

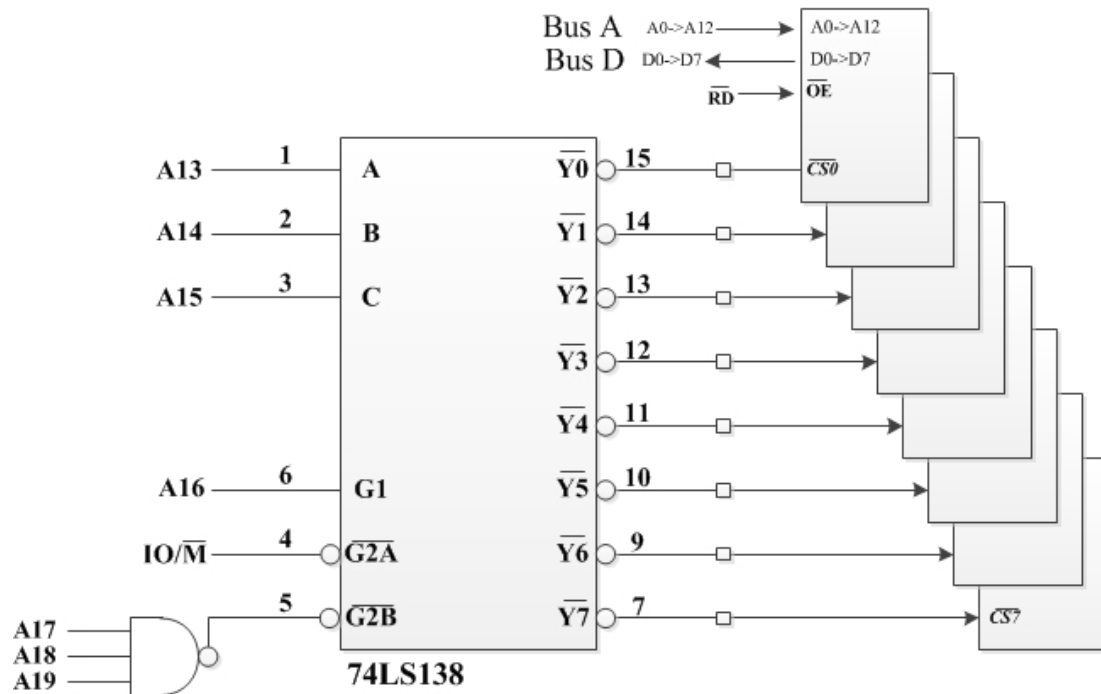


Ghép nối bộ nhớ - Tổng hợp kiến thức của thầy Dũng vào kì 2023.2

Kỹ thuật vi xử lý (Trường Đại học Bách khoa Hà Nội)



Scan to open on Studocu



Hình 4.11: Sơ đồ ghép nối vi xử lý 8088 với EPROM 2764 tạo bộ nhớ ROM 64KB

Chú ý: Cũng giống như IC giải mã 74LS138, IC giải mã 74LS139 cũng dùng để giải mã tín hiệu địa chỉ. Tuy nhiên khác với IC giải mã 74LS138, 74LS139 chỉ sử dụng 2 ngõ vào giải mã cho 4 ngõ ra tương ứng trong đó tại một thời điểm chỉ có một ngõ ra bằng 0, các ngõ ra khác bằng 1. Các trạng thái địa chỉ giải mã và sơ đồ khối của IC 74LS139 được cho lần lượt trong bảng 4.8 và hình 4.12.

Bảng 4.8: Bảng trạng thái của IC giải mã 74LS139

		Bảng trạng thái của IC giải mã 74LS139					
	A	B	G	Y0	Y1	Y2	Y3
x	x	1	1	1	1	1	1
0	0	0	0	0	1	1	1
0	1	0	1	0	1	1	1
1	0	0	1	1	0	1	1
1	1	0	1	1	1	0	1

Hình 4.12: Sơ đồ khối của IC giải mã 74LS139

4.1.3. Giải mã địa chỉ dùng PROM

Ngoài việc sử dụng các cổng logic hay các IC giải mã chuyên dụng, ta cũng có thể dùng PROM như là một bộ giải mã. Ví dụ sau đây sẽ minh họa rõ hơn về điều này.

Ví dụ: Ghép nối EPROM 2764 với vi xử lý 8088 để tạo thành bộ nhớ 64KB tại địa chỉ F0000h. Sử dụng PROM loại TPB28L42 với dung lượng 512 byte để giải mã.

Giải:

Do đây là ví dụ đã nêu ở phần trên nhưng sử dụng IC giải mã 74LS138 nên sẽ không đề cập đến việc giải mã địa chỉ cụ thể cho từng IC EPROM 2764.

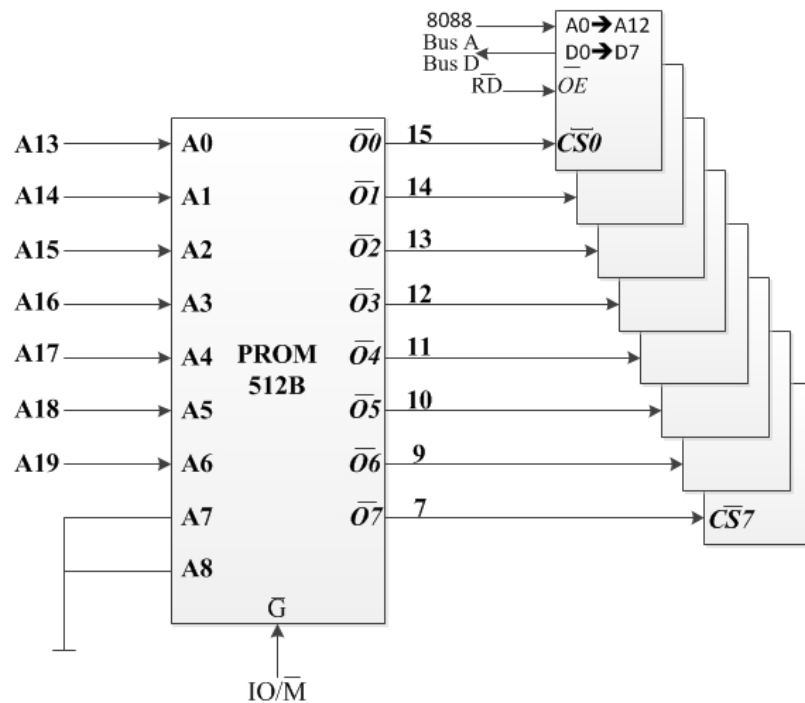
PROM TPB28L42 có dung lượng 512 byte = $2^9 \times 8\text{bit}$. Từ A_0 đến A_8 là các đường địa chỉ, \bar{O}_0 đến \bar{O}_7 là các đường dữ liệu tương ứng tích cực ở mức thấp.

Bảng 4.9: Mẫu các bit để ghi vào PROM cho bộ giải mã ở hình 4.13

Các đầu vào										Các đầu ra							
\bar{G}	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	\bar{O}_0	\bar{O}_1	\bar{O}_2	\bar{O}_3	\bar{O}_4	\bar{O}_5	\bar{O}_6	\bar{O}_7
0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	1
0	0	0	1	1	1	1	0	0	1	1	0	1	1	1	1	1	1
0	0	0	1	1	1	1	0	1	0	1	1	0	1	1	1	1	1
0	0	0	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1
0	0	0	1	1	1	1	1	0	0	1	1	1	1	0	1	1	1
0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1
0	0	0	1	1	1	1	1	1	0	1	1	1	1	1	1	0	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	Các địa chỉ khác									1	1	1	1	1	1	1	1

Theo bảng 4.9 ta thấy chỉ sử dụng hết có 8 byte trong tổng số 512 byte nhớ của PROM.

Mạch giải mã cho bộ nhớ EPROM 64KB dùng PROM được thể hiện trên hình 4.13.



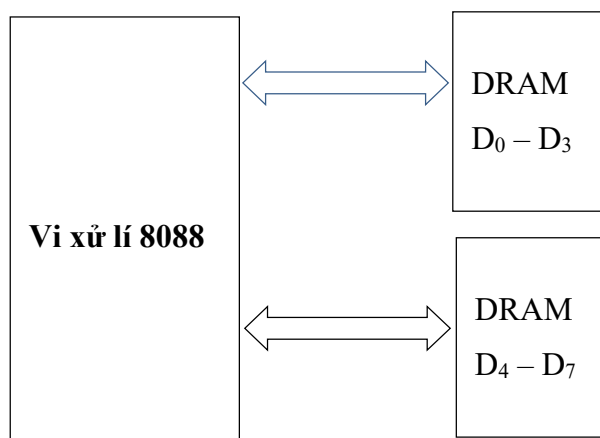
Hình 4.13: Sơ đồ ghép nối vi xử lý 8088 với EPROM 2764 tạo bộ nhớ ROM 64KB sử dụng PROM 512B giải mã

So với việc sử dụng PROM với IC giải mã 74LS138 thì không phải sử dụng thêm các mạch phụ, sẽ tiết kiệm đáng kể kích thước vật lý của bộ giải mã.

