

Lucrarea 8

Interfața paralelă de intrări/ ieșiri PIO 8255A

8.1. Introducere

Circuitul PIO (Parallel Input/Output) 8255 se mai numește și Interfață programabilă pentru periferice (Programmable Peripheral Interface (PPI)) sau uneori mai apare sub denumirea de „cip de port paralel”, aceasta deoarece se folosește în general pentru rezolvarea problemelor de interfațare a dispozitivelor care realizează transferul paralel al datelor. Circuitul 8255A (sau vom spune simplu, 8255) a fost proiectat inițial pentru 8085, dar ulterior a fost adoptat și în sisteme cu 8086 sau succesorii lui.

Circuitul are 40 pini, iar în interior, așa cum se poate observa și din Figura 8.1, are 3 porturi de câte 8 biți fiecare, denumite: port A (PA), port B (PB) și port C (PC). Portul C poate fi folosit 1) divizat în 2: parte Low și parte High, fiecare parte putând fi configurată ca intrări (In) sau ieșiri (Out), deci rezultă 2 porturi de 4 biți sau poate fi folosit 2) divizat în 8: fiecare pin poate fi configurat individual ca intrare (In) sau ieșire (Out) => 8 porturi de câte 1 bit. Porturile PA, PB, PC pot fi programate individual ca porturi de intrare sau ca porturi de ieșire, prin intermediul celor 2 blocuri de control: *Control GRUP A* și *Control GRUP B*. Busul de date este pe 8 biți (D7 – D0) și se folosește pentru a transfera date sau informație de control sau stare între CPU și porturile I/O. Logica de control este necesară pentru selecția circuitului, comanda operațiilor de citire sau scriere și programarea circuitului.

Procesorul controlează interfața 8255 printr-un **Registru de Control (RC)** pe 8 biți, care nu poate fi citit, doar înscris. În funcție de A1A0 se va specifica portul la care este asigurat accesul: dacă A1A0 = 00 -> acces la **PA**, dacă A1A0 = 01 -> acces la **PB**, dacă A1A0 = 10 -> acces la **PC**, dacă A1A0 = 11 -> acces la **RC**; altfel spus, folosind adresa de bază se accesează fiecare registru intern al interfeței.

Pinii /RD și /WR sunt conectați la pinii /RD, /WR ai microprocesorului 8086, fiind activați atunci când au loc operații de citire (deci folosind instrucțiunea IN), resp. de scriere (deci folosind instrucțiunea OUT). Un semnal High aplicat pe RESET va configura toate cele 24 linii ca fiind intrări, dar ulterior, liniile pot fi reconfigurate prin înscriserea unui Cuvânt de Comandă adecvat.

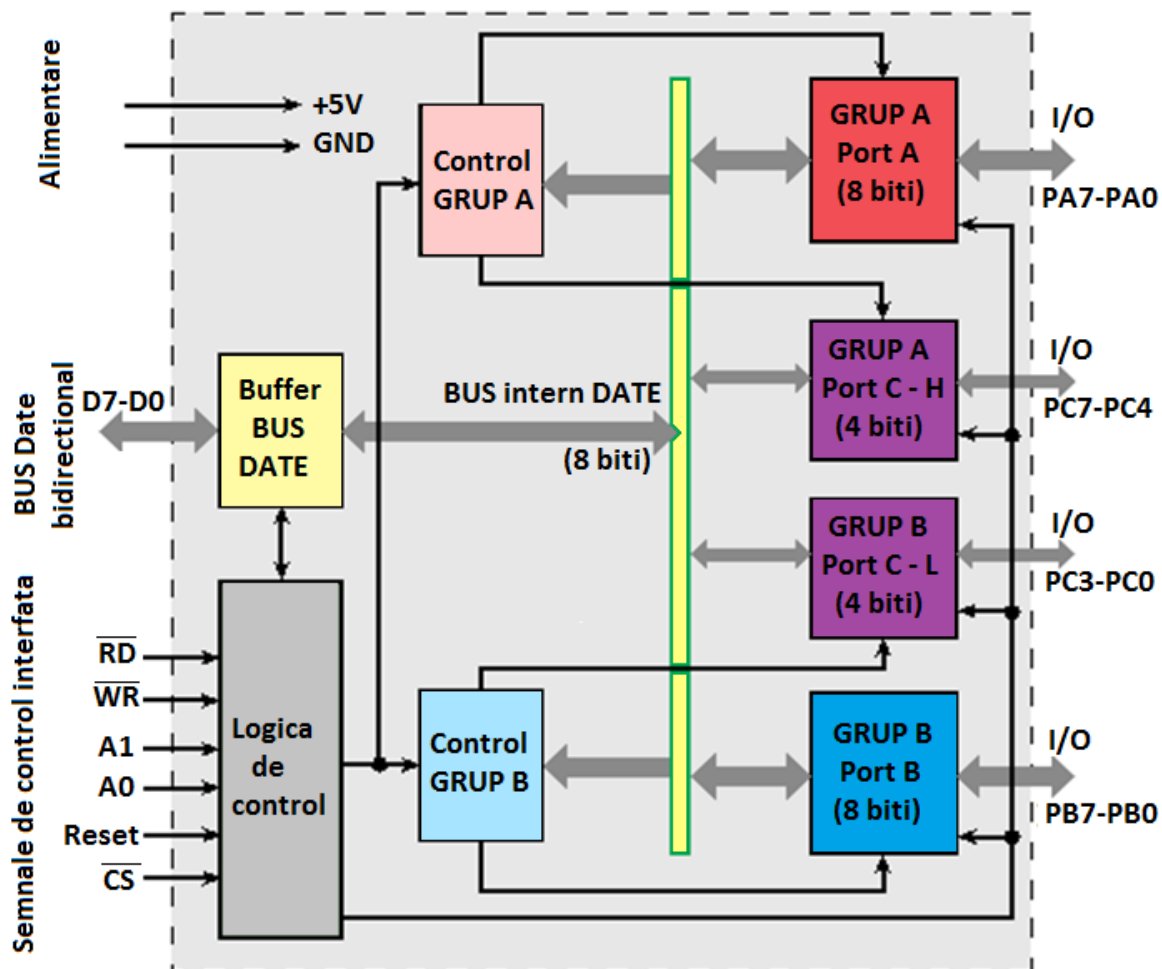


Figura 8.1 Schema bloc internă a circuitului 8255

8.2. Conectarea 8255 în sisteme

Pe măsură ce dispozitivele de intrare/ ieșire au devenit tot mai performente, deținând tot mai multe caracteristici, folosirea unei interfețe de tip PIO 8255 pentru conectarea acestor dispozitive la procesor a devenit o necesitate, în special atunci când se dorea conectarea de multiple dispozitive la CPU. Un exemplu schematizat de sistem în care s-au conectat 2 dispozitive de ieșire (o tastatură și un sistem de afișaj bazat pe LED-uri) și unul de intrare (o tastatură) la PIO este prezentat în Figura 8.2. Dispozitivele vor transmite/ recepționa datele în grupuri de câte 8 biți, nici mai mult, nici mai puțin, PIO 8255 fiind conectată cu procesorul 8086 pentru transferul datelor prin intermediul busului de date D7...D0. Pentru aceasta, va fi necesară programarea circuitului 8255 astfel: PA programat ca port de intrare, iar PB și PC programate ca porturi de ieșire.

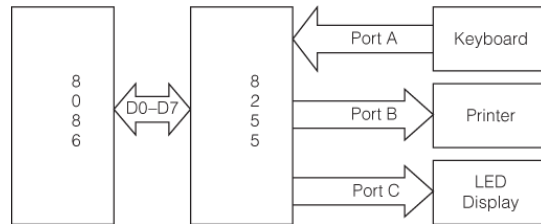


Figura 8.2 | Conectarea la procesorul 8086 a circuitului PIO 8255 la care sunt legate 3 periferice: unul de intrare și 2 de ieșire

Modul în care se va comporta circuitul 8255 va fi decis de octetul care se va înscrie în **Registrul de Control (RC)**, aflat la Adresa de bază +3. Circuitul 8255 este văzut de către microprocesor asemănător TIMER-ului 8253, ca un port de I/O sau mai bine spus la 4 adrese de port diferite: AB, AB+1, AB+2 și AB+3 (dacă A1A0 ai PIO ar fi conectați la A1A0 ai Busului de Adresă (BA) al sistemului). Activarea circuitului pentru a fi funcțional se realizează prin activarea unui pin numit /CS (activ pe 0), iar în general acest pin este activat printr-o ieșire a unui circuit numit „decodificator de adresă”. Procesul prin care se află combinația de biți de adresă care activează acest semnal /CS sau regiștrii interni ai circuitului se numește „decodificare de adresă”. Astfel, în funcție de circuitul care se conectează în sistem, în procesul de decodificare al adresei pot fi implicați doar c.m.p.s. 8 biți sau c.m.p.s. 16 biți ai adresei (nici un port de I/O nu poate avea o adresă care să se scrie cu mai mult de 16 biți). Așa cum se vede în Figura 8.1, circuitul 8255 are (asemănător TIMER-ului 8253) 2 pini de adresă A1 și A0 prin intermediul cărora (4 combinații, așa cum arată Tabelul 8.1) se vor accesa regiștrii (în acest caz numiți „porturi” PA, PB și PC, resp. RC) lui interni. În Figura 8.3 este prezentată o modalitate de conectare a circuitului 8255 la procesor, în ceea ce privește selecția circuitului: observați că liniile A1A0 ale PIO au fost legate la liniile A2A1 ale BA. Semnalul de pe pinul /CS se va activa și astfel va activa circuitul PIO atunci când ieșirea porții NAND va furniza 0 la ieșire, sau așa cum se observă din Tabelul 8.2 atunci când pe BA apare combinația de biți de adresă 1100 0xx0b, deci adresele C0h (pentru PA), C2h (pentru PB), C4h (pentru PC), C6h (pentru RC). Semnalele /RD și /WR ale 8255 sunt conectate la cele corespunzătoare ale sistemului, gestionate de procesor: /IORD și resp. /IOWR.

Tabel 8.1 | Selecția regiștrilor interni la PIO 8255

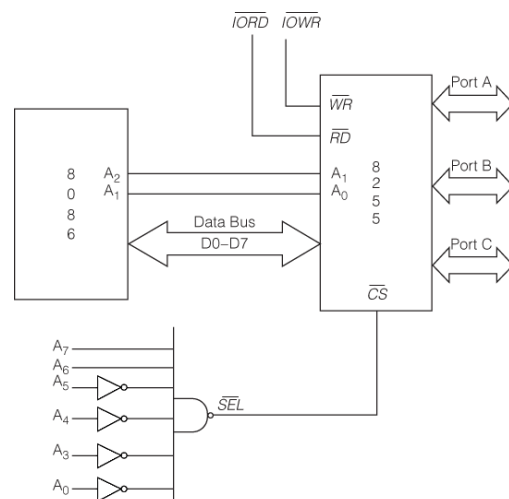
/CS	A1	A0	Regiștrul Selectat
0	0	0	PA
0	0	1	PB
0	1	0	PC
0	1	1	RC

În PC-XT: 60h(PA), 61h(PB), 62h(PC), 63h(RC)

Tabel 8.2 | Decodificarea adresei din Figura 8.3

/CS	A7	A6	A5	A4	A3	A2	A1	A0	Adresa rezultată	Regiștrul Selectat
						0	0		C0h	PA
						0	1		C2h	PB
0	1	1	0	0	0	1	0	0	C4h	PC
						1	1		C6h	RC

Figura 8.3 Posibilă conectare a 8255 la 8086 și activarea semnalului /CS: decodificarea adresei folosind porți logice
 $A7A6A5A4A3A2A1A0=1100\ 0A2A10$
C0h, C2h, C4h, C6h



În continuare sunt prezentate alte 3 posibilități de implementare a circuitului decodificator de adresă, folosind:

1. **Circuitul demultiplexor 74LS138**
2. **Circuit de memorie EPROM 2816**
3. **Comparator 74LS682**

Alte metode mai noi de implementare a decodificatoarelor de adresă pot folosi circuite programabile precum FPGA-uri.

Funcționarea schemei din Figura 8.4 (a se vedea Anexa pentru funcționarea circuitului demultiplexor 74LS138): în funcție de combinația care apare pe pini CBA ai circuitului 74LS138, deci în funcție de biții de adresă A15..A13, se va selecta una din ieșirile 0..7 pe care se va trimite semnal 0 dacă G1=1, iar G2A și G2B sunt 0 (astfel, A19A18A17=111 și A16=1). De exemplu, la fiecare din ieșirile 0..7 se pot conecta semnale de tip /CS ale unor circuite de memorie așa cum se observă în figură, acestea devenind active sau selectate doar atunci când combinația respectivă de 3 biți apare la intrările CBA ale circuitului 74LS138 (aici, un singur circuit de memorie e selectat la un moment dat).

Funcționarea schemei din Figura 8.5: în funcție de ceea ce este înscris în circuitul de memorie la o anumită adresă; de exemplu, să presupunem că la adresa 5Ch – apărută la circuitul de memorie, pe biții A7...A0, cu jumperul din stânga pus pe poziție 01- este înscrisă valoarea Bh adică 1011b; acest lucru va însemna că atunci când se selectează adresa 5Ch, pe biții D3...D0 apare combinația 1011, selectând în acest fel circuitul al cărui /CS este legat la pinul D2.

Funcționarea schemei din Figura 8.6 (a se vedea Anexa pentru funcționarea circuitului comparator 74LS682): combinația de 8 biți care apare pe pini P7...P0 ai circuitului 74LS682 se compară cu starea switch-urilor Q7...Q0 și dacă sunt egale, se scoate 0 la pinul de ieșire P al circuitului, selectând astfel circuitul conectat acolo; altfel, circuitul nu va fi selectat, deci nu va funcționa.

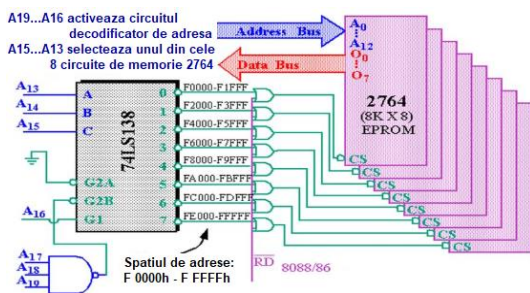


Figura 8.4. Decodificator de adresă pe bază de circuit demultiplexor 74LS138

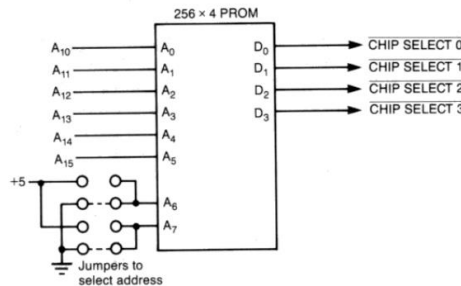


Figura 8.5. Decodificator de adresă pe bază de circuit de memorie

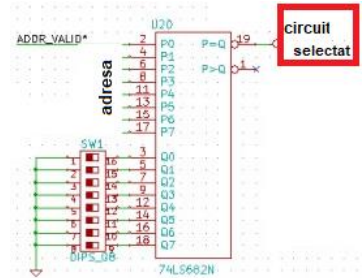


Figura 8.6. Decodificator de adresă pe bază de circuit comparator 74LS682

8.3. Modulile de lucru ale interfeței 8255

În general, la un circuit 8255 se pot conecta:

