

CHƯƠNG 6 PHỐI GHÉP 8088 VỚI BỘ NHỚ

I - CÁC LOẠI BỘ NHỚ BÁN DẪN

1. Cấu trúc chung của một vi mạch nhớ

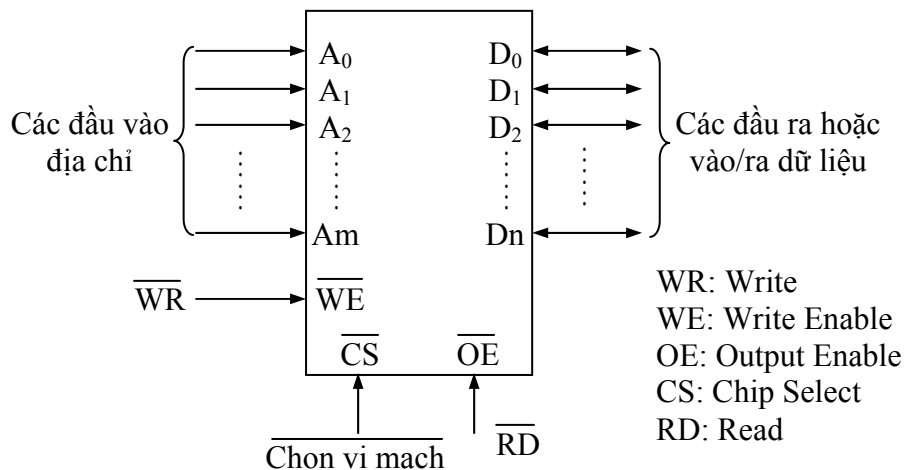
Bộ nhớ trung tâm của hầu hết các hệ vi xử lý có hai loại: bộ nhớ chỉ đọc (ROM) và bộ nhớ truy cập ngẫu nhiên hay bộ nhớ đọc/ghi (RAM). ROM dùng để chứa chương trình hệ thống và các dữ liệu cố định. RAM chứa dữ liệu tạm thời và các phần mềm ứng dụng.

Trước khi nghiên cứu cách nối ghép bộ nhớ với bộ vi xử lý chúng ta hãy tìm hiểu qua về các loại bộ nhớ bán dẫn thường dùng:

+ Bộ nhớ cố định ROM (Read Only Memory): là bộ nhớ có nội dung ghi sẵn chỉ để đọc ra, nội dung của ROM không bị mất khi mất nguồn điện nuôi. Bộ nhớ ROM có nhiều loại như EPROM, PROM, EEPROM...

+ Bộ nhớ không cố định RAM (Random Access Memory): là bộ nhớ có thể đọc và ghi, thông tin chứa trong RAM bị mất khi mất nguồn điện nuôi. Bộ nhớ RAM chia thành 2 loại là RAM tĩnh (SRAM, Static RAM) và RAM động (DRAM, Dynamic RAM). Ở SRAM mỗi phần tử nhớ là một mạch lật hai trạng thái ổn định, còn ở DRAM mỗi phần tử nhớ là một tụ điện rất nhỏ được chế tạo bằng công nghệ MOS.

Một bộ nhớ thường được tạo nên từ nhiều vi mạch nhớ. Một vi mạch nhớ thường có dạng cấu trúc tiêu biểu như sau:



Các tín hiệu của một vi mạch nhớ

- Nhóm tín hiệu địa chỉ

Tất cả các vi mạch nhớ đều có đầu vào cho các tín hiệu địa chỉ để chọn ra một ô nhớ (từ nhớ) cụ thể để đọc/ghi dữ liệu. Các đầu vào này có tên từ A_0 đến A_m . Số chân địa chỉ có liên quan đến dung lượng của vi mạch nhớ. Một vi mạch nhớ có m chân địa chỉ sẽ có dung lượng là 2^m ô nhớ. Từ số chân địa chỉ ta biết được dung lượng ô nhớ và ngược lại, từ dung lượng ô nhớ ta suy ra được số chân địa chỉ của vi mạch nhớ. Ví dụ, một vi mạch nhớ có 10 chân địa chỉ $A_0 - A_9$ sẽ có $2^{10} = 1\text{ K}$ ô nhớ, một vi mạch nhớ có dung lượng 2 K ô nhớ sẽ có số chân địa chỉ là 11.

