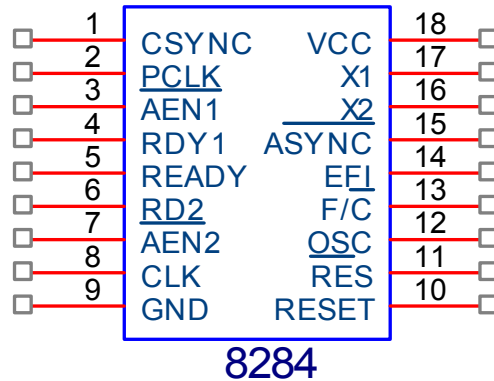


CHƯƠNG 4: TỔ CHỨC NHẬP / XUẤT

1. Các mạch phụ trợ 8284 và 8288

1.1. Mạch tạo xung nhịp 8284

Mạch tạo xung nhịp dùng để cung cấp xung nhịp cho μP .

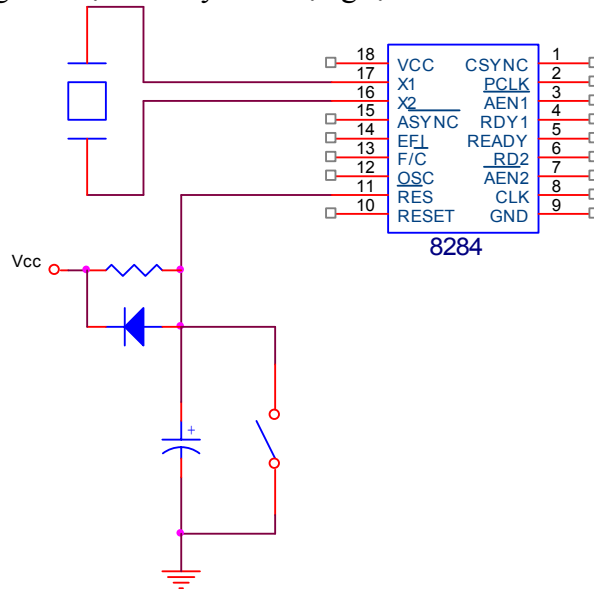


Hình 4.1 – Mạch tạo xung nhịp 8284

CSYNC (Clock Synchronisation): ngõ vào xung đồng bộ chung khi hệ thống có các 8284 dùng dao động ngoài tại chân EFL. Khi dùng mạch dao động trong thì phải nối đất.

PCLK (Peripheral Clock): xung nhịp $f = f_x/6$ (f_x là tần số thạch anh)

$\overline{AEN1}$, $\overline{AEN2}$ (Address Enable): cho phép chọn các chân RDY1, RDY2 báo hiệu trạng thái sẵn sàng của bộ nhớ hay thiết bị ngoại vi



Hình 4.2 – Mạch khởi động cho 8284

RDY1, RDY2 (Bus ready): tạo các chu kỳ đợi ở CPU



READY: nối đến chân READY của μP .

CLK (Clock): xung nhịp $f = f_x/3$, nối với chân CLK của μP .

RESET: nối với chân RESET của μP , là tín hiệu khởi động lại toàn hệ thống

\overline{RES} (Reset Input): chân khởi động cho 8284

OSC: ngõ ra xung nhịp có tần số f_x

F/ \overline{C} (Frequency / Crystal): chọn nguồn tín hiệu chuẩn cho 8284, nếu ở mức cao thì chọn tần số xung nhịp bên ngoài, ngược lại thì dùng xung nhịp từ thạch anh

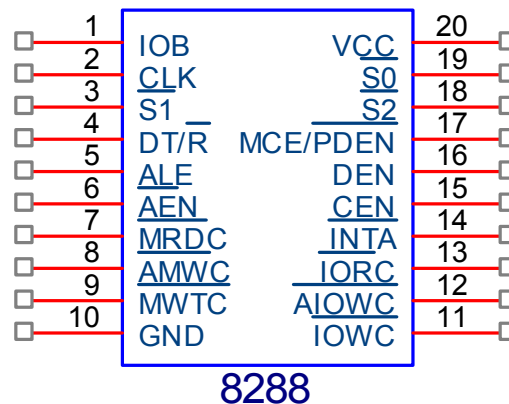
EFI (External Frequency Input): xung nhịp từ bộ dao động ngoài

\overline{ASYN} : chọn chế độ làm việc cho tín hiệu RDY.

X1,X2: ngõ vào của thạch anh

1.2. Mạch điều khiển bus 8288

Mạch điều khiển bus 8288 lấy một số tín hiệu điều khiển của μP và cung cấp các tín hiệu điều khiển cần thiết cho hệ vi xử lý.



Hình 4.3 – Mạch điều khiển bus 8288

IOB (Input / Output Bus Mode): điều khiển để 8288 làm việc ở các chế độ bus khác nhau.

CLK (Clock): ngõ vào lấy từ xung nhịp hệ thống.

$\overline{S2}$, $\overline{S1}$, $\overline{S0}$: các tín hiệu trạng thái lấy trực tiếp từ μP . Tùy theo các giá trị nhận được mà 8288 sẽ đưa các tín hiệu theo bảng 4.1.

Bảng 4.1:

$\overline{S2}$	$\overline{S1}$	$\overline{S0}$	Tạo tín hiệu
0	0	0	\overline{INTA}
0	0	1	\overline{IORC}
0	1	0	\overline{IOWC} , \overline{AIOWC}
0	1	1	Không
1	0	0	\overline{MRDC}
1	0	1	\overline{MRDC}
1	1	0	\overline{MWTC} , \overline{AMWC}
1	1	1	Không



$\overline{DT/R}$ (Data Transmit/Receive): μP truyền (1) hay nhận (0) dữ liệu.

\overline{ALE} (Address Latch Enable): tín hiệu cho phép chốt địa chỉ

\overline{AEN} (Address Enable): chờ thời gian trễ khoảng 150 ns sẽ tạo các tín hiệu điều khiển ở đầu ra của 8288 để đảm bảo rằng địa chỉ sử dụng đã hợp lệ.

\overline{MRDC} (Memory Read Command): điều khiển đọc bộ nhớ

\overline{MWTC} (Memory Write Command): điều khiển ghi bộ nhớ

\overline{AMWC} (Advanced MWTC),: giống như \overline{MWTC} nhưng hoạt động sớm hơn một chút dùng cho các bộ nhớ chậm đáp ứng kịp tốc độ μP .

\overline{IOWC} (I/O Write Command): điều khiển ghi ngoại vi

\overline{AIOWC} (Advanced IOWC),: giống như \overline{IOWC} nhưng hoạt động sớm hơn một chút dùng cho các ngoại vi chậm đáp ứng kịp tốc độ μP .

\overline{IORC} (I/O Read Command): điều khiển đọc ngoại vi

\overline{INTA} (Interrupt Acknowledge): ngõ ra thông báo μP chấp nhận yêu cầu ngắt của thiết bị ngoại vi

\overline{CEN} (Command Enable): cho phép đưa ra các tín hiệu của 8288.

\overline{DEN} (Data Enable): tín hiệu điều khiển bus dữ liệu thành bus cục bộ hay bus hệ thống.

$\overline{MCE} / \overline{PDEN}$ (Master Cascade Enable / Peripheral Data Enable): định chế độ làm việc cho mạch điều khiển ngắt PIC 8259.

2. Giao tiếp với thiết bị ngoại vi

2.1. Các kiểu giao tiếp vào / ra

2.1.1. Thiết bị ngoại vi có địa chỉ tách rời với bộ nhớ

Trong cách giao tiếp này, bộ nhớ dùng toàn bộ không gian 1 MB. Các thiết bị ngoại vi sẽ có một không gian 64 KB cho mỗi loại cổng. Trong kiểu giao tiếp này, ta phải dùng tín hiệu $\overline{IO/M}$ và các lệnh trao đổi dữ liệu thích hợp.

Bộ nhớ: $\overline{IO/M} = 0$, dùng lệnh MOV

Ngoại vi: $\overline{IO/M} = 1$, dùng lệnh IN (nhập) hay OUT (xuất)

2.1.2. Thiết bị ngoại vi và bộ nhớ có chung không gian địa chỉ

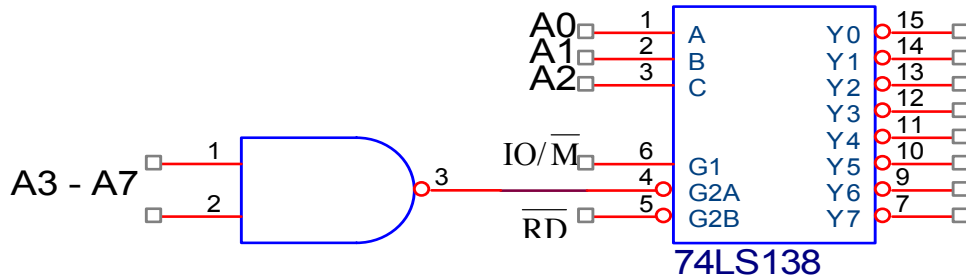
Trong kiểu giao tiếp này, thiết bị ngoại vi sẽ chiếm một vùng nào đó trong không gian địa chỉ 1 MB và ta chỉ dùng lệnh MOV để thực hiện trao đổi dữ liệu.

2.2. Giải mã địa chỉ cho thiết bị vào / ra

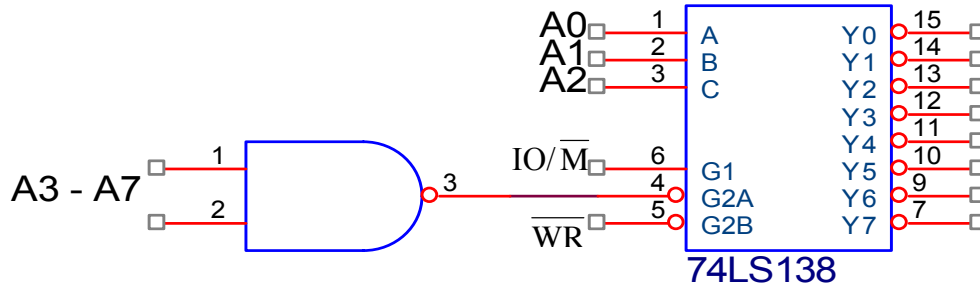
Việc giải mã địa chỉ cho thiết bị ngoại vi cũng tương tự với việc giải mã địa chỉ cho bộ nhớ. Thông thường, các cổng có địa chỉ 8 bit A0 – A7. Tuy nhiên, trong một số hệ vi xử lý, các cổng sẽ có địa chỉ 16 bit.

Ta có thể dùng mạch NAND để tạo tín hiệu chọn cổng nhưng mạch này chỉ có thể giải mã cho 1 cổng. Trong trường hợp cần nhiều tín hiệu chọn cổng, ta có thể dùng bộ giải mã 74LS138 để giải mã cho 8 cổng khác nhau.





(a) Giải mã cho công vào



(b) Giải mã cho công ra

Hình 4.4 – Giải mã cho các cổng

2.3. Các mạch cổng đơn giản

Các mạch cổng có thể được xây dựng từ các mạch chốt 8 bit (74LS373: kích theo mức, 74LS374: kích theo cạnh), các mạch đệm 8 bit (74LS245). Chúng được dùng trong các giao tiếp đơn giản để μ P và ngoại vi hoạt động tương thích với nhau.

2.4. Giao tiếp vào/ra song song lập trình được 8255A PPI (Programmable Peripheral Interface)

2.4.1. Giới thiệu

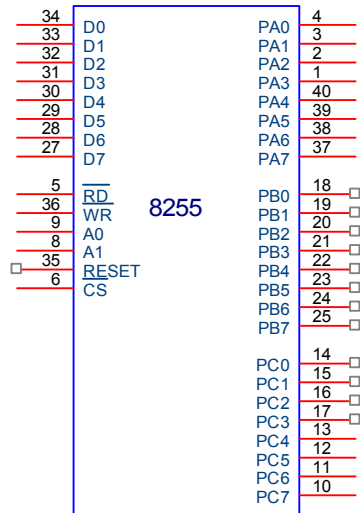
8255A là thiết bị xuất nhập song song lập trình được. Nó là một thiết bị I/O đa dụng có thể sử dụng với bất cứ μ P nào, có thể lập trình để truyền dữ liệu, từ I/O thông thường đến I/O interrupt.

8255A có thể chia thành 3 Port: A, B và C; mỗi port 8 bit trong đó Port C có thể sử dụng như 8 bit riêng hay chia thành 2 nhóm, mỗi nhóm 4 bit: PCH (PC7 ÷ PC4) và PCL (PC3 ÷ PC0).

8255A có thể hoạt động ở 2 chế độ (mode): BSR (Bit Set/Reset) và I/O.

- ❖ **Chế độ BSR:** dùng để đặt hay xóa các bit của Port C.
- ❖ **Chế độ I/O:** gồm có 3 chế độ:
 - Chế độ 0: tất cả các Port làm việc như các Port I/O đơn giản.
 - Chế độ 1 (chế độ bắt tay: handshake): các Port A và B dùng các bit của Port C làm tín hiệu bắt tay. Trong chế độ này, các kiểu truyền dữ liệu I/O có thể được cài đặt, kiểm tra trạng thái và ngắt.
 - Chế độ 2: Port A có thể dùng để truyền dữ liệu song hướng dùng các tín hiệu bắt tay từ Port C còn Port B được thiết lập ở chế độ 0 hay 1.

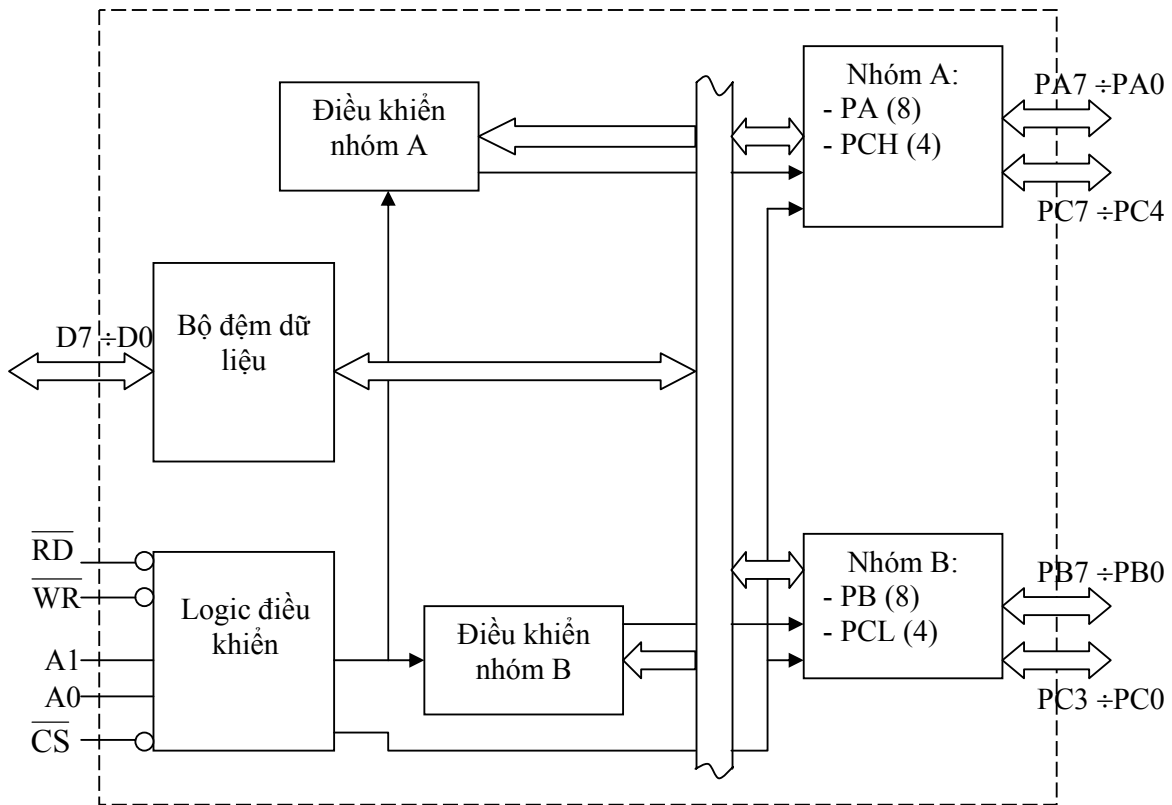




D7 – D0: bus dữ liệu
 PA7 – PA0: Port A
 PB7 – PB0: Port B
 PC7 – PC0: Port C
 A1, A0: giải mã
 RESET: ngõ vào Reset
 \overline{CS} : Chip Select
 \overline{RD} : Read
 WR: Write
 VCC: +5V
 GND: 0V

Hình 4.5 – Sơ đồ chân của 8255A

2.4.2. Sơ đồ khối



Hình 4.6 – Sơ đồ khối của 8255A

Logic điều khiển của 8255A gồm có 6 đường:

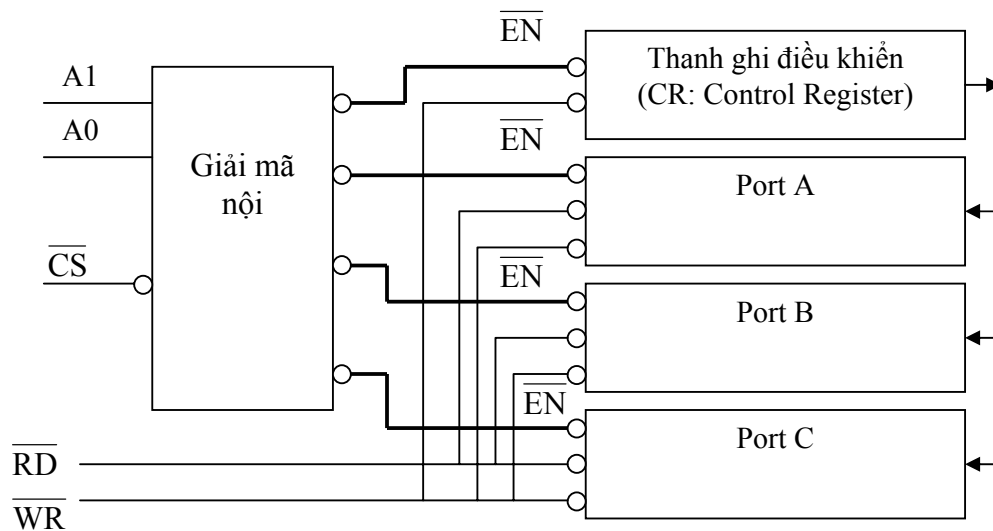
- \overline{RD} (Read): cho phép ĐỌC. Khi chân này ở mức THẤP thì cho phép đọc dữ liệu từ Port I/O đã chọn.



- \overline{WR} (Write): cho phép ghi. Khi chân này ở mức THẤP thì cho phép ghi dữ liệu ra Port I/O đã chọn.
- RESET: khi chân này ở mức cao thì sẽ xoá thanh ghi điều khiển và đặt các Port ở chế độ nhập.
- \overline{CS} (Chip Select): chân chọn chip, thông thường \overline{CS} được nối vào địa chỉ giải mã.
- A1, A0: giải mã xác định Port

Bảng 4.2:

\overline{CS}	A1	A0	Chọn
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Thanh ghi điều khiển
1	x	x	8255A không hoạt động



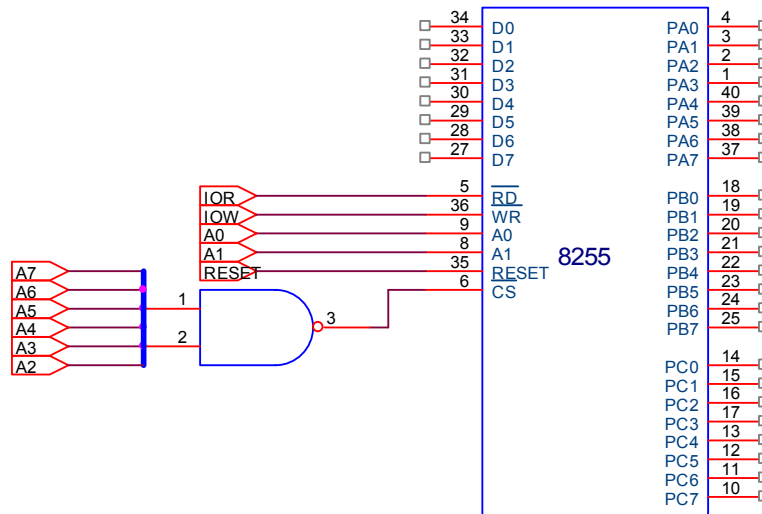
Hình 4.7 – Giải mã chọn các Port

Ví dụ: Xét sơ đồ kết nối 8255A như hình vẽ trang bên:

Theo bảng 4.2, để chọn Port A, ta phải có:

$$\begin{cases} \overline{CS} = 0 \\ A1 = 0 \\ A0 = 0 \end{cases}$$





Hình 4.8 – Logic chọn chip 8255A

Mà $\overline{CS} = 0$ khi $A7 = A6 = A5 = A4 = A3 = A2 = 1$. Từ đó ta được địa chỉ Port I/O như sau:

Bảng 4.3:

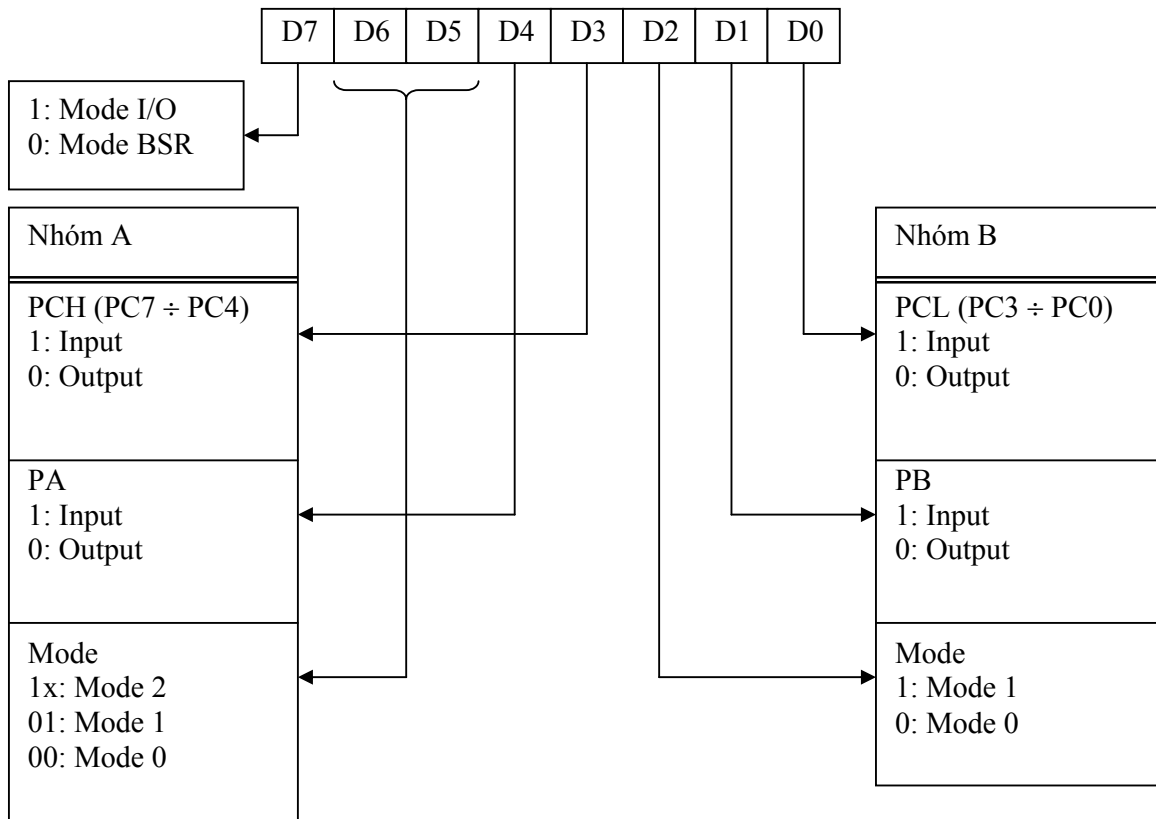
\overline{CS}						A1	A0	Port	Địa chỉ hex
A7	A6	A5	A4	A3	A2	A1	A0		
1	1	1	1	1	1	0	0	A	FCh
						0	1	B	FDh
						1	0	C	FEh
						1	1	CR	FFh

❖ **Thanh ghi điều khiển:**

Như đã biết, 8255A có 2 chế độ hoạt động và các Port của nó có thể có các chức năng I/O khác nhau. Để xác định chức năng của các Port, 8255A có một thanh ghi điều khiển (CR: Control Register). Nội dung của thanh ghi này gọi là từ điều khiển (CW: Control Word). Thanh ghi điều khiển sẽ được truy xuất khi $A1 = A0 = 1$. Chú ý rằng ta không thể thực hiện tác vụ Đọc đối với thanh ghi này.

Nếu bit $D7 = 0$, Port C làm việc ở chế độ BSR nhưng từ điều khiển BSR không ảnh hưởng đến chức năng các Port A, B.





Hình 4.9 – Dạng từ điều khiển cho 8255A ở chế độ I/O

2.4.3. Mode 0: Xuất/nhập đơn giản

Trong chế độ này, mỗi port (hay nửa port của Port C) làm việc như các port nhập hay xuất với các tính chất sau:

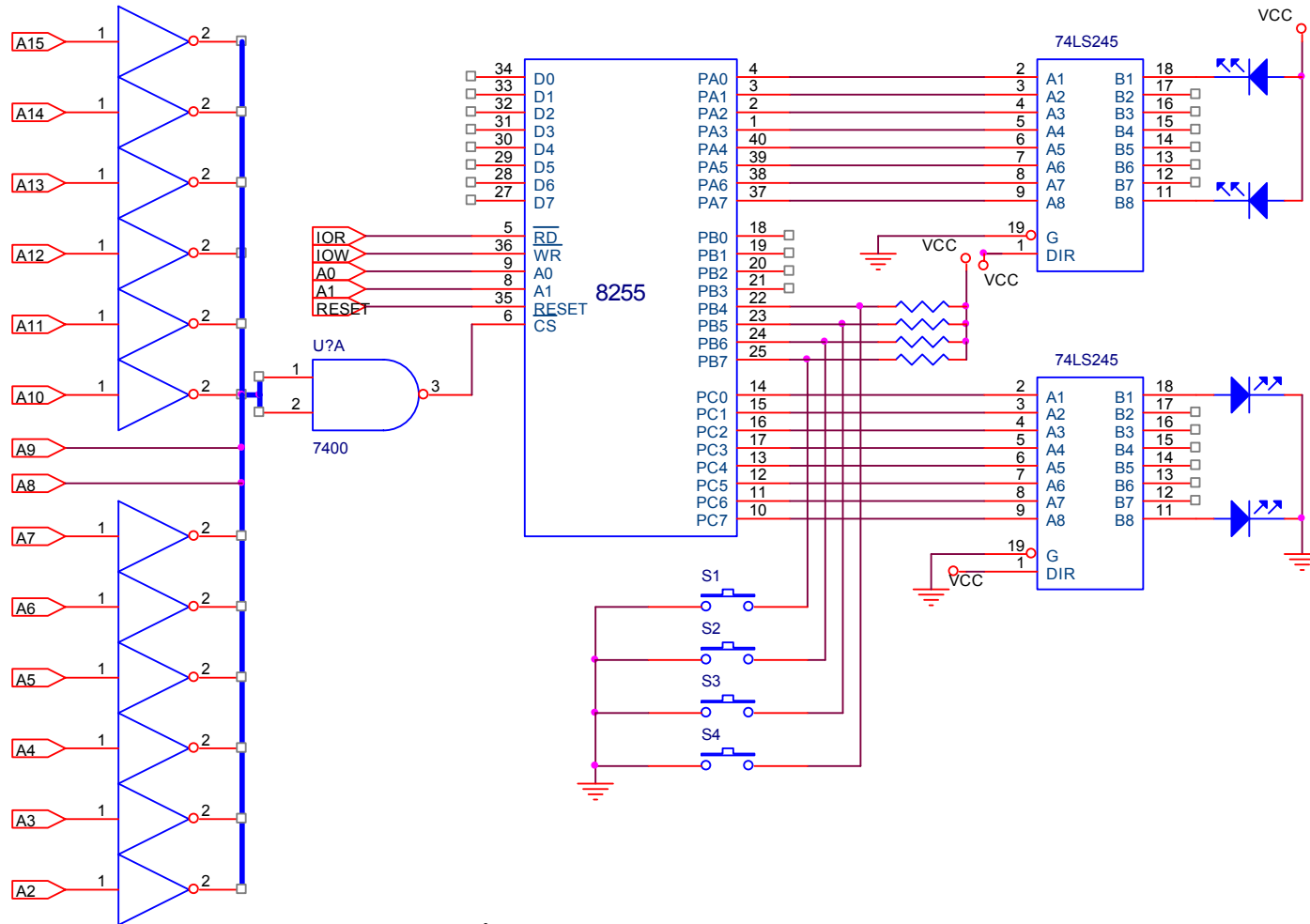
- Các ngõ ra được chốt.
- Các ngõ vào không được chốt.
- Các port không có khả năng bắt tay và ngắt.

Để giao tiếp với ngoại vi thông qua 8255A cần phải:

- *Xác định địa chỉ của các port A, B, C và CR thông qua các chân chọn chip \overline{CS} và giải mã A1, A0.*
- *Ghi từ điều khiển vào thanh ghi điều khiển.*
- *Ghi các lệnh I/O để giao tiếp với ngoại vi qua các port A, B, C.*



Ví dụ: Xét sơ đồ kết nối 8255A như sau:



Hình 4.10 – Giao tiếp các port 8255A ở mode 0



- Xác định địa chỉ port:

Bảng 4.4:

\overline{CS}														A1	A0	Port	Địa chỉ hex
A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0		
0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	A	300h
														0	1	B	301h
														1	0	C	302h
														1	1	CR	303h

- Từ điều khiển:

Bảng 4.5:

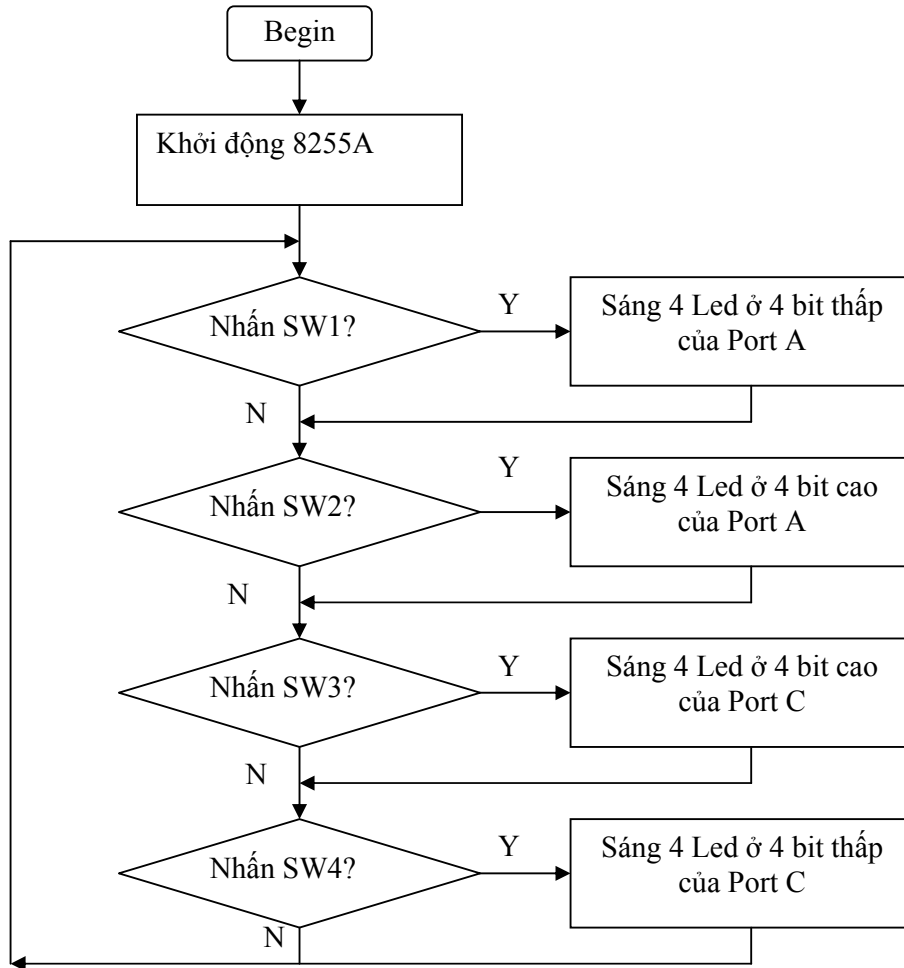
D7	D6	D5	D4	D3	D2	D1	D0	
1	0	0	0	0	0	1	0	= 82h
I/O mode	Nhóm A ở mode 0		PA: Output	PCH: Output	Nhóm B ở mode 0		PB: Input	PCL: Output

Các Port của 8255A được khởi động bằng cách đặt từ điều khiển 82h vào thanh ghi điều khiển.

Trong sơ đồ kết nối này, 4 bit cao của Port B dùng làm Port nhập còn Port A và Port C làm Port xuất. Các tác vụ Đọc và Ghi được phân biệt bằng các tín hiệu điều khiển \overline{IOR} và \overline{IOW} .



- Lưu đồ giải thuật:



- Chương trình:

```

.MODEL    SMALL
.STACK   100h
.CODE
main PROC
; Định cấu hình cho 8255
MOV  AL,82h           ; Từ điều khiển (CW) là 82h
MOV  DX,303h         ; Địa chỉ thanh ghi điều khiển (CR)
OUT  DX,AL           ; Ghi CW vào CR
cont: MOV  DX,301h     ; Địa chỉ Port B
      IN   AL,DX      ; Đọc dữ liệu từ Port B (công tắc)
      AND  AL,0F0h    ; Che 4 bit thấp
      MOV  AH,AL
      CMP  AH,01110000b ; Kiểm tra công tắc 1
      JNE  notSW1     ; Nếu không nhấn
      MOV  AL,0Fh     ; Nếu nhấn công tắc 1 thì
      MOV  DX,300h    ; xuất ra Port A
      OUT  DX,AL      ; để sáng 4 Led ở 4 bit thấp (Port A)

notSW1:  CMP  AH,10110000b ; Kiểm tra công tắc 2
         JNE  notSW2     ; Nếu không nhấn
  
```



```

MOV AL,0F0h           ; Nếu nhấn công tắc 2 thì
MOV DX,300h          ; xuất ra Port A
OUT DX,AL             ; để sáng 4 Led ở 4 bit cao (Port A)
notSW2:  CMP AH,11010000b ; Kiểm tra công tắc 3
        JNE notSW3       ; Nếu không nhấn
MOV AL,0Fh           ; Nếu nhấn công tắc 3 thì
MOV DX,302h          ; xuất ra Port C
OUT DX,AL             ; để sáng 4 Led ở 4 bit cao (Port C)
notSW3:  CMP AH,11100000b ; Kiểm tra công tắc 4
        JNE notSW4       ; Nếu không nhấn
MOV AL,F0h           ; Nếu nhấn công tắc 4 thì
MOV DX,302h          ; xuất ra Port C
OUT DX,AL             ; để sáng 4 Led ở 4 bit thấp (Port C)
notSW4:  JMP cont
main    ENDP
END     main

```

2.4.4. Mode BSR

Mode BSR chỉ liên quan đến 8 bit của Port C, có thể đặt hay xoá các bit bằng cách ghi một từ điều khiển thích hợp vào thanh ghi điều khiển. Một từ điều khiển với D7 = 0 gọi là từ điều khiển BSR, từ điều khiển này không làm thay đổi bất cứ từ điều khiển nào được truyền trước đó với D7 = 1, nghĩa là các hoạt động I/O của Port A và B không bị ảnh hưởng bởi từ điều khiển BSR.

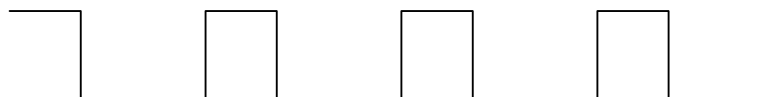
❖ Từ điều khiển BSR:

Từ điều khiển BSR khi được ghi vào thanh ghi điều khiển sẽ đặt hay xoá mỗi lần 1 bit.

D7	D6	D5	D4	D3	D2	D1	D0
0	x	x	X				S/R
Mode BSR	Không sử dụng			Chọn bit			0: Xoá (Reset) 1: Đặt (Set)
				000:	PC0		
				001:	PC1		
				010:	PC2		
				011:	PC3		
				100:	PC4		
				101:	PC5		
				110:	PC6		
				111:	PC7		

Ví dụ: Xét sơ đồ kết nối 8255A như hình 4.10. Giả sử ta cần tạo một sóng chữ nhật tại bit PC0.

Để tạo một sóng chữ nhật tại PC0, ta cần 2 mức logic là 0 và 1 tại PC0.



	D7	D6	D5	D4	D3	D2	D1	D0	
Đặt bit PC0 = 1	0	0	0	0	0	0	0	1	= 01h
Xoá bit PC0 = 0	0	0	0	0	0	0	0	0	= 00h

- Địa chỉ thanh ghi điều khiển (bảng 4.4): 303h
- Chương trình con:

```

bsr:  MOV     AL,01h      ; Từ điều khiển BSR
      MOV     DX,303h   ; Địa chỉ thanh ghi điều khiển (CR)
      OUT     DX,AL     ; Đặt PC0 = 1
      CALL    DELAY1    ; Chờ
      MOV     AL,00h    ; Từ điều khiển BSR
      OUT     DX,AL     ; Xoá PC0 = 0
      CALL    DELAY2    ; Chờ
      JMP     bsr

```

Khi sử dụng ở mode BSR, cần chú ý các điều sau:

- **Để đặt hay xoá các bit ở Port C, từ điều khiển được ghi vào thanh ghi điều khiển chứ không ghi vào Port C.**
- **Một từ điều khiển BSR chỉ ảnh hưởng đến một bit của Port C.**
- **Từ điều khiển BSR không ảnh hưởng đến I/O mode.**

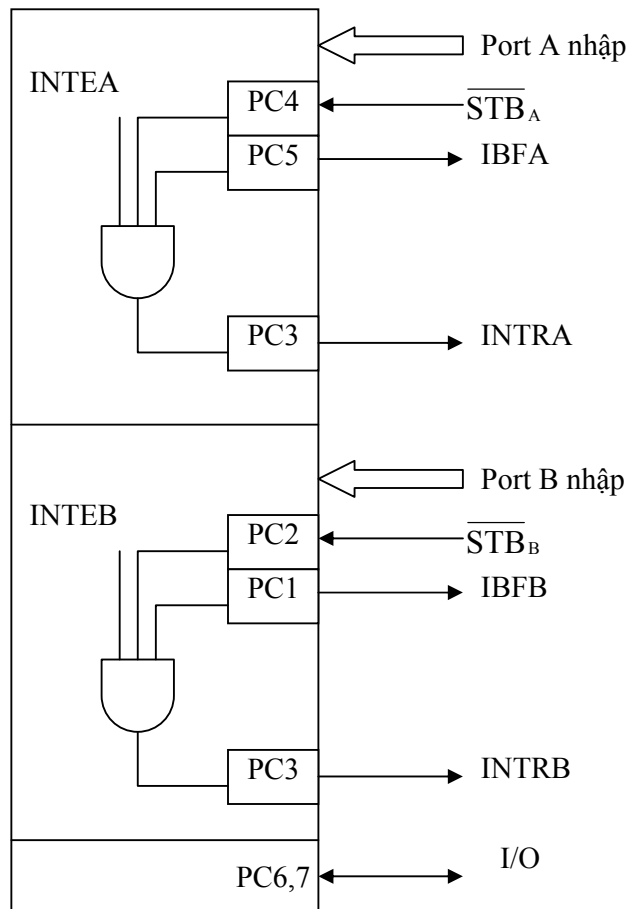
2.4.5. Mode 1: Nhập / xuất với bắt tay (handshake)

Trong mode 1, các tín hiệu bắt tay được trao đổi giữa μP và thiết bị ngoại vi trước khi truyền dữ liệu. Các đặc tính ở chế độ này là:

- Hai Port A, B làm việc như các Port I/O 8 bit.
- Mỗi Port sử dụng 3 đường từ Port C làm các tín hiệu bắt tay. Hai đường còn lại có thể dùng cho các chức năng I/O đơn giản.
- Dữ liệu nhập / xuất được chốt.
- Hỗ trợ ngắt.



2.4.5.1. Các tín hiệu điều khiển nhập

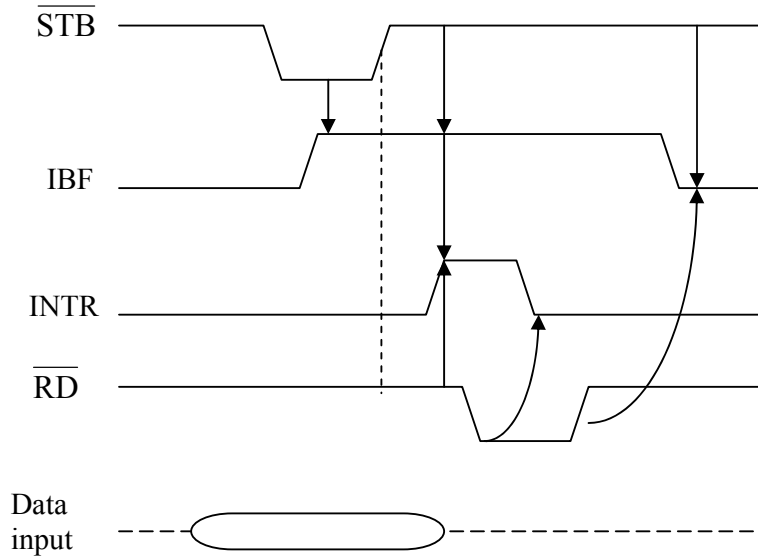


Hình 4.11 – Cấu hình nhập của 8255A ở mode 1

Theo hình vẽ, ta thấy Port A dùng 3 đường tín hiệu trên PC3, PC4 và PC5; Port B dùng 3 đường tín hiệu trên PC0, PC1 và PC2 làm các tín hiệu bắt tay. Các tín hiệu này có các chức năng sau khi các port A và B được đặt cấu hình là nhập:

- \overline{STB} (Strobe Input): tích cực mức thấp, tín hiệu này được tạo bởi thiết bị ngoại vi để xác định rằng ngoại vi đã truyền 1 byte dữ liệu. Khi 8255A đáp ứng \overline{STB} , nó sẽ tạo ra IBF và INTR (hình 4.12).
- IBF (Input Buffer Full): tín hiệu này dùng để xác nhận 8255A đã nhận byte dữ liệu. Nó sẽ bị xoá khi μP đọc dữ liệu.
- INTR (Interrupt Request): Đây là tín hiệu xuất dùng để ngắt μP . Nó được tạo ra nếu \overline{STB} , IBF và INTE (flipflop bên trong) đều ở mức logic 1 và bị xoá bởi cạnh xuống của tín hiệu RD (Hình 4.12).
- INTE (Interrupt Enable): là một flipflop dùng để cho phép hay cấm quá trình tạo ra tín hiệu INTR. Hai flipflop INTEA và INTEB được đặt / xoá dùng BSR mode thông qua PC4 và PC2.





Hình 4.12 – Dạng sóng định thời cho ngõ vào có strobe

❖ **Các từ điều khiển và trạng thái:**

- Từ điều khiển: để xác định từ điều khiển, ta sử dụng hình 3.4.5

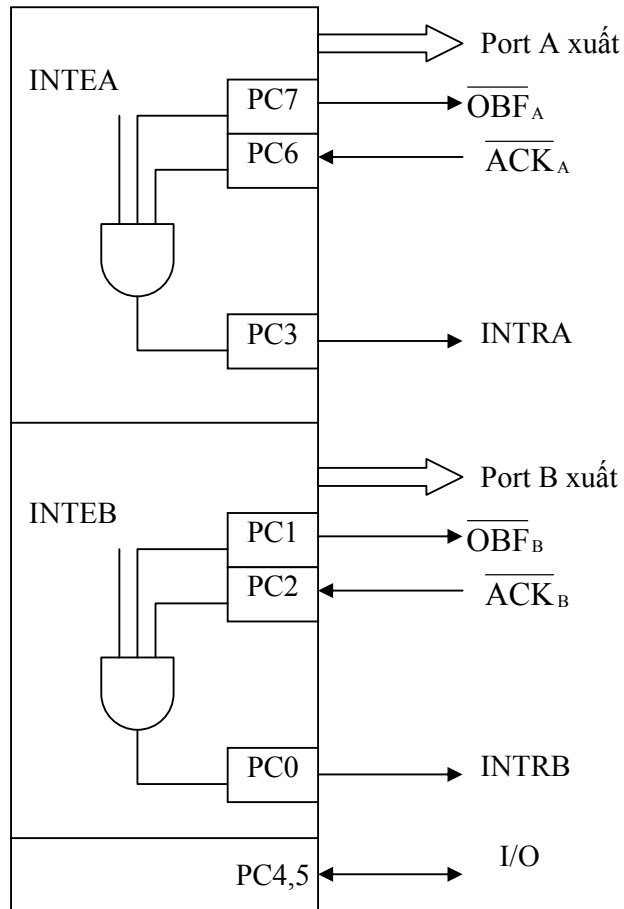
D7	D6	D5	D4	D3	D2	D1	D0
1	0	1	1	1/0	1	1	X
I/O mode	PA: Mode 1		PA: nhập	PC6,7 1: nhập 0: xuất	PB: Mode 1	PB: nhập	

- Từ trạng thái: sẽ được đặt trong thanh ghi tích lũy nếu đọc Port C.

D7	D6	D5	D4	D3	D2	D1	D0
I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB



2.4.5.2. Các tín hiệu điều khiển xuất

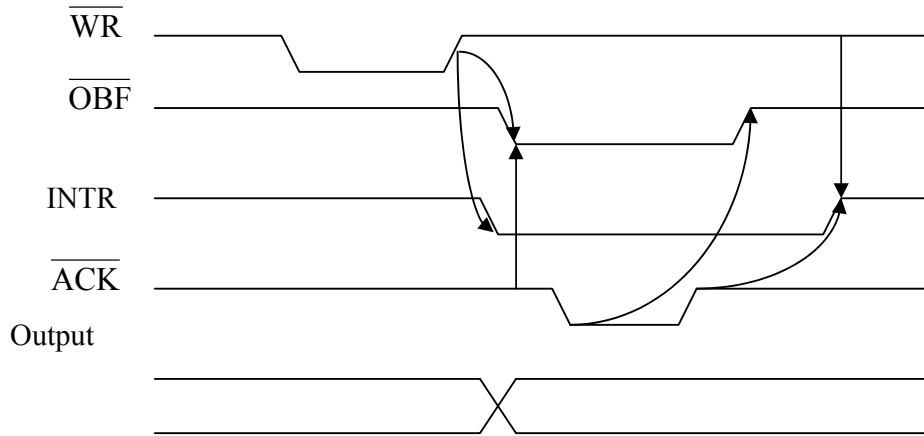


Hình 4.13 – Cấu hình xuất của 8255A ở mode 1

Chức năng các đường tín hiệu :

- \overline{OBF} (Output Buffer Full): tín hiệu này sẽ xuống mức thấp khi μP ghi dữ liệu vào Port xuất của 8225A. Tín hiệu này đưa đến thiết bị ngoại vi để xác định dữ liệu sẵn sàng đưa vào ngoại vi (Hình 4.14). Nó sẽ lên mức cao khi 8255A nhận \overline{ACK} từ ngoại vi.
- \overline{ACK} (Acknowledge): đây là tín hiệu nhập từ ngoại vi (tích cực mức thấp) xác nhận dữ liệu đã nhập vào ngoại vi.
- \overline{INTR} (Interrupt Request): đây là tín hiệu xuất, đặt bằng cạnh lên của tín hiệu \overline{ACK} . Tín hiệu này có thể dùng để ngắt μP yêu cầu byte dữ liệu kế tiếp để xuất. \overline{INTR} được đặt khi \overline{OBF} , \overline{ACK} và \overline{INTE} ở mức logic 1 (Hình 4.14) và được xoá bởi cạnh xuống của tín hiệu \overline{WR}
- \overline{INTE} (Interrupt Enable): đây là flipflop nội dùng để tạo tín hiệu \overline{INTR} . Hai flipflop \overline{INTEA} và \overline{INTEB} điều khiển bằng các bit PC6 và PC2 thông qua BSR mode.





Hình 4.14 – Dạng sóng cho xuất strobe (có lấy mẫu) (với bắt tay)

❖ **Từ điều khiển và trạng thái:**

- Từ điều khiển:

D7	D6	D5	D4	D3	D2	D1	D0
1	0	1	0	1/0	1	0	X
I/O mode	PA: Mode 1		PA: xuất	PC4,5 1: nhập 0: xuất	PB: mode 1	PB: xuất	

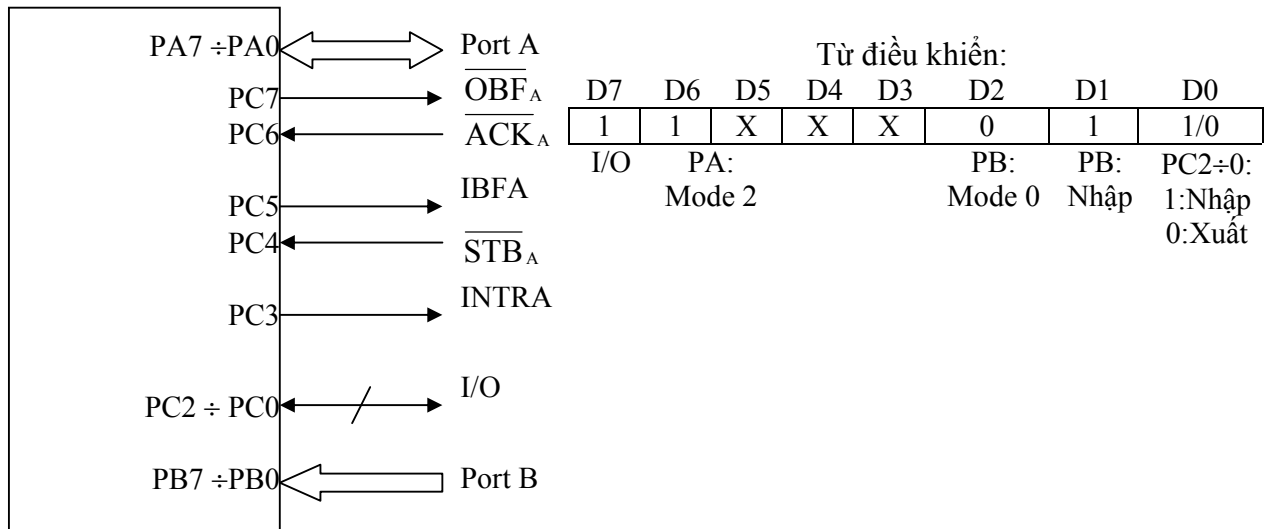
- Từ trạng thái:

D7	D6	D5	D4	D3	D2	D1	D0
$\overline{\text{OBF}}_A$	INTEA	I/O	I/O	INTRA	INTEB	$\overline{\text{OBF}}_B$	INTRB

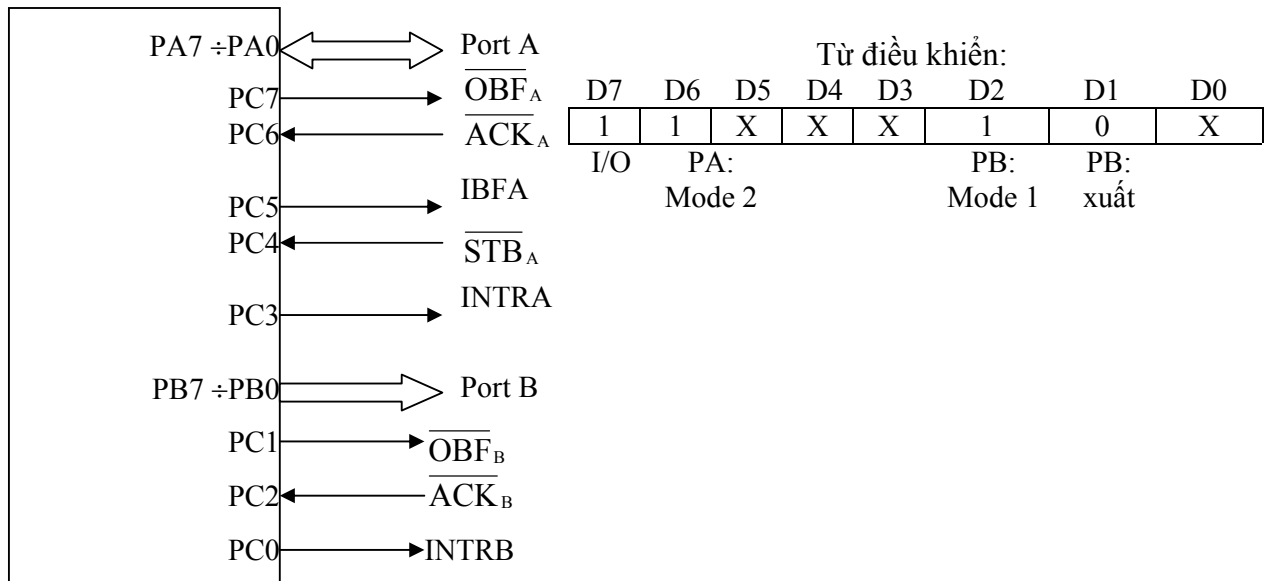
2.4.6. Mode 2: Truyền dữ liệu song hướng

Mode này dùng chủ yếu trong các ứng dụng như truyền dữ liệu giữa hai máy tính hay giao tiếp bộ điều khiển đĩa mềm. Trong mode này, Port A dùng làm Port song hướng và Port B làm việc ở Mode 0 hay 1. Port A sử dụng 5 tín hiệu tại Port C làm các tín hiệu điều khiển để truyền dữ liệu. Ba tín hiệu còn lại của Port C được dùng làm I/O đơn giản hay bắt tay cho Port B.





(a) 8255A ở mode 2 và mode 0 (nhập)



(a) 8255A ở mode 2 và mode 1 (xuất)

Hình 4.15 – 8255A dùng ở Mode 2

2.4.7. Các ví dụ minh họa

2.4.7.1. Giao tiếp với bộ chuyển đổi A/D ADC0804 dùng 8255A ở Mode 0 và Mode BSR

Ta thiết lập 8255A hoạt động như sau:

- Dùng Port A để đọc dữ liệu.
- Dùng PC0, PC3 điều khiển các chân \overline{RD} , \overline{WR} của ADC0804.



Xét sơ đồ mạch có logic chọn chip giống như hình 4.10. Tầm địa chỉ Port từ 300h ÷ 303h.

- Từ điều khiển mode 0:

Port A: nhập

Port B: không sử dụng

Port Clow: port xuất dùng để điều khiển 2 ngõ \overline{RD} , \overline{WR} của ADC0804

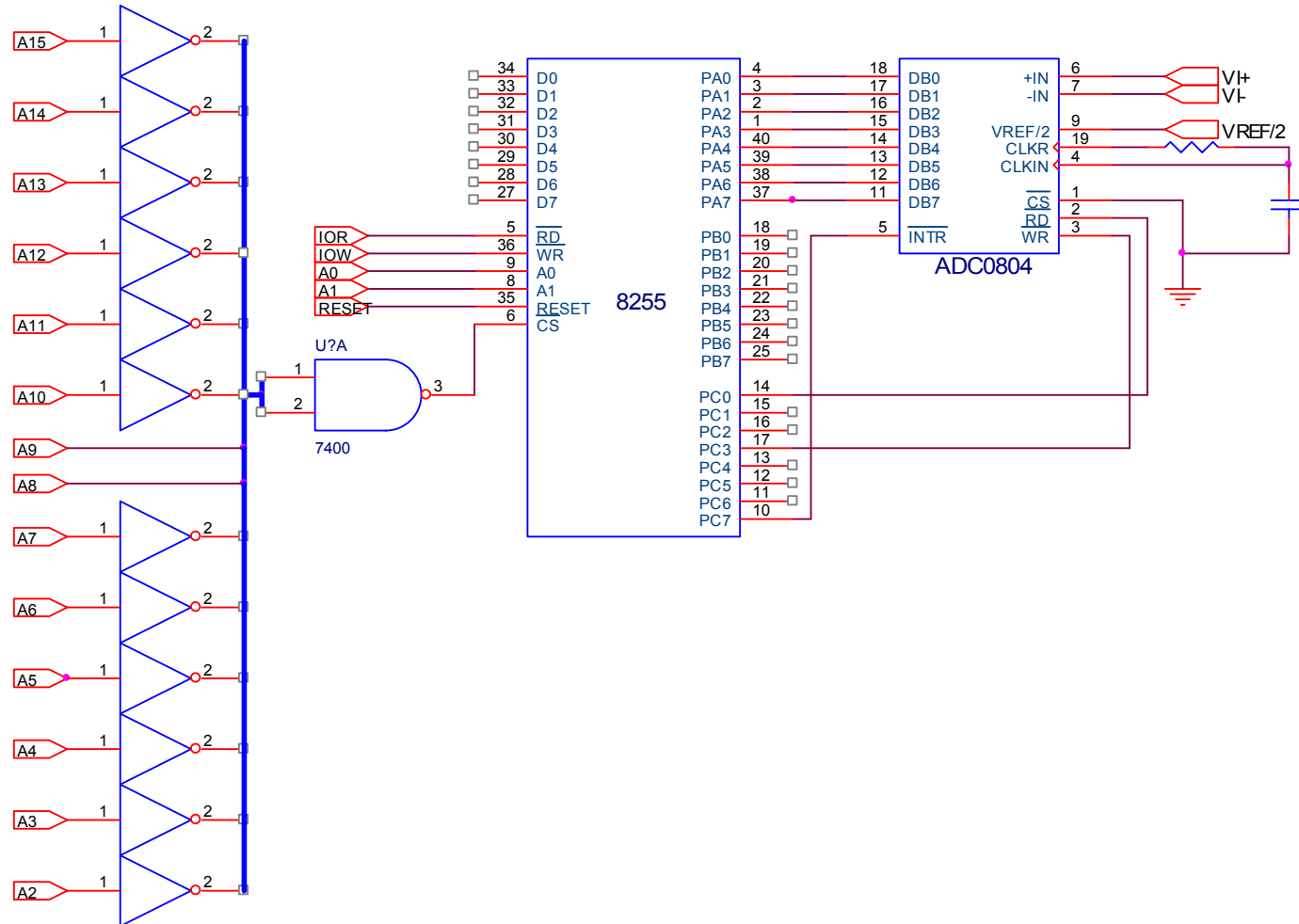
Port Chigh: port nhập dùng để đọc trạng thái ở chân \overline{INTR} của ADC0804

D7	D6	D5	D4	D3	D2	D1	D0	= 90h
1	0	0	1	0	0	0	0	
I/O	PA: mode 0	PA: nhập	PCH: xuất	PB: không sử dụng	PCL: xuất			

- Từ điều khiển BSR:

	D7	D6	D5	D4	D3	D2	D1	D0	
Đặt PC0	0	0	0	0	0	0	0	1	= 01h
Xoá PC0	0	0	0	0	0	0	0	0	= 00h
Đặt PC3	0	0	0	0	0	1	1	1	= 07h
Xoá PC3	0	0	0	0	0	1	1	0	= 06h





Hình 4.16 – Giao tiếp bộ chuyển đổi A/D ADC0804 dùng 8255A



- Mô tả chương trình:
 - Khởi động 8255A bằng cách đặt từ điều khiển mode 0 vào thanh ghi điều khiển.
 - Cấp một xung vào chân \overline{RD} của 8255A.
 - Đọc trạng thái của ADC0804 từ chân \overline{INTR} .
 - Nếu $\overline{INTR} = 0$ thì cấp một xung vào chân \overline{WR} của ADC0804 để xuất dữ liệu.
 - Đọc dữ liệu từ ADC0804 vào thông qua Port A.
- Đoạn chương trình thực hiện:

```

adc:  MOV     DX,303h    ; Địa chỉ thanh ghi điều khiển (CR)
      MOV     AL,90h   ; Từ điều khiển (CW)
      OUT     DX,AL    ; Ghi CW vào CR
      MOV     AL,01h   ; Từ điều khiển BSR để PC0 = 1 ( $\overline{RD} = 1$ )
      OUT     DX,AL    ; Xuất ra CR
      MOV     AL,07h   ; Từ điều khiển BSR để PC3 = 1
      OUT     DX,AL    ; Xuất ra CR
      MOV     AL,06h   ; Từ điều khiển BSR để PC3 = 0, tạo xung  $\overline{WR}$ 
      OUT     DX,AL    ; Xuất ra CR
      CALL    DELAY    ; Chờ quá trình chuyển đổi thực hiện xong
      MOV     AL,07h   ; Từ điều khiển BSR để PC3 = 1
      OUT     DX,AL    ; Xuất ra CR
      MOV     DX,300h  ; Địa chỉ Port A
      IN      AL,DX    ; Đọc dữ liệu đã chuyển đổi từ ADC0804
      MOV     AL,01h   ; Từ điều khiển BSR để PC0 = 1 ( $\overline{RD} = 1$ )
      OUT     DX,AL    ; Xuất ra CR
      RET                    ; vào từ Port A của 8255A

```

2.4.7.2. Giao tiếp với máy in trong chế độ bắt tay (Mode 1)

Xét mạch giao tiếp 8255A ở mode 1 với Port A được dùng làm Port nhập từ bàn phím với I/O interrupt và Port B được thiết kế làm Port xuất tới máy in với I/O kiểm tra trạng thái. Ta cần thực hiện các công việc sau:

- Xác định địa chỉ Port.
- Xác định từ điều khiển để Port A nhập và Port B xuất ở Mode 1.
- Xác định từ điều khiển BSR cho phép ngắt (INTEA).
- Xác định các byte mặt nạ để kiểm tra các đường \overline{OBF}_B trong I/O kiểm tra trạng thái.
- Viết các lệnh khởi động và chương trình con in các ký tự chứa trong bộ nhớ.

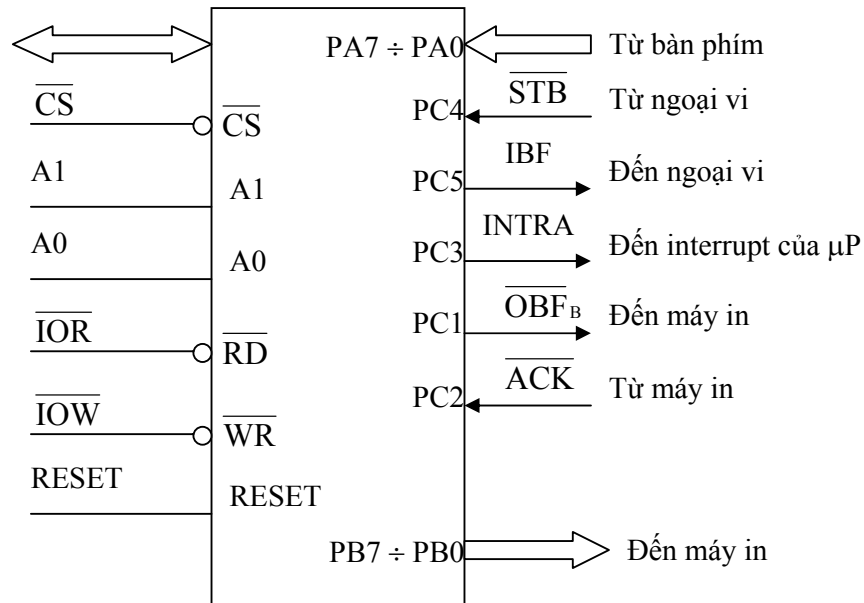
Giả sử logic chọn chip như hình 4.10, địa chỉ Port cho trong bảng 4.4:

```

PA: FCh
PB: FDh
PC: FEh
CR: FFh

```





Hình 4.17 – Giao tiếp 8255A ở Mode 1

- **Từ điều khiển:** Port A nhập, Port B xuất ở Mode 1

D7	D6	D5	D4	D3	D2	D1	D0	= B4h
1	0	1	1	0	1	0	0	
I/O	PA: Mode 1	PA: nhập	Không sử dụng	PB: Mode 1	PB: xuất	Không sử dụng		

- **Từ điều khiển BSR:** dùng để đặt flipflop cho phép ngắt của Port A (INTEA), bit PC4 = 1

D7	D6	D5	D4	D3	D2	D1	D0	= 09h
0	0	0	0	1	0	0	1	
BSR mode	Không sử dụng			Bit PC4	Đặt bit (Set)			

- **Từ trạng thái kiểm tra OBF_B:**

D7	D6	D5	D4	D3	D2	D1	D0
X	x	x	x	x	x	OBF _B	X

Byte mặt nạ: 0000 0010b

❖ **Khởi động:**

```

MOV     DX, 0FFh    ; Khởi động 8255A
MOV     AL, 0B4h    ; ở Mode 1, Port A nhập
OUT     DX, AL      ; Port B xuất
MOV     AL, 09h     ; Đặt INTEA
OUT     DX, AL      ; cho phép INTRA
CALL    print
    
```



❖ **Chương trình con PRINT:**

print:	LEA	DX,msg	; Chỉ đến vị trí chứa các ký tự
	MOV	SI, DX	
	ADD	SI,2	
next:	LODSB		; Lấy ký tự từ bộ nhớ
	CMP	AL,0	; Nếu không còn ký tự nào
	JNE	cont	; thì kết thúc
	JMP	exit	
cont:	MOV	AH,AL	; Lưu ký tự vừa đọc
	MOV	DX,0FEh	
status:	IN	AL,DX	; Đọc vào từ Port C
	AND	AL,02h	; Chỉ nhận PC1
	JE	status	; Nếu máy in không sẵn sàng thì chờ
	MOV	AL,AH	
	MOV	DX,0FDh	; Xuất ký tự đã nhận ra
	OUT	DX,AL	; máy in (Port B)
	JMP	next	; Xử lý ký tự kế tiếp
exit:	RET		

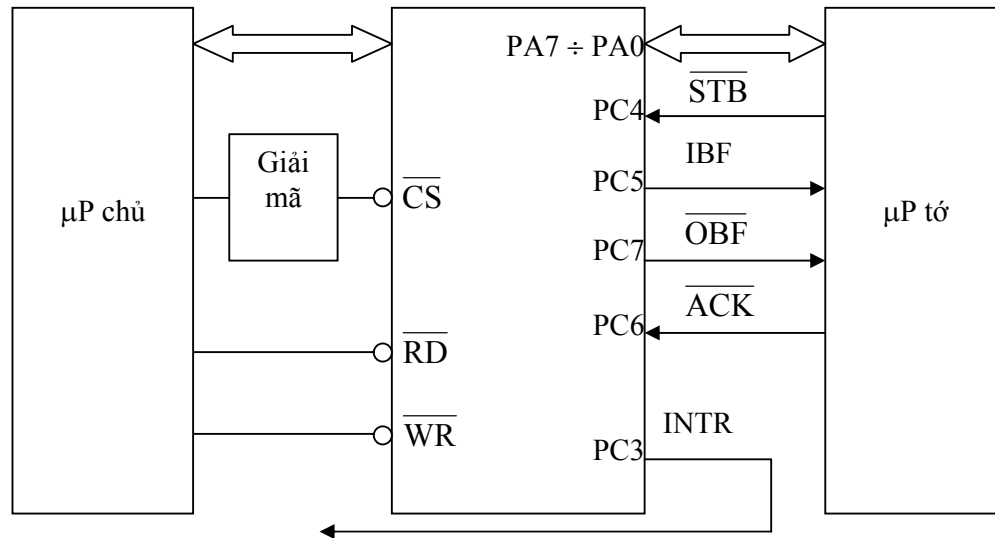
❖ **Mô tả chương trình:**

- Ta sử dụng 8255A trong phần thiết kế này cho phép 2 hoạt động: xuất ra máy in và lấy dữ liệu vào từ bàn phím. Giao tiếp với máy in dùng kiểm tra trạng thái và giao tiếp bàn phím dùng ngắt.
- Trong chương trình con PRINT, ký tự được đặt trong thanh ghi tích lũy A và trạng thái đọc từ Port C. Bàn đầu Port B trống, bit PC1 (\overline{OBF}_B) ở mức cao. Ta thực hiện lệnh OUT gửi dữ liệu ra Port B. Tín hiệu \overline{OBF}_B sẽ xuống mức thấp do tác động cạnh lên của tín hiệu \overline{WR} , xác định rằng dữ liệu đã gửi ra máy in. Sau khi nhận byte dữ liệu, máy in gửi trở lại tín hiệu \overline{ACK} xác định đã nhận. Tín hiệu \overline{ACK} làm cho \overline{OBF}_B ở mức cao xác định máy in sẵn sàng nhận ký tự kế tiếp và chương trình con PRINT tiếp tục thực hiện cho đến khi không còn ký tự nào trong vùng nhớ.
- Nếu một phím được nhấn khi chương trình con PRINT đang thực thi, byte dữ liệu truyền tới Port A và \overline{STB}_A xuống mức thấp, đặt \overline{IBFA} lên mức cao. Khi \overline{STB}_A trở lại mức cao thì sẽ tạo ra INTR. Tín hiệu này tạo ngắt đến μP và điều khiển được chuyển đến chương trình phục vụ ngắt. Chương trình này sẽ đọc nội dung Port A, cho phép ngắt và quay về chương trình con PRINT.



2.4.7.3. Truyền dữ liệu giữa hai microprocessor trong xử lý phân bổ dùng 8255A ở Mode 2

Ta thiết kế mạch giao tiếp để truyền dữ liệu hai chiều dạng chủ – tớ (master – slave) giữa hai μP .



Hình 4.18 – Thông tin 2 chiều giữa 2 μP dùng 8255A

Hình 4.18 chỉ sơ đồ khối thiết lập thông tin hay chiều giữa chủ và tớ. Sơ đồ khối chỉ hai data bus hai chiều – chủ và tớ – được nối với nhau thông qua 8225A, trong đó 8225A làm việc như thiết bị giao tiếp của μP chủ. Port A của 8225A được dùng để truyền dữ liệu hai chiều và 4 tín hiệu từ port C được dùng để bắt tay. Quá trình truyền dữ liệu tương tự như Mode 1 của 8225A. Khi μP chủ ghi 1 byte dữ liệu vào 8225A tín hiệu $\overline{\text{OBF}}$ xuống mức thấp để báo cho μP tớ biết là đã gửi dữ liệu vào, μP tớ sẽ báo nhận được khi nó đọc byte dữ liệu này. Tương tự, hai tín hiệu bắt tay khác được dùng khi μP tớ truyền 1 byte dữ liệu đến μP chủ.

μP chủ đòi hỏi các port I/O dùng để đọc và ghi dữ liệu và kiểm tra trạng thái của các tín hiệu bắt tay. Tương tự, μP tớ cần các port I/O để thực hiện Đọc và Ghi. Truyền dữ liệu có thể được thực hiện bằng cách kiểm tra trạng thái hay dùng ngắt. Tốc độ xử lý dữ liệu đối với μP chủ quan trọng hơn nên thường dùng μP chủ ở chế độ ngắt và μP tớ ở chế độ kiểm tra trạng thái. Ở ví dụ này, ta sẽ dùng cả 2 μP ở chế độ kiểm tra trạng thái.

Các hoạt động truyền dữ liệu giữa 2 I/O kiểm tra trạng thái có thể liệt kê như sau:

❖ Truyền dữ liệu từ μP chủ đến μP tớ:

1. μP chủ đọc trạng thái của $\overline{\text{OBF}}$ để kiểm tra xem μP tớ đã đọc dữ liệu chưa. Đây là chức năng nhập cho μP chủ.
2. μP chủ ghi dữ liệu vào Port A và 8225A báo cho μP tớ biết bằng cách đưa tín hiệu $\overline{\text{OBF}}$ xuống mức thấp. Đây là chức năng xuất của μP chủ.



3. μP tớ kiểm tra tín hiệu $\overline{\text{OBF}}$ (từ μP chủ) để xác định tính sẵn sàng của dữ liệu. Đây là chức năng nhập đối với μP tớ.
4. μP tớ đọc dữ liệu từ Port A và báo cho biết đã nhận được bằng cách đưa tín hiệu $\overline{\text{ACK}}$ xuống mức thấp. Đây là chức năng nhập đối với μP tớ.

❖ **Truyền dữ liệu từ μP tớ đến μP chủ:**

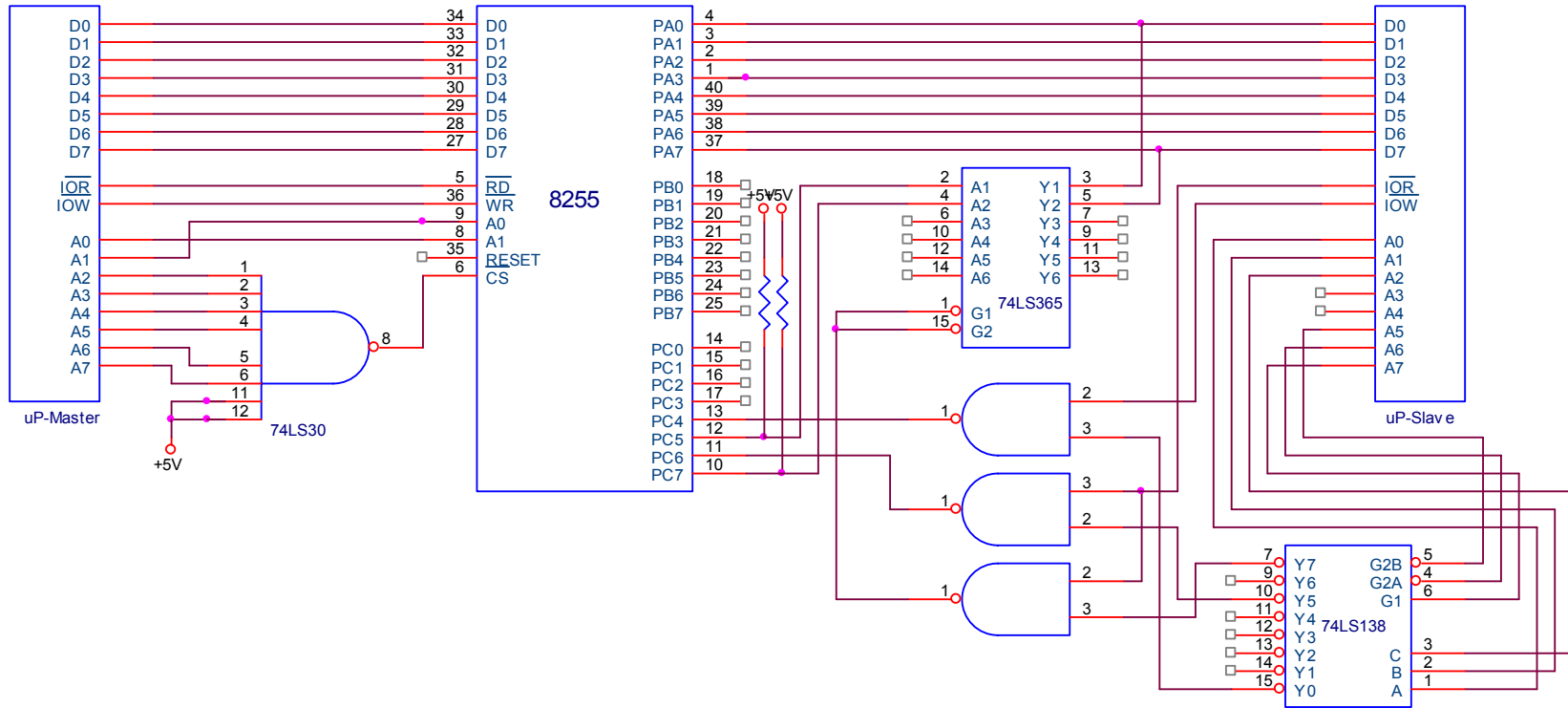
1. μP tớ kiểm tra tín hiệu bắt tay IBF để xem port A có sẵn sàng truyền dữ liệu hay không để truyền 1 byte. Đây là chức năng nhập đối với μP tớ.
2. μP đặt byte dữ liệu lên data bus và báo cho 8225A biết rằng sẵn sàng gửi dữ liệu bằng cách dùng tín hiệu $\overline{\text{STB}}$. Đây là chức năng xuất đối với μP tớ.
3. 8225A đưa IBF lên mức cao, μP chủ đọc tín hiệu này để xác định dữ liệu sẵn sàng chưa. Đây là chức năng nhập đối với μP chủ.
4. μP chủ đọc byte dữ liệu. Đây là chức năng nhập đối với μP chủ.

❖ **Kết nối phân cứng:**

Hình 4.19 cho thấy sơ đồ kết nối các port cần thiết và logic chọn chip cho 8255A. μP chủ thực hiện giải mã chọn 8255A dùng cổng NAND 8 ngõ vào nên 8255A được chọn khi tất cả các ngõ vào của cổng NAND đều ở mức 1. Từ đó, ta có các địa chỉ Port của 8255A đối với μP chủ là:

PA: FCh
PB: FDh
PC: FEh
CR: FFh





Hình 4.19 – Thông tin hai chiều giữa μ P chủ và μ P tớ



Port A được sử dụng ở Mode 2 dùng 4 tín hiệu từ Port C. μP chủ kiểm tra các tín hiệu \overline{ACK} và \overline{STB} bằng cách đọc các bit trạng thái \overline{OBF} và IBF ở Port C.

Hai tín hiệu bắt tay khác - \overline{OBF} và IBF – được nối tương ứng với các bit D7 và D0 của data bus của μP tới thông qua bộ đệm 3 trạng thái 74LS365. Logic giải mã cho các đường tín hiệu tại Port C chính là bộ giải mã 3 sang 8 74LS138. Giả sử các đường logic không sử dụng (A3 và A4) ở mức 0, 8 đường ra của bộ giải mã sẽ cho phép vùng địa chỉ 80h ÷ 87h (Bảng 4.6). Hai đường ra của bộ giải mã được kết hợp với tín hiệu điều khiển \overline{IOR} để tạo ra 2 xung chọn thiết bị nhận (85h và 87h). Xung chọn thiết bị nhận 87h được dùng để đọc trạng thái ở các đường dữ liệu D7 và D0. Đường giải mã có địa chỉ 80h được kết hợp với \overline{IOW} để tạo tín hiệu \overline{STB} .

Bảng 4.6:

A7	A6	A5	A4	A3	A2	A1	A0	Chân giải mã	Địa chỉ hex
1	0	0	0	0	0	0	0	Y0	80h
					0	0	1	Y1	81h
					0	1	0	Y2	82h
					0	1	1	Y3	83h
					1	0	0	Y4	84h
					1	0	1	Y5	85h
					1	1	0	Y6	86h
					1	1	1	Y7	87h

❖ Từ điều khiển mode 2:

D7	D6	D5	D4	D3	D2	D1	D0	
1	1	0	0	0	0	0	0	= C0h
I/O	Mode 2			Không sử dụng				

❖ Từ trạng thái mode 2:

Trạng thái của hoạt động I/O ở Mode 2 có thể kiểm tra bằng cách đọc nội dung Port C.

D7	D6	D5	D4	D3	D2	D1	D0
\overline{OBF}_A	$INTE_1$	$IBFA$	$INTE_2$	$INTRA$	X	X	X

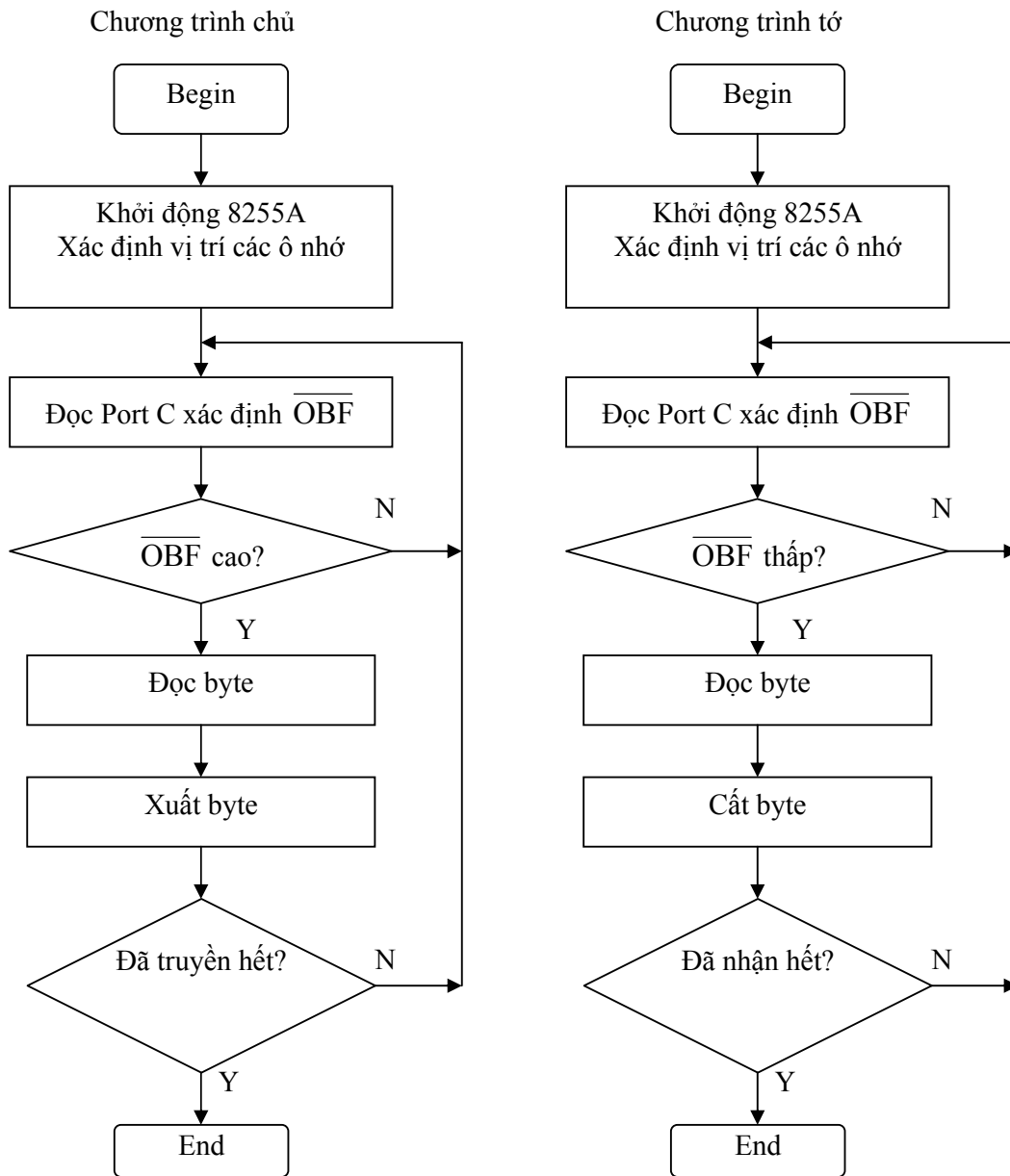
Trạng thái của tín hiệu \overline{OBF} được kiểm tra bằng cách đọc bit D7 và trạng thái của IBF kiểm tra bằng bit D0.

❖ Các tác vụ Đọc và Ghi của μP tới:

Một byte dữ liệu có thể được đọc bởi μP tới từ Port A bằng cách gửi một xung chọn thiết bị tác động mức thấp đến tín hiệu \overline{ACK} , không cần xây dựng Port nhập. Tương tự, một byte dữ liệu có thể được ghi vào μP bằng cách đưa tín hiệu \overline{STB} xuống thấp.



❖ **Lưu đồ giải thuật:**



❖ **Chương trình:**

➤ **Đoạn chương trình chủ: (Master program)**

```

MOV     SP,stack1
MOV     SI,master           ; Địa chỉ các byte cần xuất
MOV     CX,byte_no        ; Số byte cần xuất
MOV     AL,0C0h           ; Từ điều khiển
MOV     DX,0FFh           ; Địa chỉ thanh ghi điều khiển
OUT     DX,AL
    
```



```

next:   MOV     DX,0FEh    ; Địa chỉ Port C
wait:   IN      AL,DX     ; Đọc vào từ Port C
        AND    AL,80h    ; Kiểm tra  $\overline{OBF}$ 
        JNE   wait       ; Chờ đến khi  $\overline{OBF} = 0$ 
        LODSB                ; Đọc byte
        MOV   DX,0FCh    ; Xuất byte vừa đọc
        OUT  DX,AL       ; ra Port A
        LOOP next        ; Nếu còn byte truyền thì tiếp tục
        END

```

➤ **Đoạn chương trình tớ: (Slave program)**

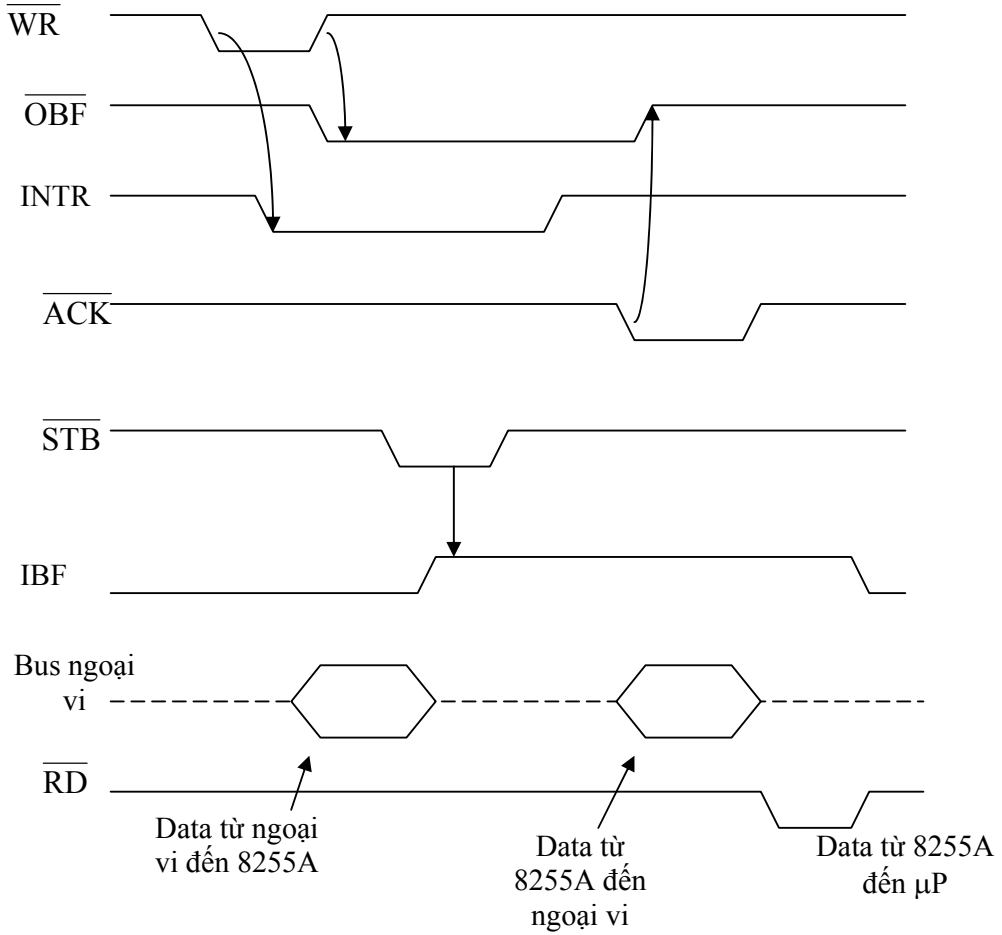
```

next:   MOV     ES,stack2
wait:   MOV     DI,slave   ; Địa chỉ các byte sẽ lưu
        MOV     CX,byte_no ; Số byte cần nhận
        MOV     DX,87h
        IN      AL,DX     ; Đọc  $\overline{OBF}$ 
        AND    AL,80h    ; Kiểm tra  $\overline{OBF}$ 
        JE    wait       ; Chờ đến khi  $\overline{OBF} = 1$ 
        MOV     DX,85h
        IN      AL,DX     ; Đọc dữ liệu
        STOSB                ; Cát vào ô nhớ
        LOOP   next        ; Nếu còn byte truyền thì tiếp tục
        END

```

- Ta thấy rằng cả hai chương trình sẽ kiểm tra trạng thái \overline{OBF} . Chương trình chủ đợi cho đến khi \overline{OBF} lên mức cao sẽ ghi một byte vào Port A. Ngược lại, chương trình tớ đợi cho đến khi \overline{OBF} xuống mức thấp thì sẽ đọc dữ liệu.
- Khi μP chủ ghi một byte dữ liệu, nó sẽ chốt tại Port A và byte dữ liệu được đặt trên data bus của μP tớ khi \overline{ACK} xuống mức thấp.
- Hai chương trình trên chỉ cho phép truyền một khối dữ liệu từ μP chủ đến μP tớ nhưng không thể truyền ngược lại. Để chuyển một khối dữ liệu từ μP tớ đến μP chủ, cần phải đọc tín hiệu IBF. μP chủ đợi cho đến khi IBF = 1 thì sẽ đọc một byte dữ liệu còn μP tớ đợi cho đến khi IBF = 0 thì ghi một byte dữ liệu.
- Giảm đồ thời gian ở hình 3.4.16 cho thấy tín hiệu INTR dùng để truyền dữ liệu bằng ngắt. Trong ví dụ này, ta không sử dụng ngắt.





Hình 4.20 – Giải đồ thời gian ở Mode 2