- **3 periferice:** câte unul la fiecare port de 8 biți, astfel că fiecare din **cele 3 porturi** va putea funcționa ca un port de I/O pe 8 biți; cei 24 pini vor fi grupați câte 8: PA (8 biți), PB (8 biți), PC (8 biți);
- **2 periferice:** câte unul la fiecare **grup de 12 biți:** astfel, dacă se folosește **portul C** divizat în **PC_H** și **PC_L**, atunci aceste porturi pe 4 biți pot fi folosite împreună cu **portul PA** și **portul PB** în vederea gestionării semnalelor de control/ stare; cele 2 grupuri nou formate (PA și PC_H, resp. PB și PC_L) se vor numi **Grup A** și respectiv **Grup B**;
- **4 periferice:** câte unul la fiecare port: (dacă se folosește **portul C** divizat în **PC_H** și **PC_L**) atunci aceste porturi pe 4 biți pot controla 2 dispozitive, iar folosind **portul PA** și **portul PB** se pot controla alte 2 dispozitive (cele 4 porturi pot funcționa independent unele de altele)

Oricare ar fi perifericele dorite a fi legate la pini circuitului 8255, acesta va trebui programat pentru funcționare; astfel, în RC se va înscrie un „cuvânt de comandă” pe 8 biți care va avea bitul 7 diferit, în funcție de categoria de mod de lucru dorită.

8255 poate funcționa în 2 moduri principale: BSR și I/O

- 1) **Modul BSR (Bit Set/Reset)**
- 2) **Modul I/O divizat în:**
 - Modul I/O simple (mod de lucru 0) - intrări/ieșiri simple**
 - Modul I/O cu handshake (mod de lucru 1) - intrări/ieșiri strobate,**
 - Modul I/O bidirecțional (mod de lucru 2) - intrări/ieșiri bidirecționale**

1) b7=0 pentru selectarea **modului BSR (bit set/reset) – doar pentru portul C**

		b7	b6	b5	b4	b3	b2	b1	b0
Grup C	PC0	0	x	x	x	0	0	0	1 – bitul PCx va fi setat 0 – bitul PCx va fi resetat
	PC1	0	x	x	x	0	0	1	
	PC2	0	x	x	x	0	1	0	
	PC3	0	x	x	x	0	1	1	
	PC4	0	x	x	x	1	0	0	
	PC5	0	x	x	x	1	0	1	
	PC6	0	x	x	x	1	1	0	
	PC7	0	x	x	x	1	1	1	

2) b7=1 pentru selectarea **modului I/O**

		Flag selectie mod	Selectie mod Grup A		Port A	Port C (H)	Selectie mod Grup B	Port B	Port C (L)
		b7	b6	b5	b4	b3	b2	b1	b0
Grup A	PA	1	00-mod 0 01-mod 1 1x-mod 2		0-Out 1-In				
	PC (H)	1				0-Out 1-In			
Grup B	PB	1					0 – mod 0 1 – mod 1	0-Out 1-In	
	PC (L)	1							0-Out 1-In

Mod 0: Intrări/ieșiri simple sau de bază (Basic or simple Input/Output)

Port PA, Port PB, Port PC (PCH, PCL) pot fi programate/configurate independent ca intrări sau ieșiri (I/O)
=> toti cei 8 biți la fel pt PA și PB, resp. toți cei 4 biți la fel pentru PCL și PCH

Modul 0		
PORT A	IN	OUT
PORT B	IN	OUT
PORT C	IN	OUT

Mod 1: Intrări/ieșiri strobdate sau cu handshake (Strobed/with handshake Input/Output)

Port PA și Port PB (8 biti) pot fi configurate independent ca porturi de I/O, fiind strobdate cu semnale de control - pentru handshaking

- dispun și de cerere de întrerupere (se folosesc câte 3 biți de la Portul PC)

Semnalele de la Portul PC sunt rezervate pentru semnale de control

=> Grupul GA format din PA și PCH și Grupul GB format din PB și PCL

Configurarea pentru handshaking:

- Două porturi (PA și PB) funcționează ca porturi de 8 biti de I/O (configurabile).
- Fiecare port folosește 3 linii din portul PC ca semnale de handshake pt control port si INTR (biții 2-0 pt PB și 5-3 pt PA la intrari, resp 2-0 pt PB și 7-6,3 pt PA la iesiri).
- Celelalte 2 linii din Port PC pot fi folosite pentru functii simple de I/O (7-6 la intrari, resp 5-4 la iesiri).

Modul 1		
PORT A	IN	OUT
PORT B	IN	OUT
PORT C		
7-6	I/O	Ctrl PA
5-4	Ctrl PA	I/O
3	INTR _A	INTR _A
2-1	Ctrl PB	Ctrl PB
0	INTR _B	INTR _B

Mod 2: Bus bidirecțional - Bidirectional Bus - rar utilizat

Doar Port PA poate fi configurat ca bus bidirecțional, portul PC (5 linii: PC7-PC3) furnizând semnalele de control (PC2-PC0 pot fi folosite doar pentru funcții I/O simple), iar portul PB poate lucra în mod 0 sau mod 1. Când portul PB este programat în modul 1, liniile PC2-PC0 ale Portului PC sunt folosite ca semnale handshaking pt controlul PB.

Acest mod permite transferul de date bidirecțional (transmisie și recepție) pe o singură magistrală de date de 8 biți (D7-D0), folosind semnale handshaking.

Modul 2	
PORT A	IN/OUT
PORT B	-
PORT C	
7-6	Ctrl - Out
5-4	Ctrl - In
3	INTR
2-1	I/O
0	I/O

8.4. Programarea interfeței 8255

Exemplul1: Să se scrie o secvență de instrucțiuni care să programeze următoarea configurare pentru circuitul 8255 având adresele date în Tabelul 8.2: 8255 va funcționa în modul 0, cu PA și PC_L ca ieșiri, iar PB și PC_H ca intrări.

Rezolvare: modul 0 se poate configura dacă b7=1, fiind un mod I/O, iar restul biților vor fi programați astfel:

b6 b5 = 00 și b2 = 0 – pentru a specifica modul 0 pentru Grupul A și Grupul B,
 b4 = 0 pentru că se dorește PA ca port de ieșire (Out) -> afisaj LCD, imprimanta
 b3 = 1 pentru că se dorește PC_H ca port de intrare (In) <- comutatoare, convertor
 b1 = 1 pentru că se dorește PB ca port de intrare
 b0 = 0 pentru că se dorește PC_L ca port de ieșire

Astfel, vom avea cuvântul de comandă: 10001010 b = 8Ah, care va trebui înscris la adresa corespunzătoare RC:

mov AL, 8Ah

out C6h, AL ; C0h (PA); C2h (PB); C4h (PC);C6h (RC)

Exemplul2: Se presupune că pe o plăcuță de circuit PCB există următoarea configurare pentru 8255: 12 switch-uri sunt conectate la PB și PC_H, iar la PA și PC_L sunt conectate 12 LED-uri. Să se scrie o secvență de program care să citească starea switch-urilor și să o redea înspre LED-uri. Se vor folosi aceleași adrese pentru PIO ca în Exemplul 1.

Rezolvare:

Programarea interfeței se va realiza așa ca în exemplul 1, întrucât

PB și PC_H trebuie să fie intrări, iar PA și PC_L trebuie să fie ieșiri.

Atunci când un switch este în starea închis, se va citi semnal "0" pe pinul respectiv din portul de intrare, iar când e în starea deschis, se va citi un semnal în "1". Pe pinii unde sunt conectate cele 12 LED-uri, există și rezistențele asociate pentru limitare; LED-urile vor străluci (vor fi aprinse) atunci când există semnal în "1" logic pe pinul respectiv din portul de ieșire.

Secvența de program pentru funcționarea circuitului este:

.data

PA EQU 0C0h

PB EQU 0C2h

PC EQU 0C4h

RC EQU 0C6h

.code

mov AL, 8Ah ; cuvântul de comandă pentru configurare 8255

out RC, AL ; înscrierea RC

in AL, PB ; citire PB

out PA, AL ; redarea conținutului lui PB la PA

in AL, PC ; citire PC

mov CL, 4

ror AL, CL ; obținerea conț. PC_H pe poziția coresp. PC_L

out PC, AL ; redarea conținutului lui PC_H la PC_L

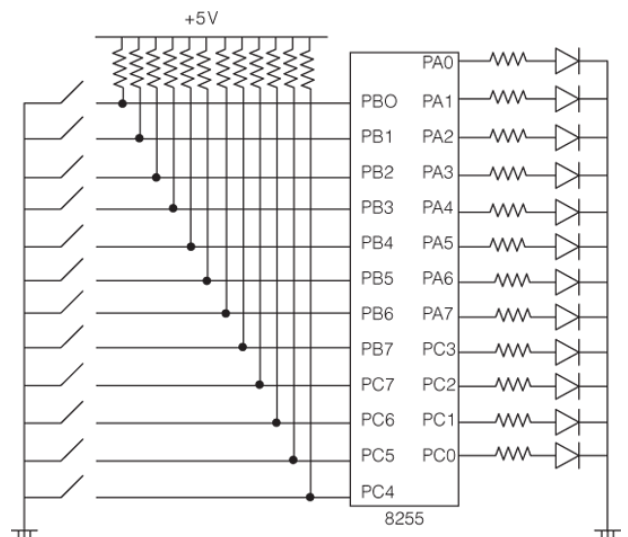


Figura 8.7. Conectarea switch-urilor și a LED-urilor la 8255 pentru Exemplul 2

8.5. Utilizarea circuitului 8255 în PC-XT și în PC-AT

Legătura din TIMER-ul 8253 și PIO 8255 în PC-XT (a se vedea Figura 3 din Lucrarea 6 cu TIMER-ul 8253/54):

PB0 al PIO 8255 (adică pinul 0 al PB al circuitului PIO, având adresa de port 61h) e conectat ca semnal Gate 2 al TIMER-ului 8253

PB1 al PIO 8255 (adică pinul 1 al PB al circuitului PIO, având adresa de port 61h) e conectat ca semnal de intrare în poarta AND ce activează (împreună cu ieșirea Canalului 2 a TIMER-ului 8253) difuzorul de pe placa de bază

Ieșirea canalului 2 a TIMER-ului 8253 (pe lângă faptul că intră în poarta AND să activeze difuzorul) mai este dată ca intrare în PC5 a PIO 8255 (adică pe pinul 5 al portului PC, aflat la adresa de port 62h).

La următoarea serie de sisteme, începând cu **PC-AT** (deci de la sisteme cu 286, 386 sau altele mai noi, inclusiv Pentium) s-a implementat în mod suplimentar o posibilitate de a crea întârzieri fixe cu ajutorul unui bit din PB4 care își schimbă valoarea (comută când în 0, când în 1) la fiecare 15,085 microsecunde.

8.6. Utilizarea unei plăci PCB externe, conectată pe bus ISA

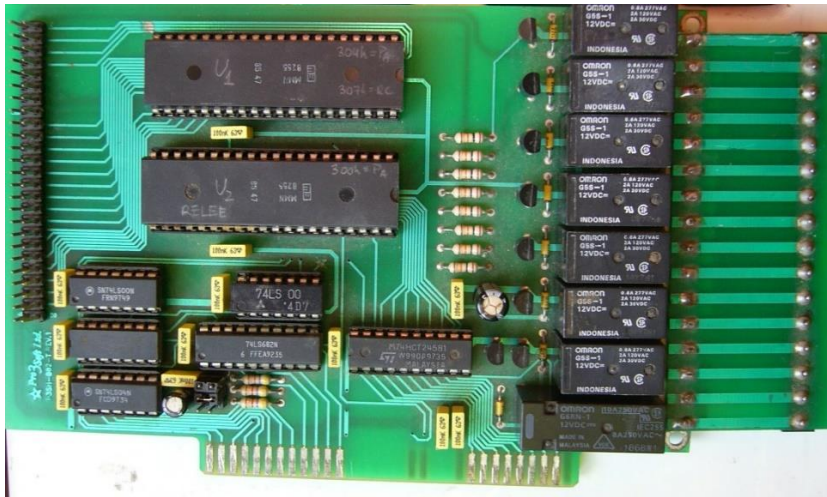


Figura 8.8. Placă PCB cu 2 circuite PIO 8255 (notate U1 și U2)

Există 2 circuite 8255 pe placă, notate U1 și U2:
 U2 – la care portul PA este folosit pentru a conduce cele 8 relee din partea dreaptă a plăcii (Figura 8.8), iar U1 poate fi folosit pentru a lega alte dispozitive, prin intermediul pinilor puși la dispoziție prin conectorul cu 50 contacte din partea stângă a plăcii.

Pe plăcuță, folosind jumperii JP1, JP2, JP3 (fizic, aceștia se află în partea de jos a plăcii, deasupra primelor contacte de conectare a plăcii pe busul ISA, în Figura 8.8) plasați pe o anumită poziție, se vor decodifica circuitele PIO la o adresă sau alta (pe schema electrică din Figura 8.9, jumperii se află la intrările Q0Q1Q2 ale circuitului comparator 74LS682): A1A0 ai circuitelor sunt legate la A1A0 de pe busul de adresă. Pinii de adresă A8, A9, A10 sunt conectați în această ordine la JP1, JP2, JP3, iar ceilalți biți de adresă (A11, A7...A3) se observă din Figura 8.9 că sunt 0 pentru ca semnalul de la ieșirea porții NAND să fie 0. A2 este folosit pentru a selecta /CS1 sau /CS2.

De exemplu, folosind jumperii JP1 și JP2 va rezulta Q0Q1=11, Q2=0 și atunci când A10A9A8=011 (deci B_P=B_Q), se va selecta ieșirea circuitului comparator (P=Q), astfel:

dacă A2=0 => semnalul /CS2 va fi activ și va selecta circuitul U1
dacă A2=1 => semnalul /CS1 va fi activ și va selecta circuitul U2

Observație:

La poziționarea jumperilor pe placă, se recomandă utilizarea unor adrese mai mari decât 200h pentru a evita posibilele conflicte între resursele sistemului.

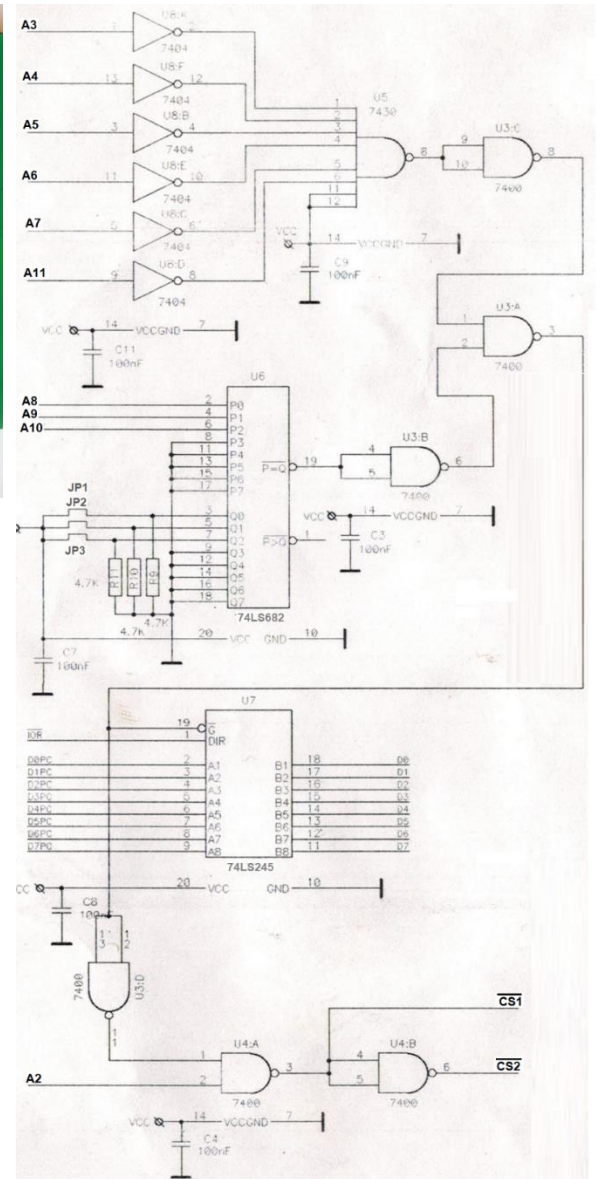


Figura 8.9. Placă cu PIO 8255 – schema decodificatorului de adresă

A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Adresele
0	JP3	JP2	JP1	0	0	0	0	0	/CSx	A1A0		
0	0	1	1	0	0	0	0	0	0	x	x	A2=0 => /CS2 activ => circ.U1 300h – 303h
0	0	1	1	0	0	0	0	0	1	x	x	A2=1 => /CS1 activ => circ.U2 304h – 307h

Anexa :

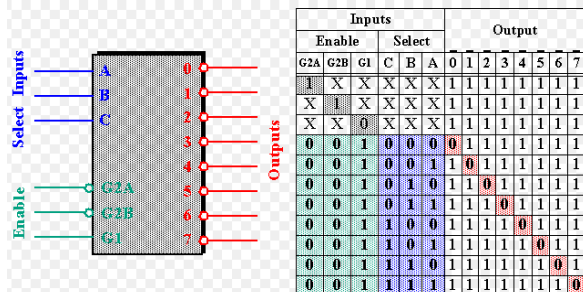


Figura A1. Funcționarea circuitului demultiplexor 74LS138

FUNCTION TABLE

INPUTS		OUTPUTS	
DATA	ENABLES	P = Q	P > Q
P, Q	G, GT, G2		
P = Q	L, L, L	L	H
P > Q	L, L, L	H	L
P < Q	L, L, L	H	H
X	H, H, H	H	H

H = HIGH Level, L = LOW Level, X = Irrelevant

Figura A2. Funcționarea circuitului comparator de 8 biți 74LS682

1. Care este rolul utilizării circuitului 8255 în aplicații cu 8086?

R: PIO8255 se folosește în general pentru rezolvarea problemelor de interfațare a dispozitivelor care realizează transferul paralel al datelor.

2. A/F Interfața Programabilă pt. Periferice (PPI), Intel 8255, este un dispozitiv I/O programabil de uz general care oferă 24 de pini GPIO.

R: Da, acesta permite conectarea la microsistem a perifericelor care utilizează transferul paralel al datelor.

3. Circuitul 8255 se folosește în general pentru rezolvarea problemelor de interfațare a dispozitivelor care realizează transferul (paralel/serial) al datelor.

R: paralel

4. A/F Poate fi folosit 8255 pentru transmisia serială a datelor?

R: În caz de necesitate da, modificând periodic, prin program, nivelul logic la un pin al unui port programat să fie de sens ieșire. Dar în familia de periferice programabile Intel există alte circuite specializate pentru comunicații serial (de ex. UART 8250).

5. (A/F) Circuitul PIO are 4 porturi denumite PA, PB, PC, PD.

R: F, are doar 3 porturi: PA, PB și PC

6. (A/F) Grupul A conține portul PA și portul PC_L.

R: Fals, grupul GA conține PA și PC_H

7. (A/F) Grupul B conține portul PB și portul PC_L.

R: A

8. (A/F) Portul C poate fi folosit ca 2 porturi separate de câte 4 biți, fiecare putând fi configurat individual ca intrare (In) sau ieșire (Out).

R: A

9. (A/F) Portul C poate fi folosit ca 8 porturi separate de câte 1 bit, fiecare putând fi configurat individual ca intrare (In) sau ieșire (Out).

R: A

10. (A/F) Cele 24 linii PIO pot fi configurate ca intrări/ ieșiri prin înscriserea unui Cuvânt de Comandă adecvat, pe 16 biți.

R: Fals, cuv de cdă se scrie pe doar 8 biți

11. A/F Fiecare grup de 12 pini I/O poate fi programat să funcționeze ca pini de intrare sau ieșire.

R: Da, vezi campurile de biți al cuvântului de control pentru Modul ZERO de lucru ("Basic I/O")

12. A/F Un impuls de "1" logic aplicat pe pinul RESET va configura toți cei 3x8 pini GPIO de sens "Intrare" (de înaltă impedanță)

R: Da, adevărat. Din motive de protecție a ieșirilor lui 8255 în situații hazard care pot apărea la pornirea sistemului.

13. (A/ F) Porturile PA, PB, PC pot fi programate individual ca porturi de intrare sau ca porturi de ieșire, prin intermediul celor 2 blocuri de control: Control GRUP A și Control GRUP B.

R: A

14. Conectarea lui 8255 în sisteme cu microprocesor folosește o magistrală de date de 8 biți, un număr de a) 1 b) 2 c) 3 d) 4 pini de adresă, semnalele "Chip Select", RD și WR. Alegeți un răspuns.

R: 2 pini de adresă, numiți A1 și A0

15. (A/F) În registrul de comandă, se programează b7=0 pentru selectarea modului BSR (bit set/reset) care se referă la configurarea individuală a biților portului A.

R: Fals, se programează b7=0 pentru selectarea modului BSR (bit set/reset) care se referă la configurarea individuală a biților unui port al PIO, dar nu PA, ci PC.

16. (A/F) În registrul de comandă, se programează b7=1 pentru selectarea modului I/O, care cuprinde alte 4 submoduri.

R: Fals, în registrul de comandă, se programează b7=1 pentru selectarea modului I/O, dar acest mod cuprinde alte 3 submoduri, nu 4.

17. A/F În Modul 0 al interfeței, portul C transferă date între microprocesor și dispozitive periferice, dar în modul 1 sau 2 el generează sau acceptă semnale de dialog ("handshaking") cu dispozitivul periferic.

R: (Adev) Da, portul C în aceste două moduri oferă posibilitatea de a folosi sistemul de intreruperi ale microprocesorului, degrevându-l de sarcina supravegherii permanente a stării comunicației, ea putând fi citită prin biții portului C.

18. Există un mod special de lucru, care permite microprocesorului să modifice cu o singură instrucțiune de tipul out DX, AL starea logică a oricărui pin al portului pentru care s-a programat acest mod. Portul poate fi: a) numai PA b) numai PB c) numai PC d) oricare e) doar PA și PB

R: Adev, Numai pt PC. Dacă bitul MSB al cuvântului de mod se înscrie "0", acel cuvânt devine o comandă de tip "bit Set/Reset"(modul BSR) pt. 8255, acționând asupra unui singur bit al portului C.

19. A/F În modul 0, octetul înscris într-un port definit de sens ieșire poate fi citit înapoi de la aceeași adresă I/O.

R: Adevărat.

20. Din moment ce interfața are 4 registre interne, disponibilă la 4 adrese de port diferite, cum se specifică registrul la care dorim acces?

R: În funcție de A1A0 se va specifica portul la care este asigurat accesul: dacă A1A0 = 00 -> acces la PA, dacă A1A0 = 01 -> acces la PB, dacă A1A0 = 10 -> acces la PC, dacă A1A0 = 11 -> acces la RC;

21. Din p.d.d.v. al adreselor de port, circuitul 8255 seamănă cu circuitul: a) 8250, b) 8253, c) 8237, d) 8259.

R: b)

22. Dacă pentru circuitul PIO 8255 conectat într-un sistem la adresele C0h, C2h, C4h și C6h s-a scris următoarea secvență de instrucțiuni: mov AL, 8Ah

out C6h, AL

ce s-a programat?

a) Modul de lucru, cu 8255 funcționând în modul 0, cu PA și PC_L ca ieșiri, iar PB și PC_H ca intrări;

b) Modul de lucru, cu 8255 funcționând în modul 0, cu PB și PC_L ca ieșiri, iar PA și PC_H ca intrări;

c) s-a scos pe PA valoarea 8Ah;

d) s-a scos pe PC valoarea 8Ah.

R: A), la ultima adresa este RC al PIO

23. (A/F) un port de I/O (nemapat în memorie) poate avea o adresă care să se scrie cu mai mult de 16 biți.

R: Fals, nr max de biti pt specificarea adresei unui periferic este de 16 biți, doar pt specificarea adresei de memorie poate fi mai mare (20 biți în PC-XT, resp 24 biți în PC-AT)

24. Dacă într-un sistem s-au conectat la PIO astfel: o tastatură pe PA, un sistem de afișaj bazat pe LED-uri pe PB și o imprimantă pe PC, cum trebuie configurat circuitul PIO?

R: programarea circuitului 8255 se va face astfel: PA programat ca port de intrare, iar PB și PC programate ca porturi de ieșire.

25. La un PIO conectat la 8086 se pot conecta 4 dispozitive? Cum ?

R: DA, dacă 2 dintre ele au nevoie de maxim 8 biți, iar alte două au nevoie de maxim 4 biți pt funcționare

26. Care dintre următoarele circuite are regiștrii interni referiți ca "porturi" (și sunt 3 la număr)? a) 8253, b) 8237, c) 8255, d) 8259, d) 8250

R: c)

27. (A/F) în PC-XT, Pinul 0 al PB al circuitului PIO, având adresa de port 61h, e conectat ca semnal Gate 2 al TIMER-ului 8253.

R: A

28. (A/F) adresa de bază a circuitului PIO inserat în sisteme PC-XT cu 8086, a fost aleasă 61h

R: fals, adresa de bază este 60h -> ultimii 2 biți trebuie neapărat să fie 00 pentru o decodificare corectă

29. Dacă într-un sistem, pinii de adresă ai circuitului PIO sunt conectați la BA pe pinii A2A1, care vor fi adresele celor 4 regiștrii interni ai PIO, în funcție de adresa de bază (notată AB)?

R: vor sări din 2 în 2, la AB va fi Pam la AB+2 va fi PB, la AB+4 va fi PC, iar la AB+6 va fi RC

30. (A/F) La PIO este posibilă conectarea unui dispozitiv care are nevoie de 12 biți de date (de ex un convertor pe 12 biți)

R: A, dar atunci se va folosi tot grupul A sau tot grupul B de către acel dispozitiv

31. în PC-XT, ieșirea canalului ____ a TIMER-ului 8253 este dată ca intrare în PC5 a PIO 8255 (adică pe pinul 5 al portului PC, aflat la adresa de port 62h).

R: ch 2

7. în PC-AT s-a implementat o posibilitate de a crea întârzieri fixe cu ajutorul bitului _____ care își schimbă valoarea (comută când în 0, când în 1) la fiecare 15,085 microsecunde.

R: PB4