Nếu vi mạch nhớ có dung lượng 1 K ($2^{10} = 1024 = 400\text{H}$) ô nhớ với địa chỉ ô nhớ đầu tiên là 10000H thì địa chỉ của ô nhớ cuối cùng sẽ là 10400H - 1H = 103FFH. Nếu vi mạch nhớ có dung lượng 4 K ($2^{12} = 1000\text{H}$) ô nhớ với địa chỉ ô nhớ đầu tiên là 14000H thì địa chỉ của ô nhớ cuối cùng sẽ là 15000H - 1H = 14FFFH.

- Nhóm tín hiệu dữ liệu

Tất cả các vi mạch nhớ đều có các chân dữ liệu ra (đối với vi mạch nhớ ROM) hoặc chân dữ liệu vào/ra (đối với vi mạch nhớ RAM). Các chân dữ liệu thường có tên từ D_0 đến D_n (thường là từ D_0 đến D_7). Số chân dữ liệu quyết định độ dài ô nhớ của vi mạch. Ví dụ vi mạch nhớ có 8 chân dữ liệu thì các ô nhớ trong vi mạch nhớ có độ dài 8 bit. Thông thường người ta hay nói dung lượng kèm theo độ dài ô nhớ, ví dụ nói vi mạch nhớ có dung lượng 1 KB, nghĩa là vi mạch nhớ có số ô nhớ 1 K, mỗi ô nhớ có độ dài 1 byte (8 bit). Trong các catalog của vi mạch nhớ thường ghi số ô nhớ nhân với số bit trên một ô nhớ, ví dụ: một vi mạch nhớ có dung lượng 1K, mỗi ô nhớ có độ dài 8 bit thường được ghi là $1K \times 8$.

- Nhóm tín hiệu chọn vi mạch

Mỗi vi mạch nhớ thường có một (có thể hơn) đầu vào để chọn vi mạch nhớ đọc/ghi dữ liệu. Loại đầu vào này gọi là đầu vào chọn chip (Chip Select, \overline{CS}) hay cho phép chip làm việc (Chip Enable, \overline{CE}). Ở RAM tín hiệu này thường gọi là \overline{CS} , còn ở ROM tín hiệu này gọi là \overline{CE} . Khi một vi mạch nhớ không được chọn thì các đầu ra dữ liệu của nó bị treo (ở trạng thái trở kháng cao).

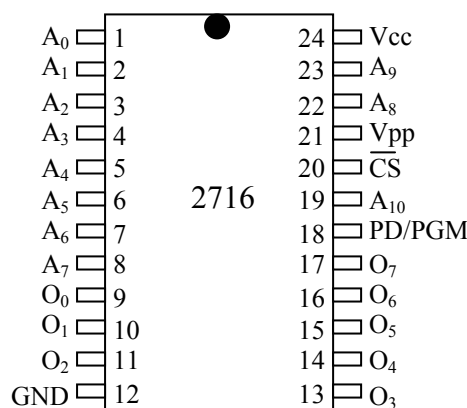
- Nhóm tín hiệu điều khiển

Các vi mạch nhớ đều có một hoặc nhiều đầu vào điều khiển. Các mạch nhớ ROM thường có một đầu vào điều khiển \overline{OE} (Output Enable) hoặc \overline{G} (Gate) để cho phép dữ liệu được đưa ra trên các chân dữ liệu. Các mạch nhớ RAM có một hoặc hai đầu vào điều khiển. Nếu một đầu vào điều khiển thì có tên là R/\overline{W} , nó điều khiển cả quá trình đọc và ghi. Nếu có hai đầu vào điều khiển thì là \overline{WE} (Write Enable) để điều khiển ghi và \overline{OE} để điều khiển đọc. Hai tín hiệu này không được phép cùng tác động (active) mà phải ngược pha nhau.

Một thông số đặc trưng của bộ nhớ là thời gian truy nhập t_{acc} , được định nghĩa là thời gian kể từ khi có xung địa chỉ trên các chân địa chỉ cho đến khi có dữ liệu ra ổn định trên các chân dữ liệu. Thời gian truy nhập của bộ nhớ phụ thuộc nhiều vào công nghệ chế tạo nên nó. Các bộ nhớ làm bằng công nghệ lưỡng cực có thời gian truy nhập nhỏ (10 - 30 ns) còn các bộ nhớ làm bằng công nghệ MOS có thời gian truy nhập lớn hơn nhiều (cỡ 150 ns hoặc lớn hơn nữa).

Sau đây chúng ta cùng tìm hiểu một số loại bộ nhớ thông dụng.

2. Bộ nhớ EPROM (Erasable Programmable Read Only Memory)



Chân	PD/PGM	\overline{CS}	Vpp	Vcc	$D_0 - D_7$
Chế độ					
Đọc	0	0	+5	+5	D_{out}
Không chọn chip	x	1	+5	+5	TK cao
Duy trì	1	x	+5	+5	TK cao
Ghi	xung f	1	+25	+5	D_{in}
Ktra ghi	0	0	+25	+5	D_{out}
Cắm ghi	0	1	+25	+5	TK cao

$A_0 - A_{10}$: Address

PD/PGM: Power Down/Program (\overline{OE} /PGM)

\overline{CS} : Chip Select (\overline{CE})

$O_0 - O_7$: Outputs ($D_0 - D_7$)

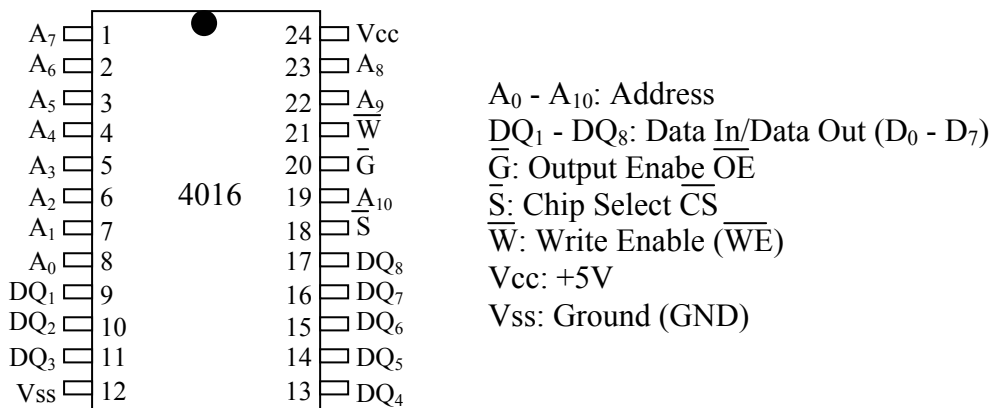
Sơ đồ chân của EPROM 2716 $2K \times 8$

Là bộ nhớ ROM có thể lập trình được bằng xung điện hoặc xóa bằng tia cực tím. Họ EPROM 27xxx có các loại sau: 2704 (512×8), 2708 (1K×8), 2716 (2K×8), 2732 (4K×8), 2764 (8K×8), 27128 (16K×8), 27256 (32K×8), 27512 (64K×8) và 271024 (128K×8).

Vì mạch nhớ EPROM 2716 có thời gian truy nhập $t_{acc} = 450 \text{ ns}$ nên để ghép nối với 8088-5MHz nó cần thêm chu kỳ đợi. Trong chế độ duy trì, công suất tiêu thụ của EPROM 2716 giảm được 75% so với công suất tiêu thụ khi nó ở chế độ tích cực. Ở trên là sơ đồ chân và các chế độ hoạt động của 2716.

3. Bộ nhớ SRAM (Static RAM)

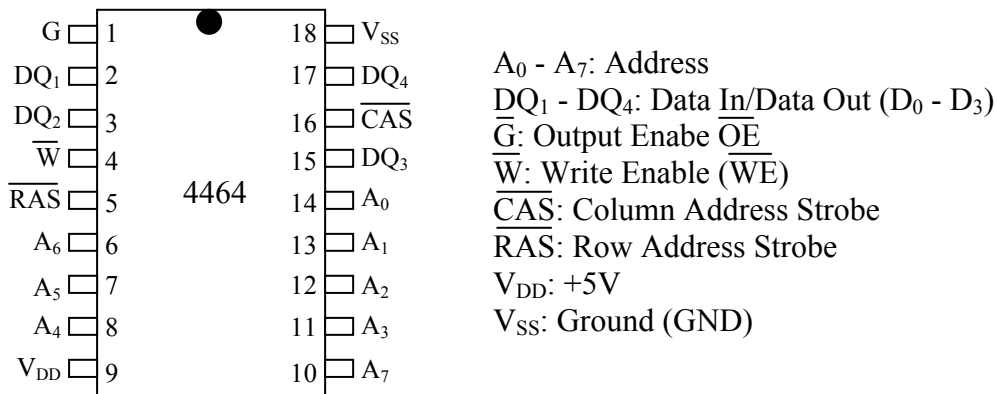
Bộ nhớ SRAM vẫn giữ được thông tin chừng nào còn được cấp điện. Dưới đây là sơ đồ chân của SRAM 4016 2K×8, đây là bộ nhớ SRAM điển hình, nó có 11 chân địa chỉ và 8 chân dữ liệu vào/ra.



Sơ đồ chân của SRAM 4016 2K×8

4. Bộ nhớ DRAM (Dynamic RAM)

Bộ nhớ DRAM lưu trữ thông tin bằng cách nạp hay không nạp điện tích lên các tụ điện công nghệ MOS. Vì vậy mỗi phần tử nhớ của bộ nhớ DRAM cần được làm tươi lại (bằng cách ghi hay đọc phần tử đó) sau một quãng thời gian cỡ 15,6 μs , nếu không điện tích trên các tụ điện (mỗi tụ điện là một phần tử nhớ) sẽ bị tiêu tán và dẫn đến mất thông tin. Các mạch DRAM cần có các mạch logic phụ để đảm bảo việc làm tươi và vì thế việc phối ghép nó với bộ vi xử lý là rất phức tạp. Bù lại nhược điểm này các bộ nhớ DRAM lại có ưu điểm là có thể chế tạo được một số lượng rất lớn các phần tử nhớ trên một diện tích. Số lượng phần tử nhớ lớn dẫn đến số chân địa chỉ cũng nhiều. Để giảm bớt số chân địa chỉ trên một vi mạch nhớ người ta chia địa chỉ ra hai nhóm: địa chỉ hàng và địa chỉ cột rồi ghép kênh chúng vào các chân địa chỉ của vi mạch nhớ, đồng thời cung cấp các tín hiệu cho phép chốt giữ riêng rẽ địa chỉ hàng và địa chỉ cột bên trong vi mạch nhớ.

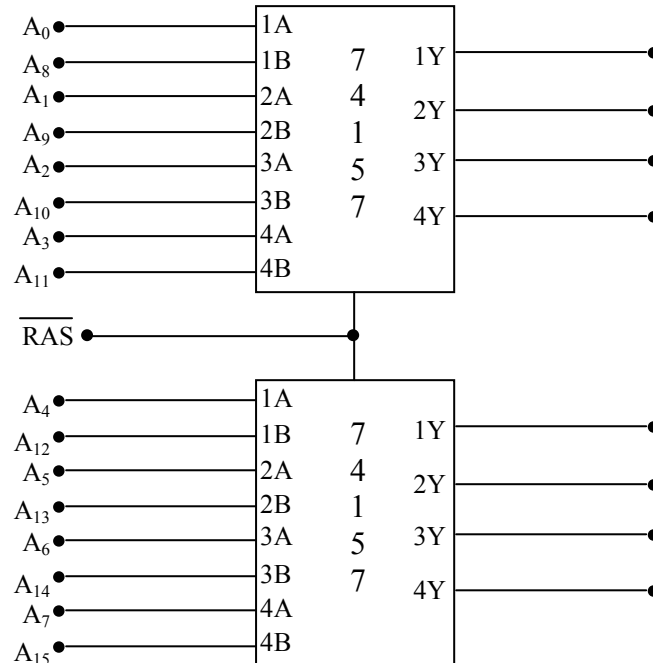


Sơ đồ chân của DRAM 4464 64K×4

Các bộ nhớ DRAM thường được chế tạo với độ dài ô nhớ là 1 hoặc 4 bit. Trên thực tế đã có các vi mạch nhớ DRAM $1M \times 1$, $4M \times 1$ và $16M \times 1$. Các vi mạch nhớ DRAM thường được đặt trên một bảng mạch nhỏ gọi là SIMM (Single Inline Memory Modules).

Ta thấy DRAM TMS4464 chỉ có 8 đầu vào địa chỉ trong khi 64 K ô nhớ cần tới 16 đầu vào địa chỉ. Bởi vậy người ta phải ép 16 bit địa chỉ vào 8 chân địa chỉ của vi mạch nhớ bằng cách ghép kênh. Đầu tiên 8 bit địa chỉ $A_0 - A_7$ đưa tới 8 chân địa chỉ và được chốt giữ bởi một bộ chốt hàng bên trong vi mạch khi có tín hiệu \overline{RAS} . Tiếp theo, 8 bit địa chỉ $A_8 - A_{15}$ được đưa đến 8 chân địa chỉ và lại được chốt giữ bởi một bộ chốt cột bên trong vi mạch khi có tín hiệu \overline{CAS} . 16 bit địa chỉ giữ trong các chốt địa chỉ hàng và cột cho ta địa chỉ của một ô nhớ 4 bit trong vi mạch nhớ.

Bộ ghép kênh địa chỉ cho DRAM TMS4464 như sau:



Khi $\overline{RAS} = 1$ các đầu vào B được nối tới các đầu ra Y của bộ ghép kênh, còn khi $\overline{RAS} = 0$ các đầu vào A được nối tới các đầu ra Y.

II - GIẢI MÃ ĐỊA CHỈ CHO BỘ NHỚ

1. Tại sao cần giải mã địa chỉ cho bộ nhớ

Khi ghép nối các vi mạch nhớ với bộ vi xử lý cần giải mã địa chỉ do bộ vi xử lý đưa ra để chọn một phần nhớ cụ thể trong toàn bộ không gian nhớ. Nếu không có bộ giải mã địa chỉ thì chỉ nối được một vi mạch nhớ với bộ vi xử lý.

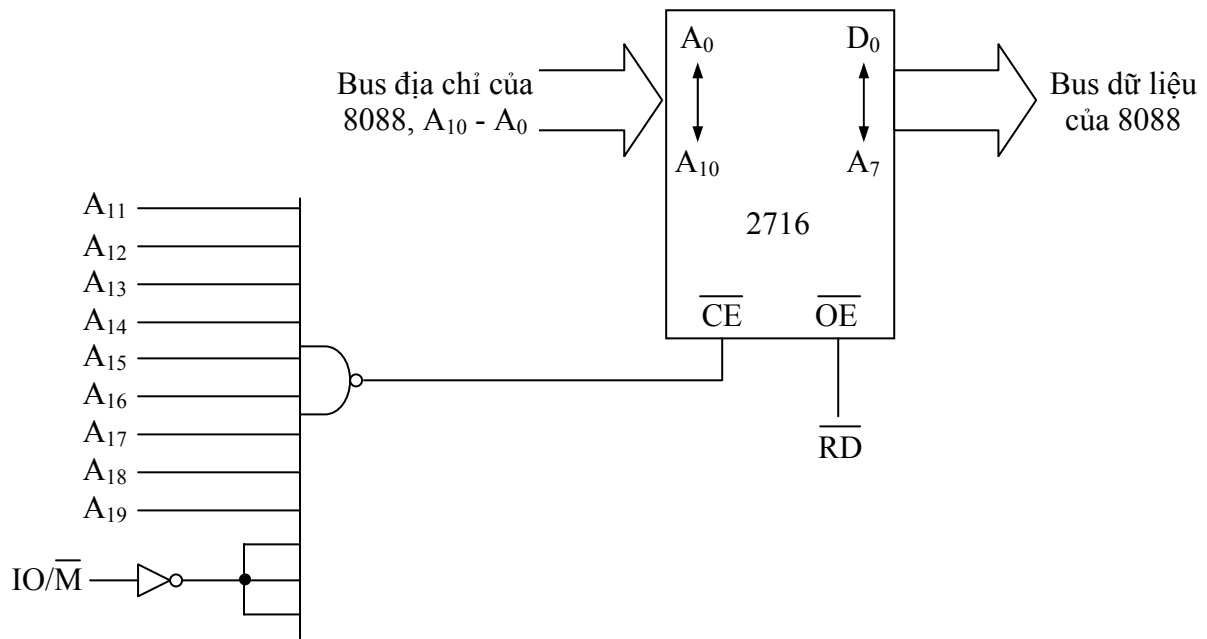
Giả sử ta cần nối vi mạch nhớ EPROM 2716 $2K \times 8$ tới bộ vi xử lý 8088. Ta biết số chân địa chỉ của 8088 là 20 còn số chân địa chỉ của 2726 là 11. Có nghĩa là khi đọc/ghi dữ liệu bộ vi xử lý 8088 sẽ gửi địa chỉ 20 bit trên bus địa chỉ. Nếu chỉ nối 11 chân địa chỉ của 8088 tới 2716 thì 8088 chỉ nhìn thấy 2KB bộ nhớ thay vì 1 MB như nó có thể. Vấn đề này được khắc phục bằng cách giải mã các chân không nối tới bộ vi xử lý. Sau đây chúng ta sẽ tìm hiểu một vài bộ giải mã.

2. Bộ giải mã cổng NAND

Đây là bộ giải mã đơn giản nhất, chỉ sử dụng một cổng NAND để giải mã.

Ví dụ ta sử dụng vi mạch nhớ EPROM 2716 $2K \times 8$ làm bộ nhớ. Khi nối ghép với bộ vi xử lý 8088, các chân từ $A_{10} - A_0$ của 8088 được nối tới các đầu vào địa chỉ từ $A_{10} - A_0$ của EPROM, các chân còn lại của 8088 ($A_{19} - A_{11}$) được nối tới đầu vào của bộ giải mã cổng

NAND. Bộ giải mã có nhiệm vụ chọn một phần 2 KB (vi mạch EPROM) trong 1 MB bộ nhớ mà 8088 có thể quản lý được.



Đầu ra của cổng NAND có logic 0 khi các chân địa chỉ A₁₉ - A₁₁ nối tới đầu vào của nó đều có logic 1 và tín hiệu điều khiển $\overline{IO/\overline{M}} = 0$. Đầu ra logic 0 của bộ giải mã cổng NAND được đưa tới đầu vào \overline{CE} của vi mạch nhớ EPROM và sẽ chọn vi mạch này. Nếu đầu vào \overline{OE} cũng có logic 0 thì dữ liệu sẽ được đọc từ EPROM. \overline{OE} được kích hoạt bằng tín hiệu \overline{RD} của 8088.

Vì chỉ khi các bit A₁₉ - A₁₁ bằng 1 thì EPROM mới được chọn, do đó địa chỉ 20 bit của các ô nhớ trong EPROM có dạng: 9 bit bên trái bằng 1 còn 11 bit bên phải tùy ý.

1111 1111 1XXX XXXX XXXX

Từ đây ta suy ra khoảng địa chỉ dành cho EPROM là từ

1111 1111 1000 0000 0000 = FF800H

đến

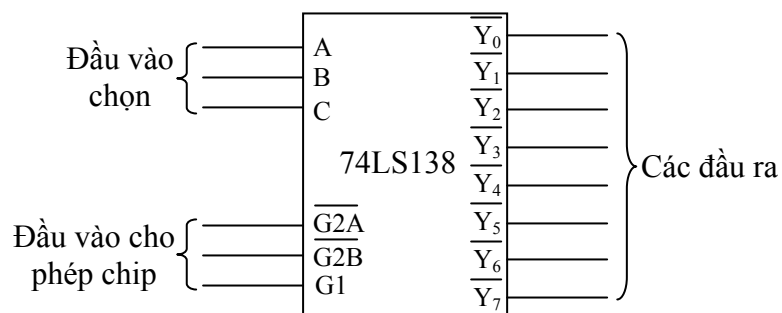
1111 1111 1111 1111 1111 = FFFFFH

Trong thực tế bộ giải mã cổng NAND hiếm khi được dùng vì mỗi vi mạch nhớ cần một bộ giải mã.

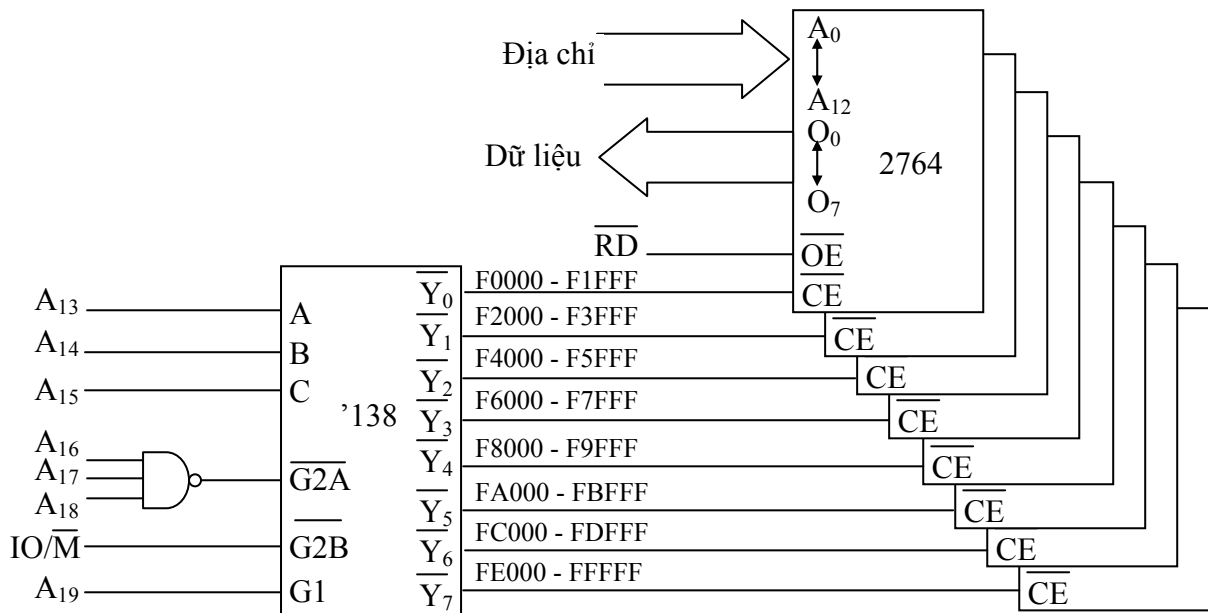
3. Bộ giải mã 3-8 (74LS138)

Đây là bộ giải mã ở dạng vi mạch (IC), được sử dụng khá phổ biến.

Từ bảng chân lý ta thấy tại bất kỳ thời điểm nào cũng chỉ có một đầu ra bằng 0. Để có đầu ra bằng 0 thì 3 đầu vào cho phép bộ giải mã hoạt động đều phải ở mức tác động, tức là $\overline{G2A} = 0$, $\overline{G2B} = 0$ và $G1 = 1$. Khi bộ giải mã được phép hoạt động thì các đầu vào C, B và A sẽ chọn đầu ra nào bằng 0. Các đầu ra của bộ giải mã sẽ được nối tới các đầu vào \overline{CE} hay \overline{CS} của các vi mạch nhớ.



Các đầu vào						Các đầu ra							
Cho phép			Chọn										
$\overline{G2A}$	$\overline{G2B}$	G1	C	B	A	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$	$\overline{Y_4}$	$\overline{Y_5}$	$\overline{Y_6}$	$\overline{Y_7}$
1	x	x	x	x	x	1	1	1	1	1	1	1	1
x	1	x	x	x	x	1	1	1	1	1	1	1	1
x	x	0	x	x	x	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1
0	0	1	0	0	1	1	0	1	1	1	1	1	1
0	0	1	0	1	0	1	1	0	1	1	1	1	1
0	0	1	0	1	1	1	1	1	0	1	1	1	1
0	0	1	1	0	0	1	1	1	1	0	1	1	1
0	0	1	1	0	1	1	1	1	1	1	0	1	1
0	0	1	1	1	0	1	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0



Giải mã địa chỉ cho bộ nhớ 64Kx8 (gồm 8 EPROM 2764) trong hệ vi xử lý 8088

Trong mạch trên các đầu ra của bộ giải mã 74LS138 được nối tới đầu vào \overline{CE} , còn tín hiệu \overline{RD} từ 8088 được nối tới đầu vào \overline{OE} của 8 vi mạch nhớ EPROM 2764. Tại bất kỳ thời điểm nào cũng chỉ có một đầu ra bằng 0 và do đó chỉ có một vi mạch nhớ được chọn và gửi dữ liệu của nó trên bus dữ liệu khi tín hiệu $\overline{RD} = 0$.

Nhìn vào mạch điện ta thấy, khi tất cả các bit địa chỉ từ $A_{19} - A_{16}$ đều bằng 1 và có tín hiệu chọn bộ nhớ từ 8088 ($IO/\overline{M} = 0$) sẽ làm cho $\overline{G2A} = 0$, $\overline{G2B} = 0$, $G1 = 1$, và do đó kích hoạt bộ giải mã. Khi bộ giải mã đã hoạt động thì các bit địa chỉ từ $A_{15} - A_{13}$ sẽ xác định đầu ra nào bằng 0 và tương ứng vi mạch nhớ nào sẽ được chọn.

Như vậy, dạng địa chỉ của một ô nhớ được giải mã là

1111 XXXX XXXX XXXX XXXX

Từ đây suy ra khoảng địa chỉ của cả bộ nhớ là từ

1111 0000 0000 0000 0000 = F0000H

đến

1111 1111 1111 1111 1111 = FFFFFH

Ta cũng có thể xác định được khoảng địa chỉ của từng vi mạch nhớ nối tới đầu ra của bộ giải mã. Ví dụ xác định khoảng địa chỉ của vi mạch nhớ nối tới đầu ra 0: vì vi mạch này được chọn khi CBA = 000 nên địa chỉ của ô nhớ trong vi mạch này có dạng

CBA

1111 000X XXXX XXXX XXXX

Từ đây suy ra khoảng địa chỉ của vi mạch nhớ là từ

1111 0000 0000 0000 = F0000H

đến

1111 0001 1111 1111 = F1FFFH

CBA

1111 001X XXXX XXXX XXXX

Từ đây suy ra khoảng địa chỉ của vi mạch nhớ là từ

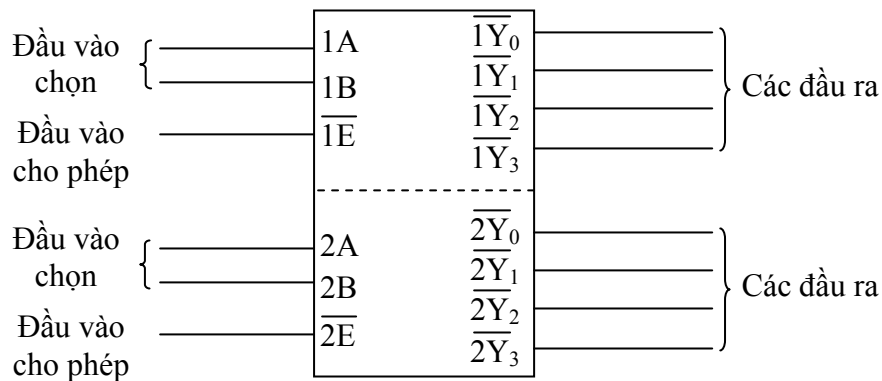
1111 0010 0000 0000 = F2000H

đến

1111 0011 1111 1111 = F3FFFH

4. Bộ giải mã kép 2-4 (74LS139)

74LS139 chứa hai bộ giải mã 2-4, mỗi bộ giải mã có đầu vào địa chỉ, cho phép và đầu ra riêng. Sơ đồ chân và bảng chân lý của bộ giải mã 74LS139 như sau:



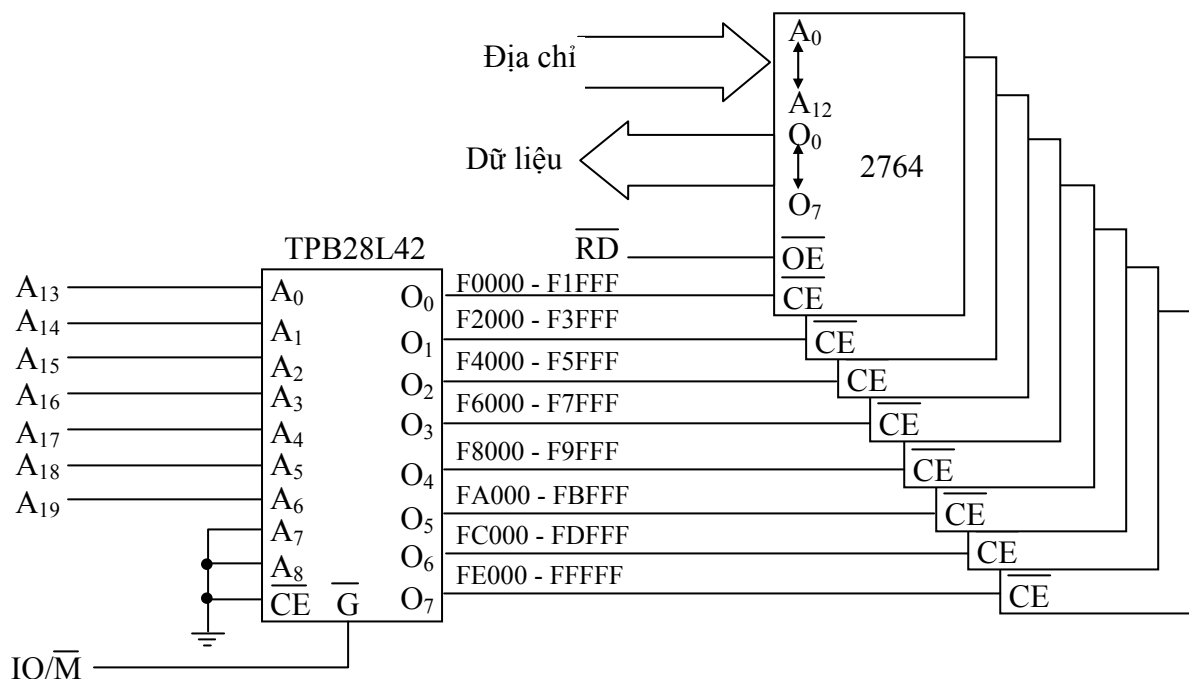
Đầu vào			Đầu ra			
\overline{E}	A	B	\overline{Y}_0	\overline{Y}_1	\overline{Y}_2	\overline{Y}_3
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0
1	x	x	1	1	1	1

5. Bộ giải mã PROM

Các bộ giải mã 74LS138 và 74LS139 có số đầu vào địa chỉ và đầu vào cho phép hạn chế nên vẫn phải dùng thêm các mạch logic phụ (cổng NAND). Đây cũng chính là lý do để người ta thay thế các bộ giải mã này bằng các bộ giải mã dùng PROM hoặc PLA (Programable Logic Array), với ưu điểm chính là chúng có nhiều đầu vào cho các bit địa chỉ và vì thế rất thích hợp cho các hệ vi xử lý sau này với không gian địa chỉ lớn.

Để minh họa cho việc dùng PROM làm bộ giải mã ta lấy lại ví dụ giải mã cho bộ nhớ 64 KB EPROM trong phần trước. