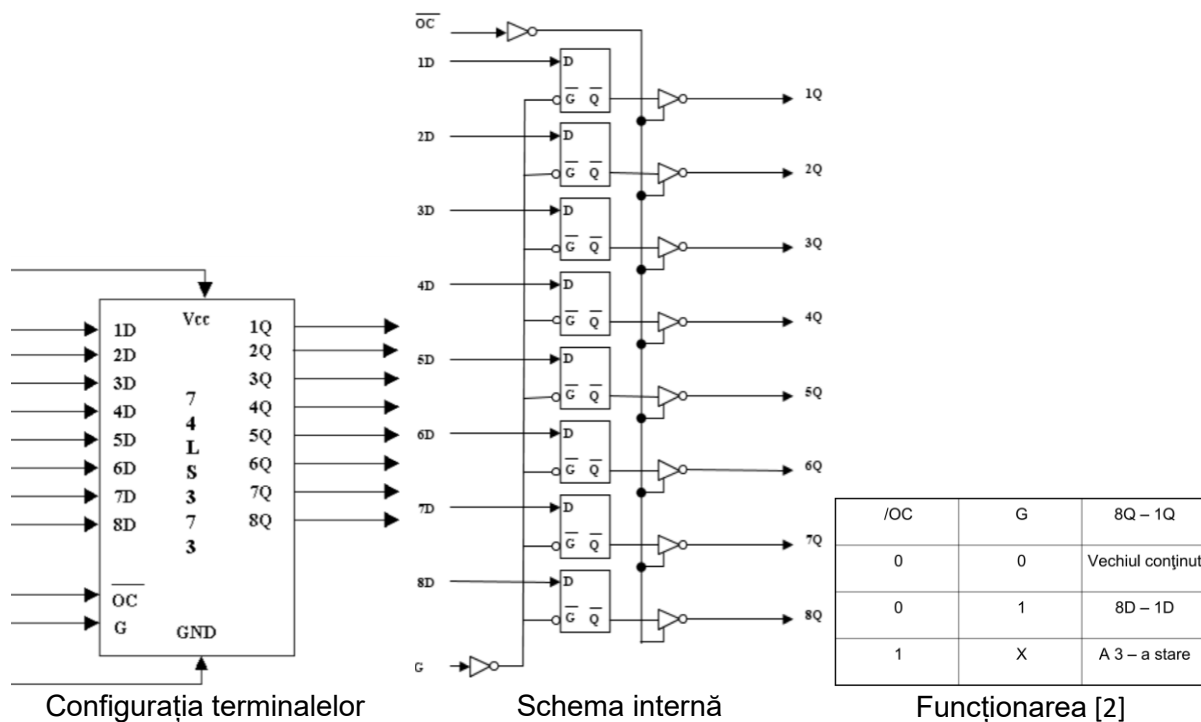
Funcționarea [2]

1.1.4. Circuitul 74x373

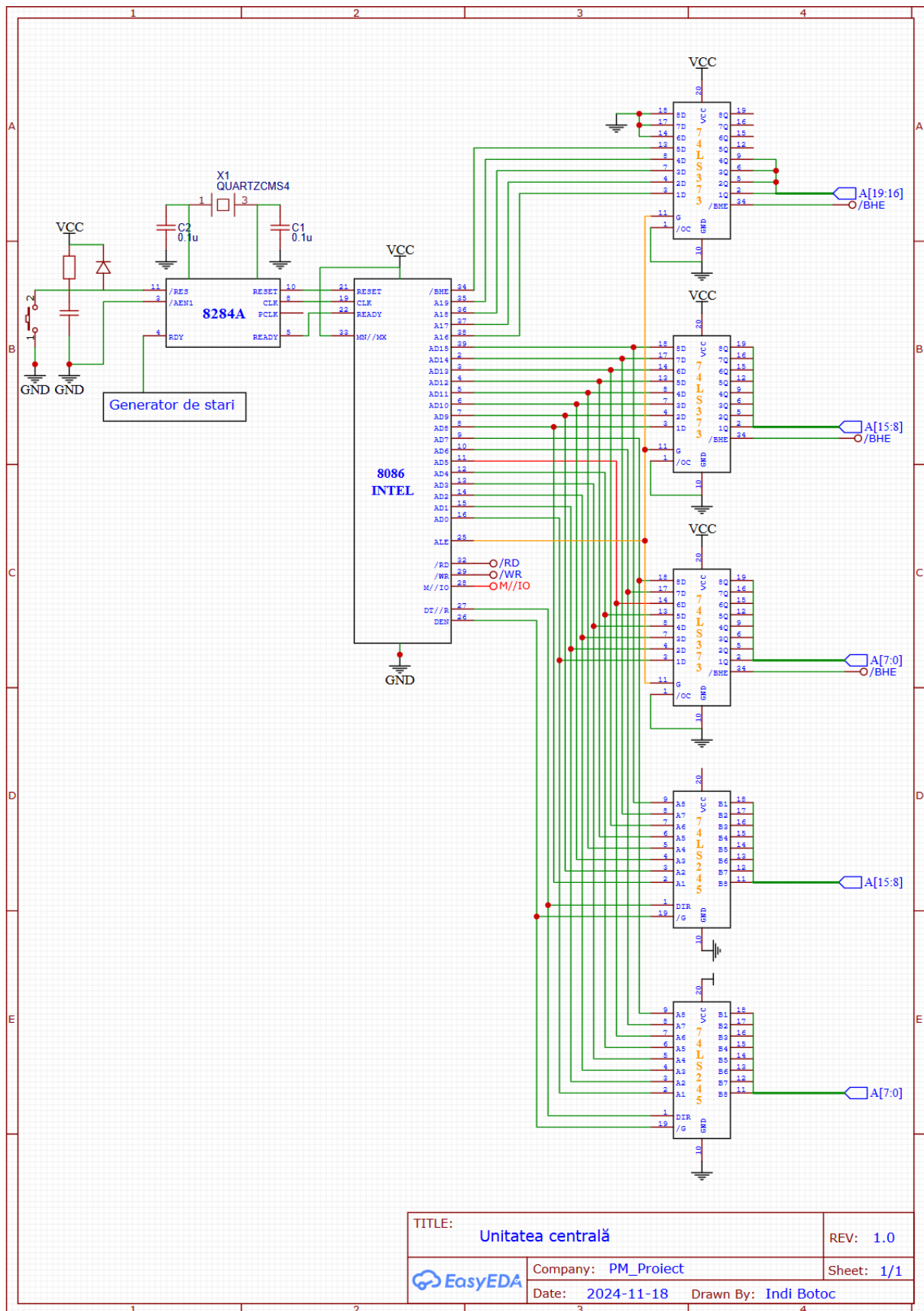
Circuitul 74x373 este un latch de tip D pe 8 biți (transparent latch) utilizat pentru stocarea temporară a datelor și interfațarea magistrelor de date. Este o componentă esențială în sistemele digitale, fiind folosit pentru bufferizarea magistrelor și separarea adreselor în sisteme cu magistrale multiplexate.

Circuitul 74x373 este format din 8 latch-uri D independente, fiecare având o intrare D (date) și o ieșire Q. Funcționarea acestuia este controlată de două semnale: G (Enable) și /OC (Output Control). Semnalul G permite captarea datelor în latch-uri, iar semnalul /OC controlează activarea ieșirilor tri-state.

Când G este activ (1 logic), circuitul este în mod transparent, iar datele aplicate pe intrările D0-D7 sunt transmise direct la ieșirile Q0-Q7. Când G devine inactiv (0 logic), latch-urile rețin ultima valoare a datelor, iar ieșirile rămân constante indiferent de modificările intrărilor D0-D7. Semnalul /OC este utilizat pentru controlul ieșirilor tri-state. Atunci când /OC este activ (0 logic), ieșirile Q0-Q7 sunt active și transmit datele memorate. Când /OC este inactiv (1 logic), ieșirile sunt deconectate (high-impedance), permițând partajarea magistralei cu alte dispozitive.[3]



1.1.5. Schema unității centrale a microprocesorului 8086



1.2. Conectarea memoriilor

1.2.1a. Memoria EPROM

EPROM (erasable programmable read-only memory) reprezintă un tip de memorie nevolatilă care este compusă dintr-o serie de celule de tranzistoare MOSFET (sub forma unei porți logice) care pot stoca câte un bit sub formă de sarcină electrică (o sarcină electrică stocată reprezintă starea 0). Dacă utilizatorul dorește să șteargă biții dintr-o memorie de tip EPROM este nevoie ca memoria să fie expusă la o sursă de lumină ultravioletă.

1.2.1b. Circuitul 27C512

Este un circuit de tip EPROM având capacitatea de 128KB și un timp de acces la memorie de 45 ns.

Circuitul are în componența sa:

- 16 intrări de adrese: A[15:0];
- output: Q[7:0];
- activare chip: $\sim E$;
- activare ieșire / alimentare program: $\sim G$
- tensiune de alimentare: VCC
- ground: VSS [3]

1.2.2a. Memoria SRAM

SRAM (Static Random Access Memory), reprezintă un tip de memorie semiconductoare caracterizată prin faptul că nu necesită cicluri periodice de reîmprospătare a datelor, spre deosebire de memoriile DRAM. Această funcționalitate este posibilă datorită utilizării circuitelor logice combinaționale pentru stocarea fiecărui bit de informație. În SRAM, datele sunt menținute atât timp cât memoria este alimentată cu energie, iar stabilitatea datelor elimină necesitatea unor mecanisme suplimentare de reîmprospătare, ceea ce contribuie la viteza mai mare de acces și la fiabilitatea acestui tip de memorie.[1]

1.2.2b. Circuitul 62256

Este un circuit de tip SRAM având capacitatea de 128KB și un timp de acces la memorie de 45-84 ns.

Acesta are în componența sa:

- 15 intrări de adrese: A[14:0];
- input/output: I/O[7:0];
- linie de validare a ieșirilor: $\sim OE$;
- intrare de activare a scrierii: $\sim WE$;
- intrare de selecție a cipului: $\sim CS$;
- alimentări: VCC și GND. [3]

1.2.3. Decodificarea memoriilor

EPROM: 128 KB avem nevoie

$$\Rightarrow \left\lceil \frac{128 \text{ KB}}{64 \text{ KB}} \right\rceil = 2 \Rightarrow 2 \text{ circuite } 27C512$$

$$\bullet 128 \text{ KB} \Leftrightarrow 2^7 \cdot 2^{10} \text{ B} = 2^{17} \text{ B} \Rightarrow \text{adrese de intrare pe } 17 \text{ b.}$$

DAR EPROM suportă doar 16 b

SRAM: 64 KB avem nevoie

$$\Rightarrow \left\lceil \frac{64 \text{ KB}}{32 \text{ KB}} \right\rceil = 2 \Rightarrow 2 \text{ circuite } 62258$$

$$\bullet 64 \text{ KB} \Leftrightarrow 2^6 \cdot 2^{10} \text{ B} = 2^{16} \text{ B} \Rightarrow \text{adrese de intrare pe } 16 \text{ b}$$

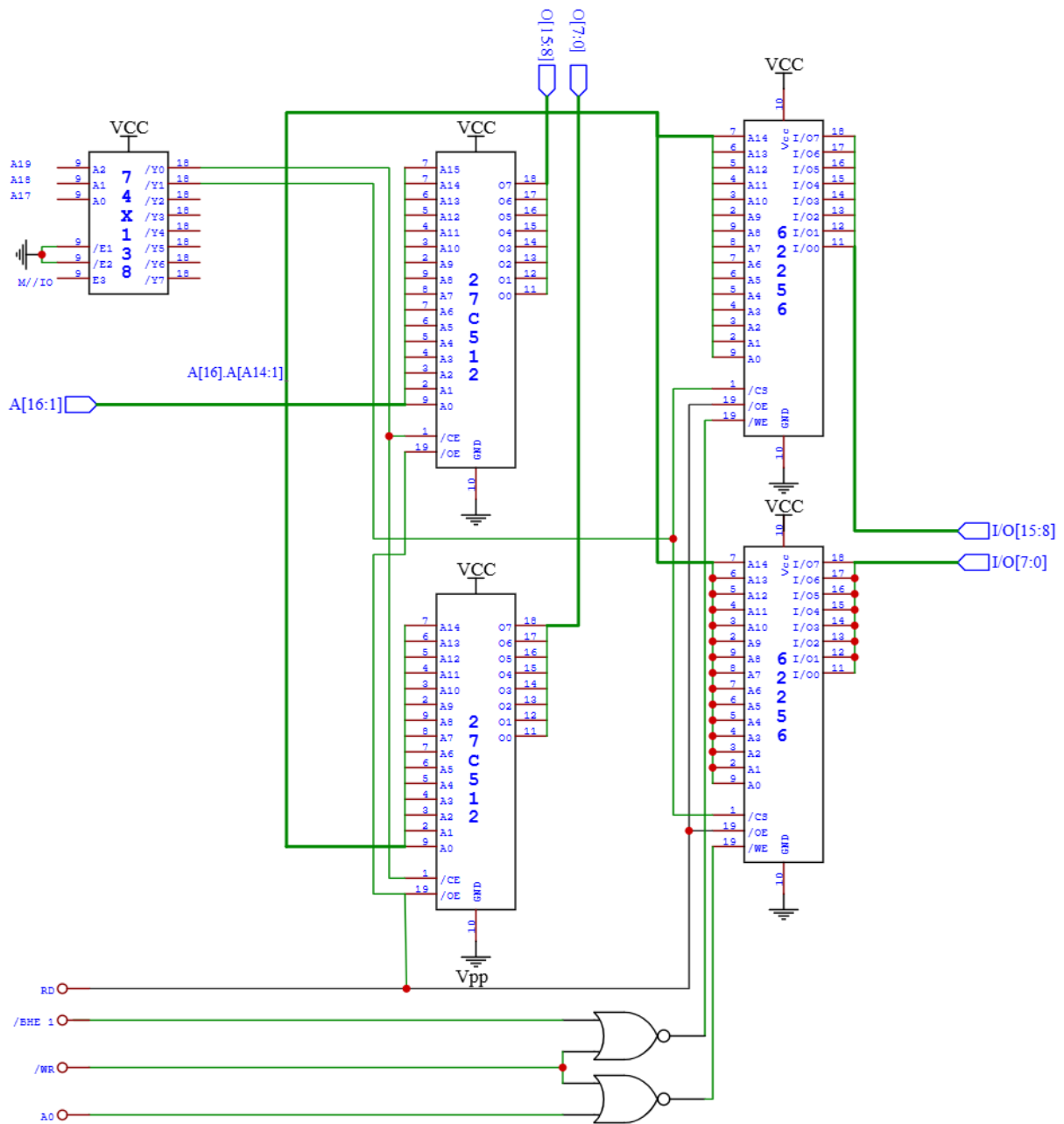
DAR SRAM suportă doar 15 b

MEMORIA	Mem. No.	A19	A18	A17	A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	HEX
EPROM	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00000H
		0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0FFFFH
	2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	10000H
		0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFFFH
SRAM	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20000H
		0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	27FFFH
	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	28FFFH
		0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2FFFFH

Memoria EPROM a fost plasată în zona 00000h-1FFFFh și SRAM în zona 20000h-2FFFFh. În urma decodificării liniilor de adresă a celor două zone de memorie, liniile de adresă A19, A18 și A17 generează semnale de selecție.

$$SEL_{EPROM} = (\sim A19) * (\sim A18) * (\sim A17) \text{ //corespunzător ieșirii /Y0 din 74LS138}$$

$$SEL_{SRAM} = (\sim A19) * (\sim A18) * (A17) \text{ //corespunzător ieșirii /Y1 din 74LS138}$$



1.3. Interfața serială și paralelă

1.3.1. Conectarea porturilor

Interfața serială				Interfața paralelă			
1		2		1		2	
- port comenzi: 04D0h		- port comenzi: 05D0h		- port A: 0250h		- port A: 0A50h	
- port date: 04D2h		- port date: 05D2h		- port B: 0252h		- port B: 0A52h	
				- port C: 0254h		- port C: 0A54h	
				- port RCC: 0256h		- port RCC: 0A56h	

	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
04D0	0	0	0	0	0	1	0	0	1	1	0	1	0	0	0	0
04D2	0	0	0	0	0	1	0	0	1	1	0	1	0	0	1	0
05D0	0	0	0	0	0	1	0	1	1	1	0	1	0	0	0	0
05D2	0	0	0	0	0	1	0	1	1	1	0	1	0	0	1	0
0250	0	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0
0252	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0
0254	0	0	0	0	0	0	1	0	0	1	0	1	0	1	0	0
0256	0	0	0	0	0	0	1	0	0	1	0	1	0	1	1	0
0A50	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0
0A52	0	0	0	0	1	0	1	0	0	1	0	1	0	0	1	0
0A54	0	0	0	0	1	0	1	0	0	1	0	1	0	1	0	0
0A56	0	0	0	0	1	0	1	0	0	1	0	1	0	1	1	0

Folosind Decodificatorul 74x138, se vor activa ieșirile dacă:

- ($\sim E1$) : M/ $\sim IO$
- ($\sim E2$) : **A15|A14|A13|A12|A5** // pentru validarea biților de zero comuni
- E3 : (**A10^A9**)&(**A6&A4**)

Selecția pentru circuitul 8251 în funcție de comutatorul S1:

$$/S51_1 = (\sim A11) \& (\sim A8) \& A7 = (\sim Y4)$$

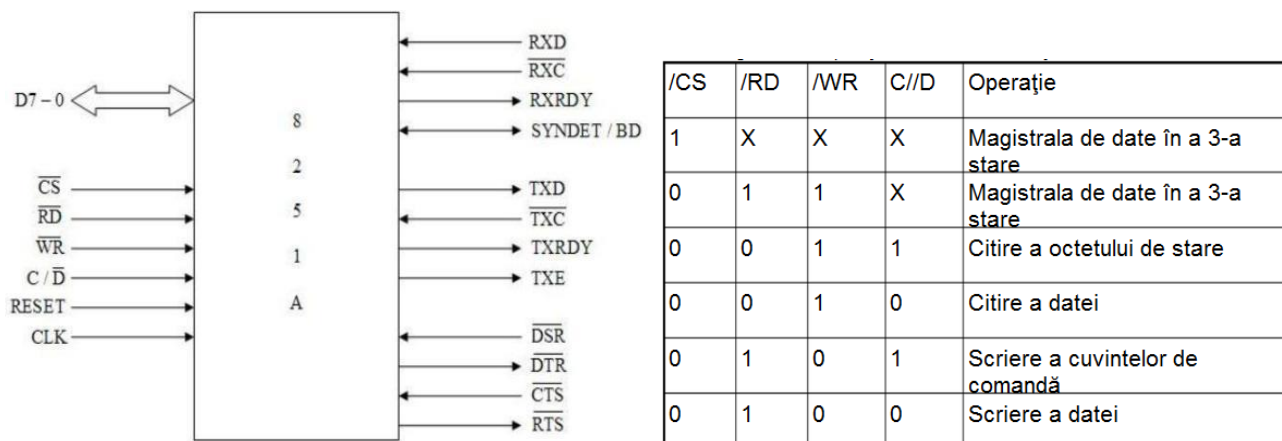
$$/S51_2 = A11 \& (\sim A8) \& (\sim A7) = (\sim Y5)$$

Selecția pt circuitul 8255 în funcție de comutatorul S2:

$$/S55_1 = (\sim A11) \& (\sim A8) \& (\sim A7) = (\sim Y0)$$

$$/S55_1 = (\sim A11) \& (\sim A8) \& A7 = (\sim Y1)$$

Funcționarea circuitului 8251A:



D7-D0(Data Bus): pinii bidirecționali ai magistralei de date.

/CS(Chip Select): activează cipul pentru comunicarea cu procesorul, activ pe 0.

/RD(Read): semnalul de citire, activ pe 0.

/WR(Write): semnalul de scriere, activ pe 0.

C//D(Control|Data): determină tipul de acces, 0(acces la date, citire, scriere) sau 1(acces la registrul de control sau la starea circuitului).

RESET: resetează circuitele interne și inițializează circuitul.

CLK: semnalul de tact pentru sincronizarea funcțiilor interne.

RXD(Receive Data Input): intrarea pentru datele seriale recepționate.

/RXC(Receive Clock): sincronizează recepția datelor seriale pe RXD, activ pe 0.

RXRDY(Receiver Ready): indică faptul că circuitul 8251 a recepționat date noi și sunt gata să fie citite.

SYNDET|BD(Synchronous Detect|Baud Rate Output):

Partea SYNDET: în modul sincron, acest pin indică faptul că s-a detectat un caracter de sincronizare, iar în modul asincron, acest pin poate fi folosit ca semnal output pentru sincronizare.

Partea BD: ieșirea semnalului de tact pentru transmisia asincronă, fiind utilizat pentru controlul ratei de transmisie.

TXD(Transmit Data Output): ieșirea pentru datele seriale transmise.

/TXC(Transmit Clock): sincronizează transmisia datelor seriale pe TXD, activ pe 0.

TXRDY(Transmit Ready): indică faptul că circuitul 8251 este pregătit să transmită noi date.

TXE(transmit Enable): semnal de control ce activează transmiterea datelor.

/DSR(Data Set Ready): semnal care confirmă că datele sunt pregătite pentru a lucra cu ele, activ pe 0.

/DTR(Data Terminal Ready): semnal pentru indicarea disponibilității echipamentului terminal de date, activ pe 0.

/CTS(Clear To Send): semnal de control pentru recepție, activ pe 0.

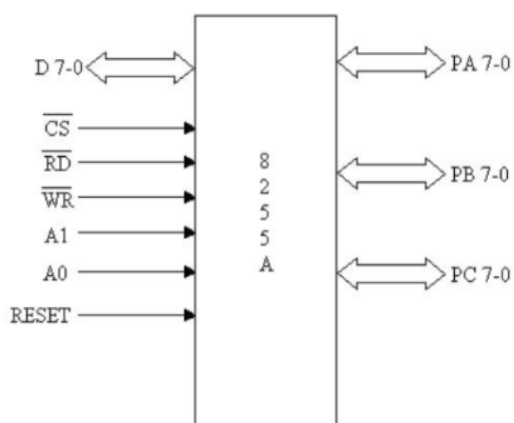
/RTS(Request To Send): semnal de control pentru transmitere, putând fi utilizat pentru a semnala că 8251 este gata să transmită date, activ pe 0. [3]

1.3.3. Interfața paralelă. Circuitul 8255A

O interfață paralelă este un tip de conexiune care transmite mai mulți biți de date simultan, utilizând mai multe fire sau canale în paralel. În contrast cu transferul serial, unde datele sunt transmise bit cu bit, transferul paralel permite trimiterea simultană a 8 biți, iar procesul de transfer include semnale de control. Comunicarea cu magistrala de date a unității centrale de prelucrare se realizează prin intermediul unui buffer, iar datele sunt transmise pe liniile D0 - D7.

Aceste linii sunt utilizate pentru transmiterea atât a datelor, cât și a semnalelor de control sau stare. Într-o interfață paralelă, emițătorul plasează simultan toți biții de date pe liniile de date și trimite un semnal de control (de exemplu, „Strobe”) pentru a informa receptorul că datele sunt gata de a fi citite. Receptorul preia datele de pe liniile de date și trimite un semnal de confirmare („Acknowledge”) către emițător. Acest proces se repetă pentru fiecare pachet de date.

Funcționarea circuitului 8255A:



/CS	/RD	/WR	A1	A0	Operația
0	1	0	0	0	Scriere în portul A
0	1	0	0	1	Scriere în portul B
0	1	0	1	0	Scriere în portul C
0	1	0	1	1	Scriere în portul cuvântului de comandă
0	0	1	0	0	Citire din portul A
0	0	1	0	1	Citire din portul B
0	0	1	1	0	Citire din portul C
0	0	1	1	1	Fără operație – magistrala de date este în a 3-a stare
0	1	1	X	X	Fără operație – magistrala de date este în a 3-a stare
1	X	X	X	X	Magistrala de date este în a 3-a stare

D7-D0(Data Bus): pinii bidirecționali ai magistralei de date.

/CS(Chip Select): activează cipul pentru comunicarea cu procesorul, activ pe 0.

/RD(Read): semnalul de citire, activ pe 0.

/WR(Write): semnalul de scriere, activ pe 0.

A1,A0: pinii de adresare pentru selectarea modurilor de operare ale 8255A.

RESET: resetează circuitele interne și inițializează circuitul.

PA7-PA0: pinii pentru transferul de date pe portul A, ce pot fi configurați ca intrare sau ieșire.

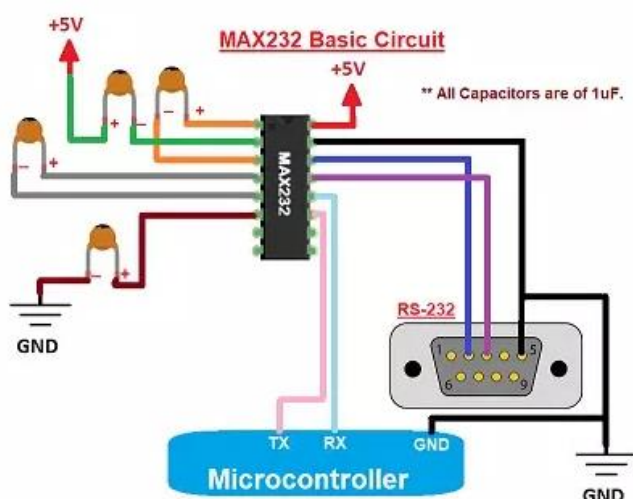
PB7-PB0: pinii pentru transferul de date pe portul B, ce pot fi configurați ca intrare sau ieșire

PC7-PC0: pinii pentru transferul de date pe portul C, ce pot fi pot fi utilizați fie pentru intrarea și ieșirea datelor, fie pentru semnale de control. [3]

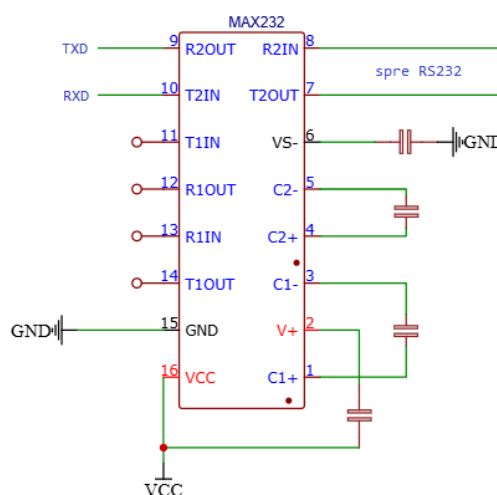
1.3.4. Circuitul MAX232

Circuitul MAX232 este un dispozitiv integrat utilizat pentru a realiza conversia nivelurilor de tensiune între standardele RS-232 și TTL (Transistor-Transistor Logic), fiind esențial pentru comunicarea între dispozitive care operează la tensiuni diferite, cum ar fi computere, microcontrolere și periferice seriale. Standardul RS-232 folosește niveluri de tensiune mai mari, de obicei între +3V și +15V pentru un „1 logic” și între -3V și -15V pentru un „0 logic”, în timp ce logica TTL operează la niveluri mai mici, de obicei între 0V și 5V. MAX232 convertește bidirecțional aceste semnale, facilitând comunicarea între dispozitive. [3]

Structura internă a circuitului include convertoare de nivel de tensiune, două transmițătoare care transformă semnalele TTL în semnale RS-232, două receptoare care convertesc semnalele RS-232 în semnale TTL și un regulator intern de tensiune. Acesta funcționează astfel: semnalele TTL de pe intrările transmițătorului sunt transformate în semnale RS-232 la ieșire, iar semnalele RS-232 de pe intrările receptorului sunt convertite în semnale TTL la ieșire.



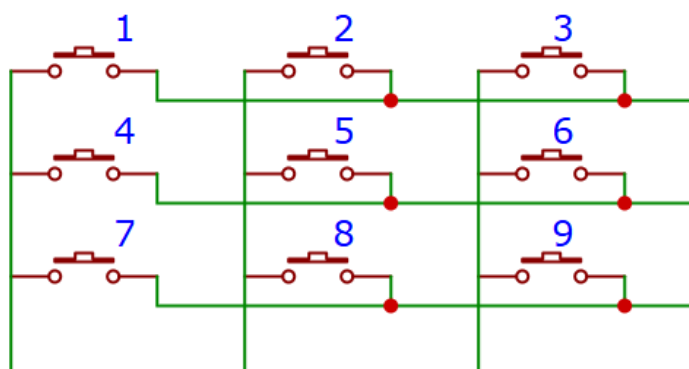
Conectarea porturilor circuitului MAX232 [8]



Versiunea utilizată în proiect

1.4. Ministastatura cu 9 contacte

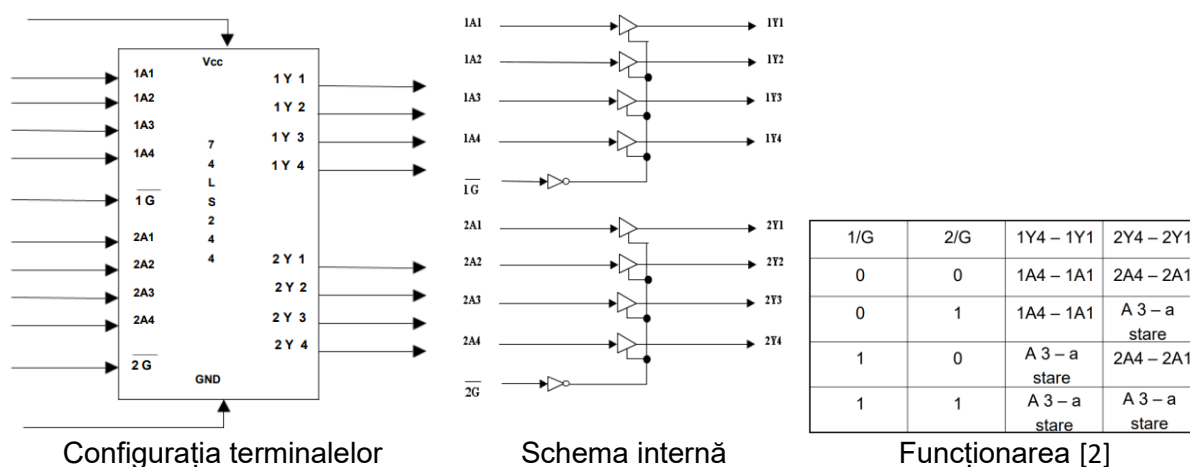
Minitastatura cu 9 contacte este un dispozitiv de intrare format din 9 comutatoare electrice independente, dispuse de obicei într-o matrice 3x3. Fiecare comutator are un contact care închide un circuit electric atunci când este acționat mecanic prin apăsare, generând un semnal digital ce poate fi detectat de un microcontroler sau un alt dispozitiv de procesare. Într-o implementare tipică, liniile și coloanele matricei sunt conectate la pini de intrare-ieșire (GPIO) ai microcontrolerului, utilizând o metodă de scanare matricială pentru a identifica poziția exactă a butonului apăsat.



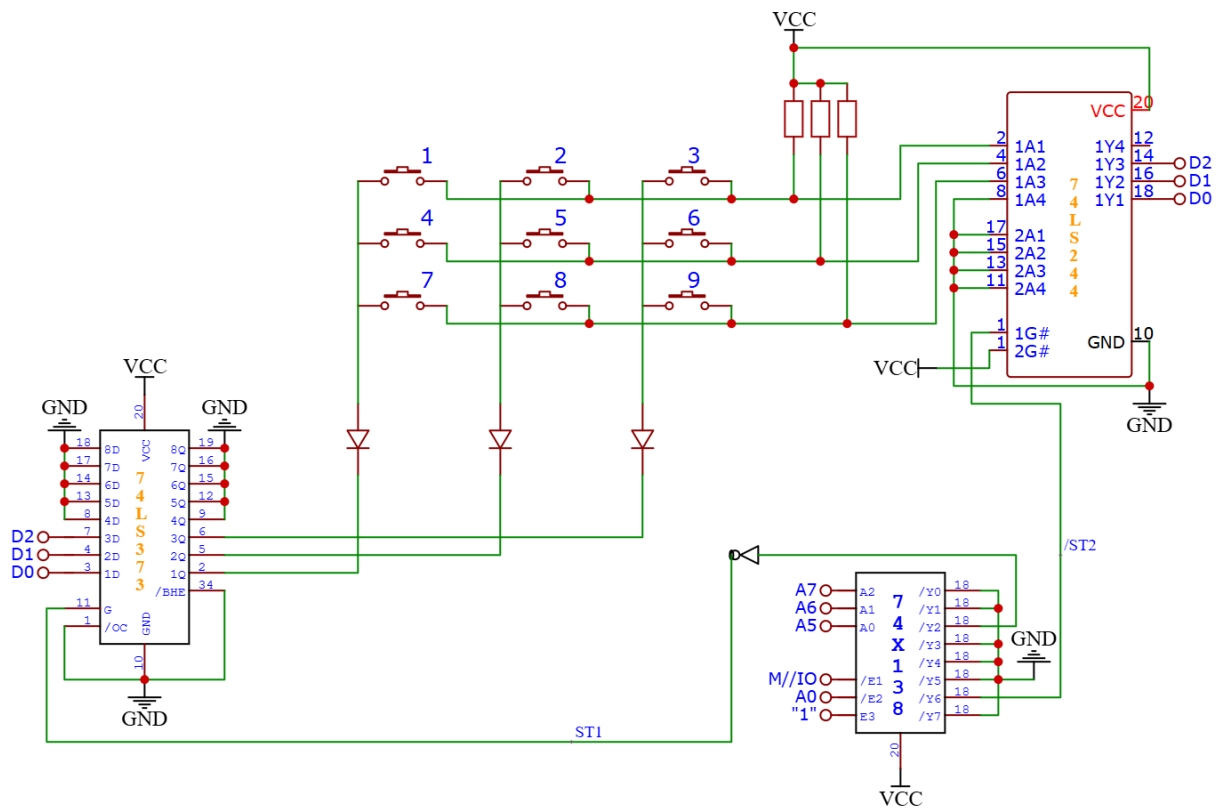
1.4.1. Circuitul 74x244

Circuitul 74LS244 este un buffer bidirecțional cu 8 canale, utilizat în principal pentru amplificarea semnalelor digitale și izolarea circuitelor de pe magistralele de date sau adrese. Este un circuit logic din familia TTL (Transistor-Transistor Logic) cu consum redus de putere (Low Power Schottky) și oferă capacitatea de a conduce sarcini mari, fiind ideal pentru aplicații care necesită transfer rapid de date între componente. [3]

74LS244 este proiectat pentru a asigura o amplificare robustă a semnalelor și pentru a reduce efectele de încărcare pe magistralele comune. Fiecare dintre cele 8 canale ale circuitului poate fi controlat independent, iar dispozitivul este configurat în două grupuri de câte 4 canale, fiecare având o intrare de activare (Enable). Acest design permite controlul selectiv al datelor transmise, ceea ce îl face util pentru multiplexare sau separarea datelor pe magistrale partajate.



1.4.2. Conectarea minitastaturii cu 9 contacte



Circuitul 74LS373 este utilizat pentru a reține semnalele de la tastatură și pentru a le transmite către magistrala de date și adrese.

Circuitul 74LS244 este un buffer/driver care preia semnalele de pe coloanele tastaturii, cu rolul de a:

- izola magistrala de intrare de la restul circuitului.
- întări semnalele pentru a preveni degradarea acestora.

Există 3 comutatoare conectate la rezistențe pull-up (de 10k ohmi). Când un comutator este apăsat, acesta trimite un semnal LOW către circuit (intrare activă pe LOW). Ele permit utilizatorului să interacționeze cu circuitul, trimițând semnale de control.

Minitastatura are o structură matriceală, iar la intersecția liniilor și coloanelor se găsesc tastele.

În portul de ieșire se va scrie 0 logic numai pe o coloană, iar pe restul 1 și se citesc linii. Dacă pe o linie se detectează 0 logic, atunci tasta a fost acționată.

Tastatura are o structură matriceală 3x3, la intersecția liniilor și coloanelor găsim tastele.

Port de intrare - 0840h

Port de ieșire - 08C0h

Ecuatii:

$$ST1 = A10 * /A7 * /A6 * /A5$$

$$ST2 = A10 * /A7 * /A6 * A5$$

	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
ST1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
ST2	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0

1.5. LED-urile

LED (light emitting diode) este o diodă semiconductoare, care emite lumină la polarizarea directă joncțiunii p-n. Pentru ca un led sa fie aprins la ieșirea portului trebuie sa fie 0 logic. Se folosește un registru 74LS373 conectat la magistrală pentru a le menține aprinse sau stinse.

Port de intrare – 0C40h

Port de ieșire – 0CC0h

Ecuatii:

$$L1 = A11 * A10 * /A7 * /A6 * A5$$

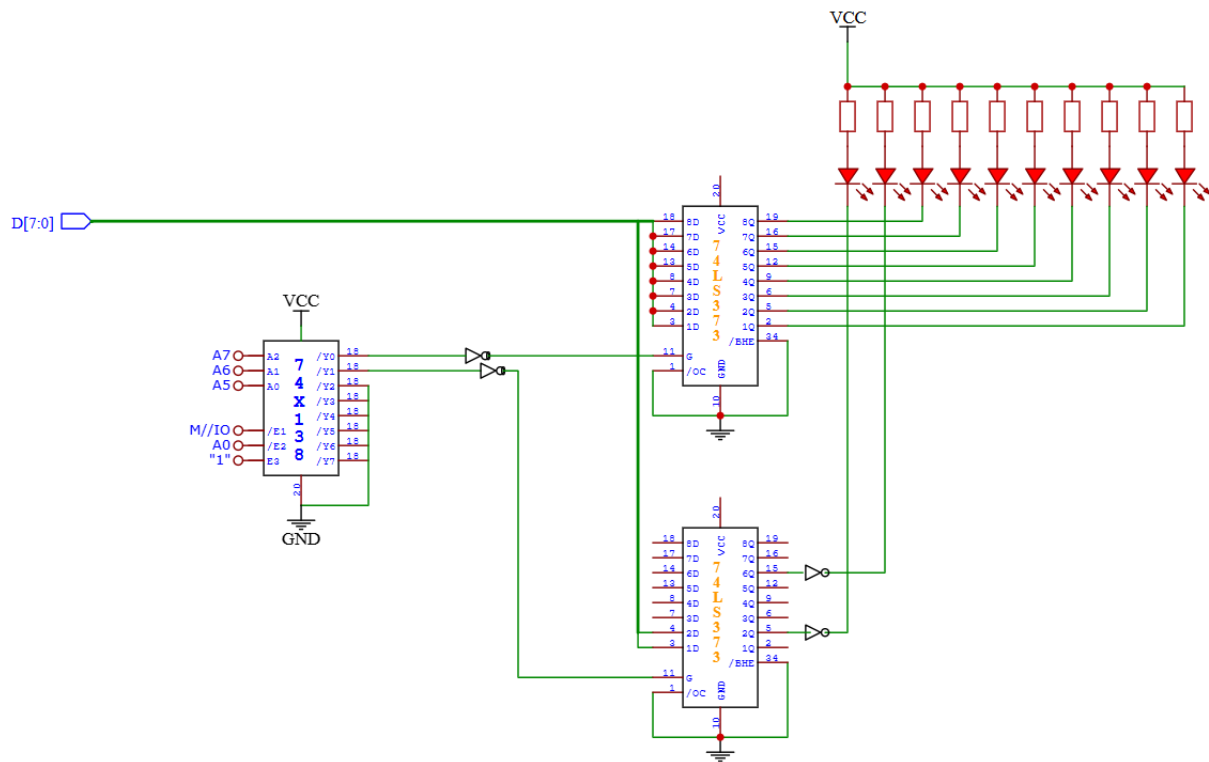
$$L2 = A11 * A10 * /A7 * A6 * A5$$

Interfață	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
L1	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0
L2	0	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0

1.5.1. Conectarea celor 10 LED-uri

10 Led-uri conectate la 2 circuite 74LS373.

Circuitele 74LS373 unde sunt conectate Led-urile au pinul G legat la pinii /Y7 și /Y6 de la decodificatorul de porturi.



1.6. Modulul de afișaj cu 7 segmente

Un afișaj pe 7 segmente este un dispozitiv utilizat pentru a afișa cifre și caractere limitate prin iluminarea selectivă a celor 7 segmente LED dispuse în formă de „8”. Fiecare segment este identificat printr-o literă (a, b, c, d, e, f, g), iar aprinderea sau stingerea acestora este controlată prin aplicarea unui semnal logic la pinii corespunzători.

/SA3	/SA2	/SA1	Modul
0	0	0	M1
0	0	1	M2
0	1	0	M3
0	1	1	M4
1	0	0	M5
1	0	1	M6
1	1	0	M7
1	1	1	M8

Modulul de afisaj este conectat la iesirea /Y0 a decodicatorului de porturi.

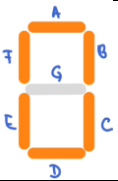
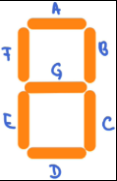


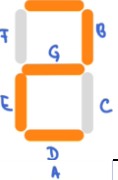
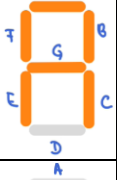
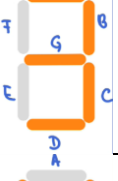

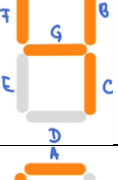
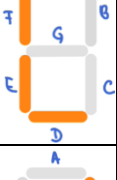

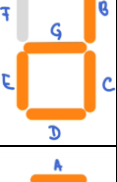

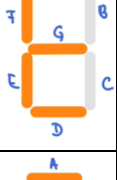

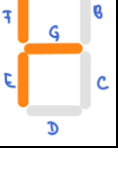
Afisajul este controlat prin rezistente limitatoare de curent.

Circuitele de afișare cu segmente, pot afișa valori hexazecimale, în funcție de setarea intrărilor a-g. Cele 8 ranguri ale afișajului cu segmente au fiecare câte un port de ieșire. Porturile afișajului sunt registre 74LS373 ce au semnalele de selecție SA1, SA2, SA3.

Întrucât vom folosi 8 module de afișare cu 7 segmente, va trebui să mai adăugăm un circuit de decodificare 74LS138, în care vor intra cele 3 semnale de selecție

SA1, SA2 și SA3. Pe baza lor vom decodifica ce modul va afișa ceea ce trebuie.

În cazul afișajului cu anod comun, toate anodurile segmentelor sunt conectate împreună, iar segmentele individuale sunt activate prin aplicarea unui nivel logic „0” la catodul fiecărui segment.

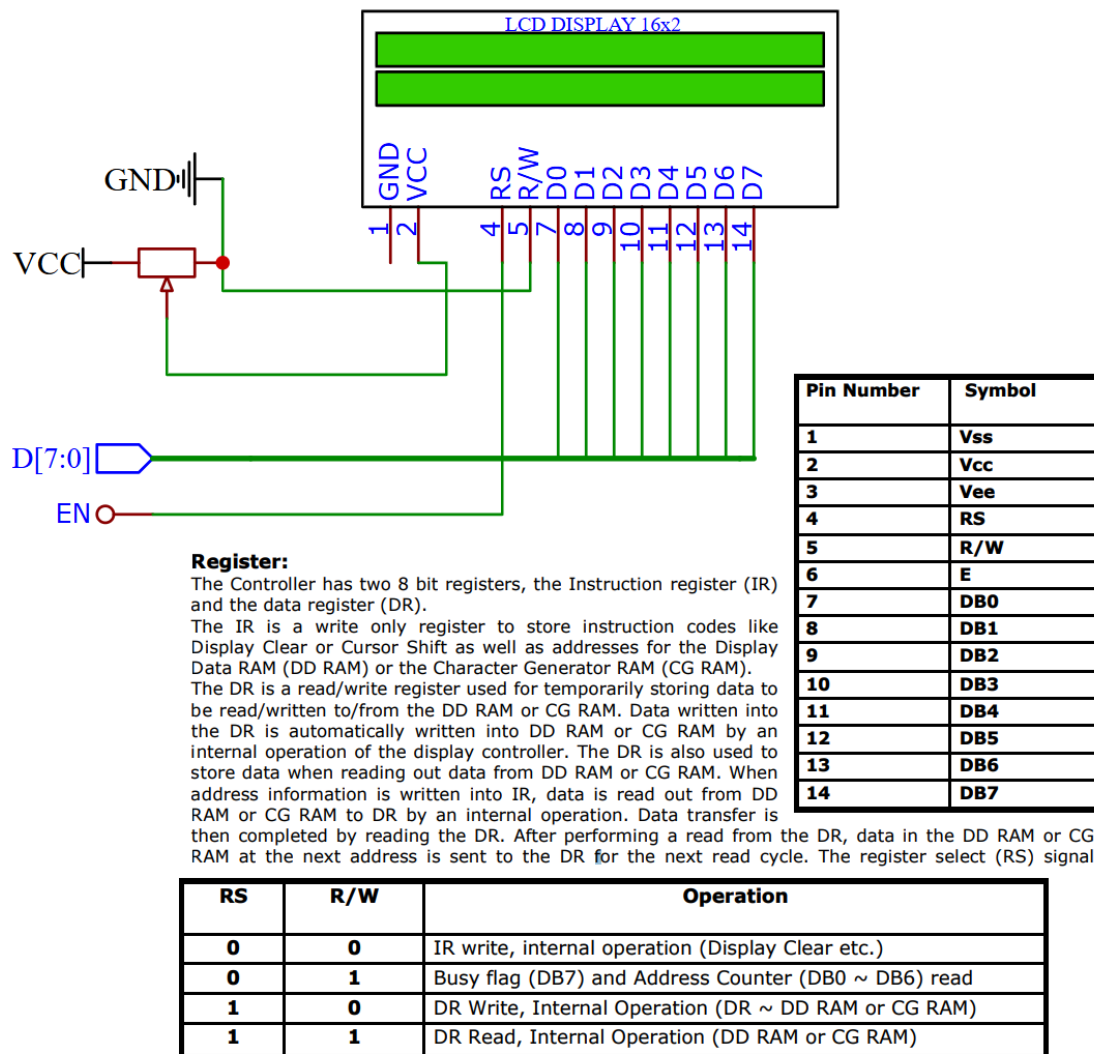
	a	b	c	d	e	f	g	p	HEX		a	b	c	d	e	f	g	p	HEX
	0	0	0	0	0	0	1	1	03H		0	0	0	0	0	0	0	1	01H
	1	0	0	1	1	1	1	1	9FH		0	0	0	0	1	0	0	1	09H
	0	0	1	0	0	1	0	1	25H		0	0	0	1	0	0	0	1	11H
	0	0	0	0	1	1	0	1	0DH		1	1	0	0	0	0	0	1	C1H
	1	0	0	1	1	0	0	1	99H		0	1	1	0	0	0	1	1	63H
	0	1	0	0	1	0	0	1	49H		1	0	0	0	0	1	0	1	85H
	0	1	0	0	0	0	0	1	41H		0	1	1	0	0	0	0	1	61H
	0	0	0	1	1	1	1	1	4FH		0	1	1	1	0	0	0	1	71H

Schema hardware a afișajului se află în anexa cu scheme.

1.7. Modulul LCD

Un modul LCD (Liquid Crystal Display) este un tip de afișaj electronic care utilizează cristale lichide pentru a forma imagini și caractere pe un ecran. Acesta funcționează prin manipularea cristalelor lichide între două straturi de electrozi, care sunt controlate de semnale electrice. Aceste cristale nu emit lumină, ci o modifică în funcție de tensiunea aplicată, permițând trecerea sau blocarea luminii. În general, LCD-urile sunt backlit (au un iluminator din spate), ceea ce le permite să fie vizibile chiar și în condiții de iluminare scăzută. Modulul poate fi configurat pentru a afișa caractere sau imagini, iar complexitatea afișajului depinde de numărul de pixeli sau de linii de caractere pe care le poate manipula. [3] [5] [7]

Unul dintre cele mai comune tipuri de module LCD este cel cu 16 caractere pe două linii (16x2), care poate afișa texte simple și este adesea folosit în aplicații precum afișaje de temperatură, ceasuri digitale sau meniuri de control ale dispozitivelor. Dintre modele și versiuni, am optat pentru LCD-016N002B-CFH-ET, deoarece este același pe care îl folosesc și în micile aplicații de acasă (termometre, măsurare umiditate, tensiune, radiații, etc.)



determines which of these two registers is selected.

2. Descrierea Software.

Subrutinele în limbaj de asamblare

2.1. Subrutinele de programare ale circuitelor 8251 și 8255

2.1.1. Subrutina de programarea a interfeței seriale

```
serial_prog:
    MOV DX, 0AF0H      ; Adresa portului de comenzi pentru serială
    MOV AL, 40H        ; Resetare USART
    OUT DX, AL         ; Trimite comanda de resetare
    MOV AL, 11001100B  ; 8 biți, fără paritate, 1 stop bit, asincron
    OUT DX, AL         ; Configurează modul de lucru
    MOV AL, 00010101B ; Cuvânt de comandă
    OUT DX, AL         ; Trimite comanda de activare
RET
```

2.1.2. Subrutina de programarea a interfeței paralele

```
paralel_prog:
    MOV DX, 0D76      ; Adresa portului de comandă RCC
    MOV AL, 10000000B ; Configurare: Port A – Output, Port B –
                        ; Input, Port C – Split
    OUT DX, AL        ; Trimite configurația RCC
RET
```

2.2. Subrutinele de emisie / recepție caracter pe interfața serială

2.2.1. Subrutina de transmisie a unui caracter pe interfața serială

```
serial_transm:
    MOV DX, 04d2h      ; Portul de date serială (variantă 1)
W_T_S:
    IN AL, DX          ; Citire status din port
    TEST AL, 20h       ; Verificăm dacă bufferul de transmisie e gol
    JZ W_T_S           ; (bit 5)
                        ; Așteaptă până când bitul 5 devine 1
    MOV DX, 04d2h      ; Portul de date
    MOV AL, [CHAR]     ; Caracterul de transmis din memorie
    OUT DX, AL         ; Transmite caracterul
RET
```

2.2.2. Subrutina de recepție a unui caracter pe interfața serială

```
serial_rec:
    MOV DX, 0AF2h      ; Portul de date serială (variantă 1)
W_R_S:
    IN AL, DX          ; Citire status din port
    TEST AL, 10        ; Verificăm dacă datele sunt disponibile (bit
                        ; 4)
    JZ W_R_S           ; Așteaptă până când bitul 4 devine 1
    MOV DX, 0AF2h      ; Portul de date
    IN AL, DX          ; Citește caracterul recepționat
    MOV [CHAR], AL     ; Stocază caracterul în memorie
RET
```

2.3. Subrutina de emisie a unui caracter pe interfața paralelă

```
paralel_emisie:
    MOV DX, 0250h      ; Portul A (variantă 1)
    MOV AL, CL         ; Preluăm caracterul din registrul CL
    OUT DX, AL         ; Transmitem caracterul pe port
RET
```

2.4. Subrutina de scanare a minitastaturii

```
tastatura_scan:
col_1:
    ; Punem 0 pe prima coloană și se verifică dacă
    ; s-au acționat tastele 1, 4, sau 7
    MOV AL, 0840H      ; se pune 0 pe prima coloana (0111 1111 B)
    OUT 0100H, AL      ; selectez ST1 (ieșirea tastaturii)

    IN AL, 08C0H        ; încarc conținutul aflat la portul /ST2
    AND AL, 80H         ; verific prima poziție
    JZ TASTA1           ; salt la tasta 1 dacă rezultatul este 0

    IN AL, 08C0H        ; verific a doua poziție
    AND AL, 40H         ; salt la tasta 4 dacă rezultatul este 0
    JZ TASTA4

    IN AL, 08C0H        ; verific a treia poziție
    AND AL, 20H         ; salt la tasta 7 dacă rezultatul este 0
    JZ TASTA7

col_2:
    ; Punem 0 pe a doua coloană și se verifică
    ; dacă s-au acționat tastele 2, 5, sau 8
    MOV AL, 0840H      ; se pune 0 pe prima coloana (0111 1111 B)
```

```

OUT 0100H, AL      ; selectez ST1 (ieșirea tastaturii)

IN AL, 08C0H       ; încarc conținutul aflat la portul /ST2
AND AL, 80H        ; verific prima poziție
JZ TASTA2          ; salt la tasta 2 dacă rezultatul este 0

IN AL, 08C0H       ; verific a doua poziție
AND AL, 40H        ; salt la tasta 5 dacă rezultatul este 0
JZ TASTA5

IN AL, 08C0H       ; verific a treia poziție
AND AL, 20H        ; salt la tasta 8 dacă rezultatul este 0
JZ TASTA8

col_3:             ; Punem 0 pe a treia coloană și se verifică
                   ; dacă s-au acționat tastele 3, 6, sau 9
MOV AL, 0840H      ; se pune 0 pe prima coloana (0111 1111 B)
OUT 0100H, AL      ; selectez ST1 (ieșirea tastaturii)

IN AL, 08C0H       ; încarc conținutul aflat la portul /ST2
AND AL, 80H        ; verific prima poziție
JZ TASTA3          ; salt la tasta 3 dacă rezultatul este 0

IN AL, 08C0H       ; verific a doua poziție
AND AL, 40H        ; salt la tasta 6 dacă rezultatul este 0
JZ TASTA6

IN AL, 08C0H       ; verific a treia poziție
AND AL, 20H        ; salt la tasta 9 dacă rezultatul este 0
JZ TASTA9

```

2.5. Subrutinele de aprindere / stingere a unui LED

2.5.1. Subrutina de aprindere a unui LED

```

LED_aprinde:
    MOV DX, 0C40H
    MOV AL, FFEH      ; ia valoarea în funcție de led-ul care se
                       ; vrea aprins
    OUT DX, AL
RET

```

2.5.2. Subrutina de stingere a unui LED

```

LED_stinge:
    MOV AL, FFFH      ; stingem totul (punem pe „1”)
    OUT DX, AL
RET

```


2.6. Rutina de afișare a unui caracter hexa pe un rang cu segmente

```
AFISARE_HEX:          ; CL conține caracterul hexazecimal de afișat
                      ; CH conține rangul afișajului (al câtelea
MOV BL, CL             afișaj va fi folosit)
SUB BL, 30H            ; Copiem numărul hexazecimal în BL

MAPARE_HEX:
CMP BL, 0              ; Verificăm dacă este cifra 0
JE TABEL_0
CMP BL, 1              ; Verificăm dacă este cifra 1
JE TABEL_1
CMP BL, 2              ; Verificăm dacă este cifra 2
JE TABEL_2
CMP BL, 3              ; Verificăm dacă este cifra 3
JE TABEL_3
CMP BL, 4              ; Verificăm dacă este cifra 4
JE TABEL_4
CMP BL, 5              ; Verificăm dacă este cifra 5
JE TABEL_5
CMP BL, 6              ; Verificăm dacă este cifra 6
JE TABEL_6
CMP BL, 7              ; Verificăm dacă este cifra 7
JE TABEL_7
CMP BL, 8              ; Verificăm dacă este cifra 8
JE TABEL_8
CMP BL, 9              ; Verificăm dacă este cifra 9
JE TABEL_9
CMP BL, 'A'            ; Verificăm dacă este litera A
JE TABEL_A
CMP BL, 'B'            ; Verificăm dacă este litera B
JE TABEL_B
CMP BL, 'C'            ; Verificăm dacă este litera C
JE TABEL_C
CMP BL, 'D'            ; Verificăm dacă este litera D
JE TABEL_D
CMP BL, 'E'            ; Verificăm dacă este litera E
JE TABEL_E
CMP BL, 'F'            ; Verificăm dacă este litera F
JE TABEL_F

TABEL_0:               ; 0: a, b, c, d, e, f aprinse
MOV AL, 0x3F
RET

TABEL_1:               ; 1: b, c aprinse
MOV AL, 0x06
RET

TABEL_2:               ; 2: a, b, d, e, g aprinse
```

```

        MOV AL, 0xB6
        RET
TABEL_3:                ; 3: a, b, c, d, g aprinse
        MOV AL, 0xB3
        RET
TABEL_4:                ; 4: b, c, f, g aprinse
        MOV AL, 0x66
        RET
TABEL_5:                ; 5: a, c, d, f, g aprinse
        MOV AL, 0xB5
        RET
TABEL_6:                ; 6: a, c, d, e, f, g aprinse
        MOV AL, 0xF5
        RET
TABEL_7:                ; 7: a, b, c aprinse
        MOV AL, 0x07
        RET
TABEL_8:                ; 8: a, b, c, d, e, f, g aprinse
        MOV AL, 0x7F
        RET
TABEL_9:                ; 9: a, b, c, d, f, g aprinse
        MOV AL, 0xB7
        RET
TABEL_A:                ; A: a, b, c, e, f, g aprinse
        MOV AL, 0x77
        RET
TABEL_B:                ; B: c, d, e, f, g aprinse
        MOV AL, 0xF6
        RET
TABEL_C:                ; C: a, d, e, f aprinse
        MOV AL, 0xD1
        RET
TABEL_D:                ; D: b, c, d, e, g aprinse
        MOV AL, 0xF3
        RET
TABEL_E:                ; E: a, d, e, f, g aprinse
        MOV AL, 0xD5
        RET
TABEL_F:                ; F: a, e, f, g aprinse
        MOV AL, 0x75

        ; Selectăm afișajul corespunzător rangului
        ; (CH)
        MOV DX, 0AF0h    ; Portul de date al afișajului cu 7 segmente
        ADD DX, CH        ; Adăugăm rangul pentru a selecta portul
                           ; correct
        OUT DX, AL        ; Tritem codul de segmente la portul de date
        RET

```

Bibliografie

- [1] Mircea Popa, Sisteme cu microprocesoare, Editura Orizonturi Universitare, Timișoara, 2003
- [2] Materialele de curs, laborator și proiect, aferente disciplinei de Proiectare cu Microprocesoare, disponibile pe [Campus Virtual](#)
- [3] Foile de catalog, pentru componente:

Producător	Model / Serie	Sursă, ultima accesare
Intel	8086	inf.pucrs.br , 10 noiembrie 2024
Intel	8284A	alldatasheet.com , 16 noiembrie 2024
Texas Instr.	74LS245	ti.com , 16 noiembrie 2024
Texas Instr.	74LS373	ti.com , 16 noiembrie 2024
Microchip	27C512	jaapsch.net , 17 noiembrie 2024
Cypress	62256	infineon.com , 18 noiembrie 2024
Intel	8251A	alldatasheet.com , 20 noiembrie 2024
Intel	8255A	alldatasheet.com , 20 noiembrie 2024
Texas Instr.	MAX232	ti.com , 20 noiembrie 2024
Texas Instr.	74LS244	ti.com , 1 decembrie 2024
Vishay	LCD-016N002B-CFH-ET	vishay.com , 1 decembrie 2024

- [4] Microprocesorul 8086, lucrare de laborator UTCN, [utcluj.ro](#), ultima accesare: 11 noiembrie 2024
- [5] Principiul de funcționare al ecranului cu cristale lichide, [ro.panasyslcd.com](#), ultima accesare: 1 decembrie 2024
- [6] Silicon Gate MOS 8251, [deramp.com](#), ultima accesare: 20 noiembrie 2024
- [7] Microcontroller and Interfacing Lab, Department of Electrical and Electronic Engineering, BUET, LCD interface with 8086, [itrat.buet.ac.bd](#), ultima accesare: 30 noiembrie 2024
- [8] David Watson, Introduction to MAX232, [theengineeringprojects.com](#), ultima accesare: 20 noiembrie 2024

Cuprins

0. Tema proiectului.....	2
1. Descrierea Hardware.....	3
1.1. Unitatea centrală.....	3
1.1.1. Microprocesorul Intel 8086	3
1.1.2. Generatorul de tact 8284A.....	4
1.1.3. Circuitul 74x245	5
1.1.4. Circuitul 74x373	6
1.1.5. Schema unității centrale a microprocesorului 8086	7
1.2. Conectarea memoriilor.....	8
1.2.1a. Memoria EPROM.....	8
1.2.1b. Circuitul 27C512.....	8
1.2.2a. Memoria SRAM	8
1.2.2b. Circuitul 62256	8
1.2.3. Decodificarea memoriilor	9
1.3. Interfața serială și paralelă.....	11
1.3.1. Conectarea porturilor	11
1.3.2. Interfața serială. Circuitul 8251A	12
1.3.3. Interfața paralelă. Circuitul 8255A.....	14
1.3.4. Circuitul MAX232	15
1.4. Ministastatura cu 9 contacte.....	16
1.4.1. Circuitul 74x244	16
1.4.2. Conectarea ministastaturii cu 9 contacte	17
1.5. LED-urile	18
1.5.1. Conectarea celor 10 LED-uri.....	19
1.6. Modulul de afișaj cu 7 segmente.....	19
1.7. Modulul LCD	21
2. Descrierea Software. Subrutinele în limbaj de asamblare	22
2.1. Subrutinele de programare ale circuitelor 8251 și 8255.....	22
2.1.1. Subrutina de programarea a interfeței seriale.....	22
2.1.2. Subrutina de programarea a interfeței paralele	22
2.2. Subrutinele de emisie / recepție caracter pe interfața serială	22
2.2.1. Subrutina de transmisie a unui caracter pe interfața serială	22
2.2.2. Subrutina de recepție a unui caracter pe interfața serială	23

2.3. Subrutina de emisie a unui caracter pe interfața paralelă	23
2.4. Subrutina de scanare a minitastaturii	23
2.5. Subrutinele de aprindere / stingere a unui LED	24
2.5.1. Subrutina de aprindere a unui LED	24
2.5.2. Subrutina de stingere a unui LED	24
2.6. Rutina de afișare a unui caracter hexa pe un rang cu segmente	25
3. Bibliografie	27