4.2. Ghép nối vi xử lý 8086 với bộ nhớ

4.2.1. Nguyên lý ghép nối với IC nhớ

Khi làm việc với 8088, đôi khi gặp phải trường hợp ghép nối 8088 với DRAM họ 44xxx có 4 bit dữ liệu. Để giải quyết vấn đề này, vi xử lý thực hiện ghép nối song song theo kiểu dữ liệu bank thấp 4 bit và bank cao 4 bit dữ liệu như trong hình 4.14.

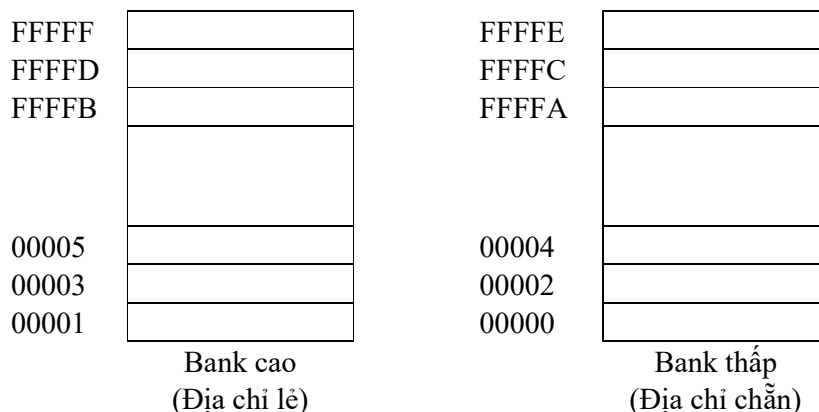


Hình 4.14: Minh họa vi xử lý 8088 ghép nối với DRAM 44xxx 4 bit

Khác với 8088, vi xử lý 8086 của hãng Intel có khả năng xử lý tối đa 16 bit dữ liệu. Khi ghép nối trao đổi dữ liệu với bộ nhớ phần dữ liệu này được chia ra thành 2 bank như sau:

- Từ D0 đến D7 được gọi là dữ liệu phần thấp: bank thấp.
- Từ D8 đến D15 được gọi là dữ liệu phần cao: bank cao.

Hình 4.15 minh họa chia hai bank thấp và cao của vi xử lý 8086.



Hình 4.15: Bank thấp, bank cao và địa chỉ tương ứng

Bên cạnh đó một số tín hiệu điều khiển ghép nối trao đổi dữ liệu của vi xử lý 8086 khác với vi xử lý 8088 là:

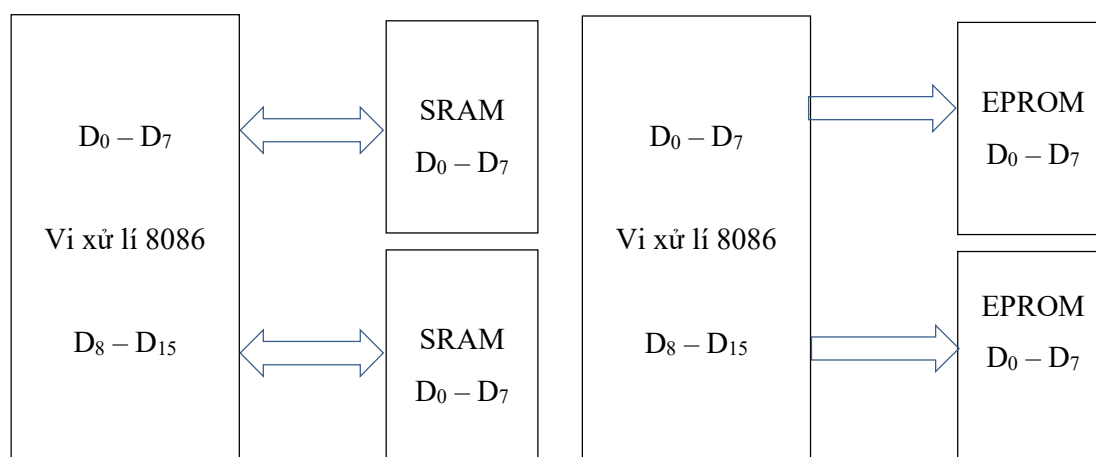
- **M/\overline{IO}** : Tín hiệu điều khiển ghép nối với bộ nhớ hoặc cổng vào ra. Khi ghép nối với bộ nhớ, tín hiệu này bằng 1.
- **\overline{BHE}** : Bit High Enable – Tín hiệu điều khiển chọn dữ liệu phần cao làm việc.
- **$A0$** : Vi xử lý 8086 sử dụng tín hiệu địa chỉ $A0$ làm tín hiệu điều khiển chọn dữ liệu phần thấp làm việc.

Mối quan hệ giữa \overline{BHE} và $A0$ được biểu diễn trong bảng 4.10.

Bảng 4.10: Quan hệ giữa \overline{BHE} và $A0$

\overline{BHE}	$A0$	Chức năng
0	0	Chọn cả hai bank
0	1	Chọn bank cao
1	0	Chọn bank thấp
1	1	Không chọn bank nào

Dựa theo nguyên lý, người ta thực hiện việc ghép nối 8086 với EPROM (8 bit dữ liệu) và SRAM (8 bit dữ liệu) theo kiểu ghép bank. Ta có thể hình dung dựa theo nguyên lý cho trên hình 4.16.



Hình 4.16: Tổ chức ghép nối bộ nhớ của vi xử lý 8086

Vi xử lý 8086 mang đầy đủ các cơ sở giải mã địa chỉ của 8088 như: giải mã bằng cổng NAND; giải mã bằng Decoder 74LS139, 74LS138, 74LS154; giải mã bằng PROM.

4.2.2. Ghép nối 8086 với IC nhớ 8 bit dữ liệu

Ví dụ: Ghép nối 8086 với IC nhớ EPROM 2716 để được vùng bộ nhớ FF000 đến FFFFFh.

Giải:

IC nhớ 2716 : EPROM (2Kx8) chỉ có chế độ đọc dữ liệu.

Có 11 đường địa chỉ A0 – A10 nối với A1-A11 của vi xử lý 8086

Có 8 đường dữ liệu D0 – D7

Dung lượng = địa chỉ cuối – địa chỉ đầu +1

$$= \text{FFFFh} - \text{FF000h} + 1 = 1000\text{H} (2 \times 2^{11})$$

$$\text{Số IC cần} = 2 \times 2^{11} : 2^{11} = 2 \text{ IC}$$

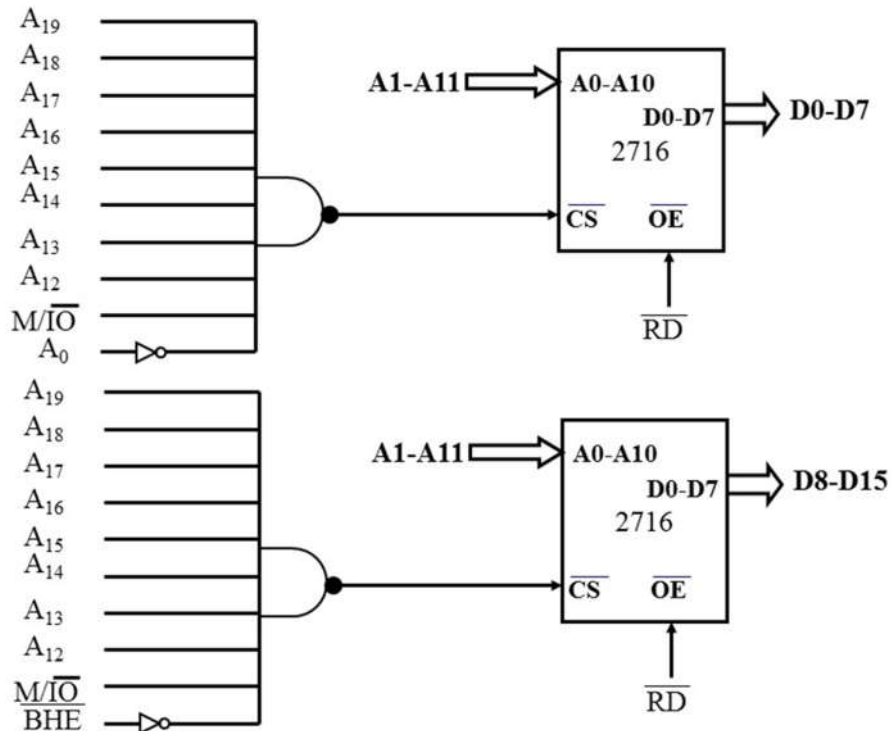
Có 1 bank chẵn và 1 bank lẻ được biểu diễn như sau:

A ₁₉	A ₁₈	A ₁₇	A ₁₆	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

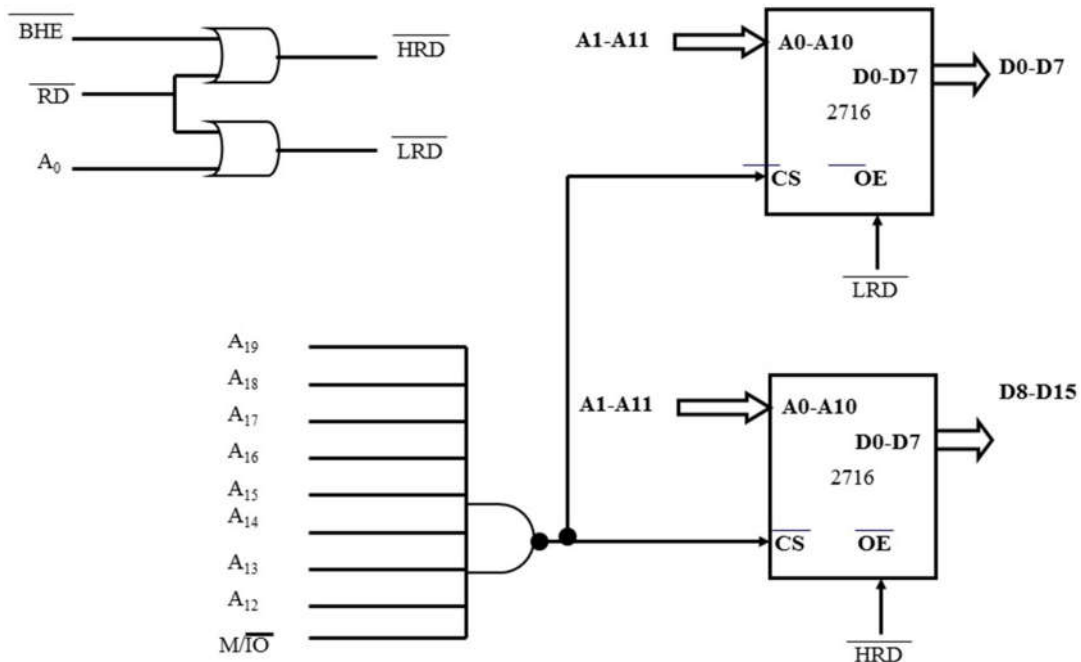
Địa chỉ chẵn IC (bank thấp): FF000 - FFFFEh

Địa chỉ lẻ IC (bank cao): FF001 – FFFFFh

Sơ đồ ghép nối được biểu diễn theo hai cách lần lượt như trong các hình 4.17 và 4.18 dưới đây.



Hình 4.17: Ví dụ ghép nối vi xử lý 8086 với bộ nhớ điều khiển bằng A0 và \overline{BHE}



Hình 4.18: Ví dụ ghép nối vi xử lý 8086 với bộ nhớ điều khiển bằng các tín hiệu điều khiển \overline{HRD} và \overline{LRD}

Các tín hiệu điều khiển \overline{HRD} và \overline{LRD} (high và low read enable) được tạo ra từ các cổng logic OR kết hợp giữa tín hiệu điều khiển đọc \overline{RD} và A0 hoặc \overline{BHE} như trong hình 4.18.

Tương tự, các tín hiệu điều khiển \overline{HWR} và \overline{LWR} (high và low write enable) được tạo ra từ các cổng logic OR kết hợp giữa tín hiệu điều khiển đọc \overline{WR} và A0 hoặc \overline{BHE} .

Ví dụ: Ghép nối 8086 với IC nhớ 62256 để được vùng nhớ có địa chỉ bắt đầu là 40000H và địa chỉ kết thúc là 6FFFFH

Giải:

IC nhớ 62256 : SRAM (32Kx8) có cả chế độ đọc và chế độ ghi.

Có 15 đường địa chỉ A0 – A14 nối với A1-A15 của vi xử lý 8086

Có 8 đường dữ liệu D0 – D7

Dung lượng = địa chỉ cuối – địa chỉ đầu + 1

$$= 6FFFFh - 40000h + 1 = 30000h \quad (3 \times 2^{16})$$

$$\text{Số IC cần} = 3 \times 2^{16} : 2^{15} = 6 \text{ IC}$$

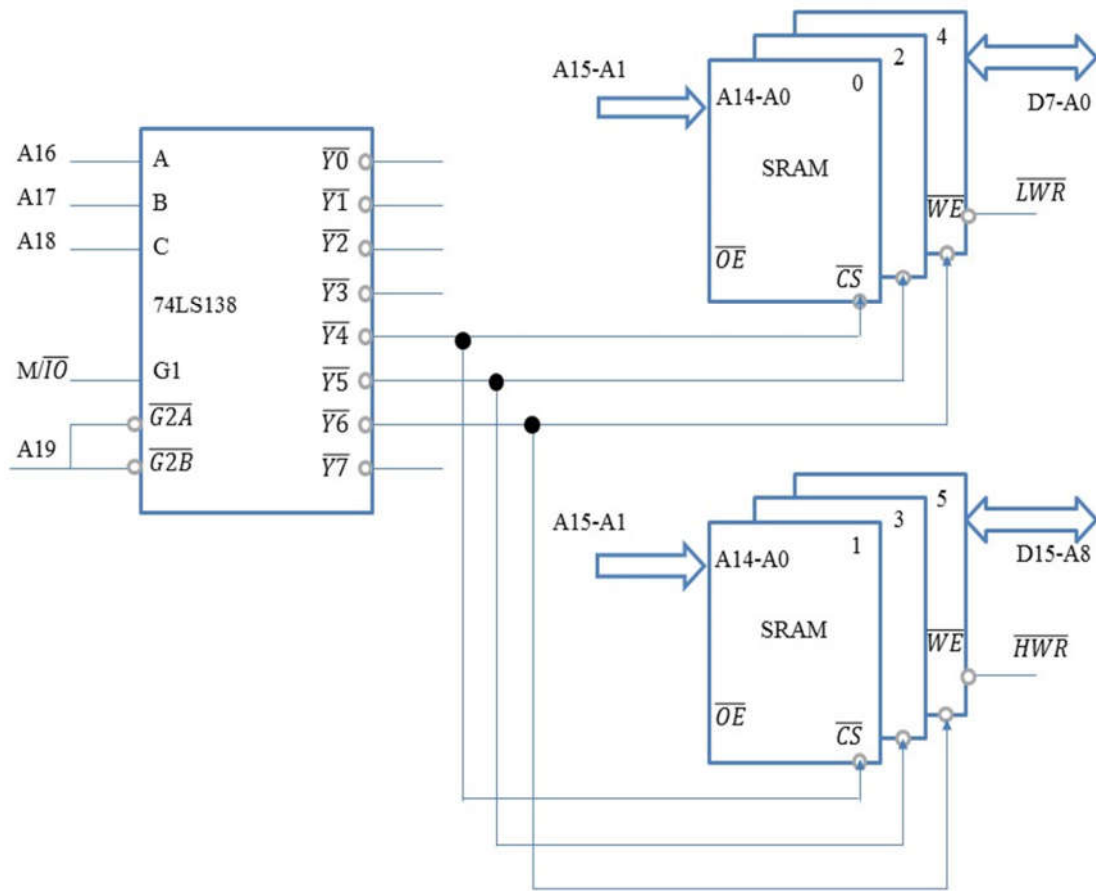
Có 3 bank chẵn và 3 bank lẻ được biểu diễn trong bảng 4.11.

Địa chỉ chẵn của các IC		Địa chỉ lẻ của các IC	
IC0	40000h – 4FFFEh	IC1	40001h – 4FFFFh
IC2	50000h – 5FFFEh	IC3	50001h – 5FFFFh
IC4	60000h – 6FFFEh	IC5	60001h – 6FFFFh

Bảng 4.11: Giải địa chỉ của từng IC nhớ 62256 (bank cao và bank thấp)

A19	A18	A17	A16	A15.....A1	A0	
0	1	0	0	0.....0	1	IC1
0	1	0	0	1.....1	1	
0	1	0	1	0.....0	1	IC3
0	1	0	1	1.....1	1	
0	1	1	0	0.....0	1	IC5
0	1	1	0	1.....1	1	
0	1	0	0	0.....0	0	IC0
0	1	0	0	1.....1	0	
0	1	0	1	0.....0	0	IC2
0	1	0	1	1.....1	0	
0	1	1	0	0.....0	0	IC4
0	1	1	0	1.....1	0	

Sơ đồ ghép nối 6 IC SRAM 32k x 8 được trình bày trong hình 4.19.



Hình 4.19: Sơ đồ ghép nối vi xử lý 8086 với 6 IC SRAM 32k x 8 với khoảng địa chỉ từ 40000 đến 6FFFFh

V. GHÉP NỐI 8088 VỚI THIẾT BỊ NGOẠI VI

5.1. Các phương pháp điều khiển vào ra dữ liệu

Đối với vi xử lý 8088 (hay cả họ 80x86 nói chung) có hai cách phối ghép CPU với các thiết bị ngoại vi (các cổng vào ra, I/O):

- Thiết bị ngoại vi có địa chỉ tách rời với bộ nhớ
- Thiết bị ngoại vi có địa chỉ chung với bộ nhớ

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

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

- Bộ nhớ: $\text{IO}/\overline{\text{M}} = 0$, sử dụng lệnh MOV.
- Thiết bị ngoại vi: $\text{IO}/\overline{\text{M}} = 1$, sử dụng lệnh IN (nhập dữ liệu) hoặc OUT (xuất dữ liệu).

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

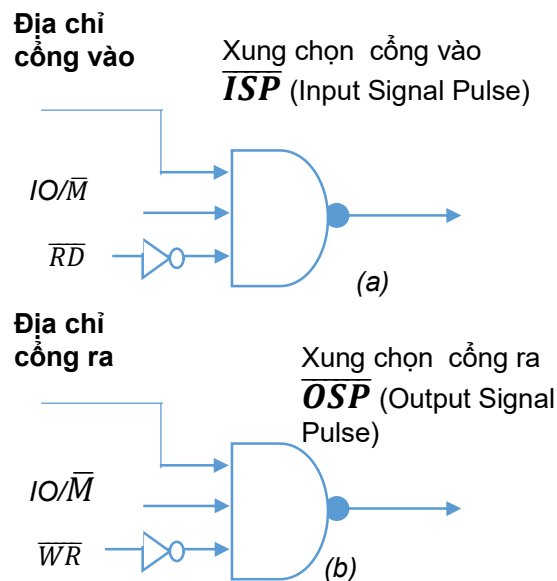
Trong kiểu giao tiếp này thiết bị ngoại vi sẽ chiếm một vùng nào đó trong không gian địa chỉ 1MB của bộ nhớ mà vi xử lý 8088 quản lý. Thiết bị ngoại vi muốn trao đổi dữ liệu với vi xử lý tức là trao đổi dữ liệu qua bộ nhớ nên do đó phải sử dụng lệnh MOV để thực hiện trao đổi. Lúc này tín hiệu IO/\overline{M} dùng để ghép nối với bộ nhớ nên $IO/\overline{M} = 0$.

5.2. Giải mã địa chỉ cho thiết bị vào ra

Giải mã địa chỉ công vào ra cũng giống như giải mã địa chỉ cho bộ nhớ. Thông thường các cổng có địa chỉ 8 bit từ A_0 đến A_7 . Tuy nhiên trong một số hệ vi xử lý các cổng sẽ có địa chỉ 16 bit từ A_0 đến A_{15} .

Có thể sử dụng mạch NAND để tạo tín hiệu chọn cổng nhưng mạch này chỉ có thể giải mã cho 1 cổng vào hoặc ra. 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.

Hình 4.20 và 4.21 minh họa lần lượt mạch giải mã địa chỉ đơn giản cho thiết bị ngoại vi bằng cổng NAND và bằng IC giải mã 74LS138.



Hình 4.20: Giải mã địa chỉ cổng đơn giản bằng NAND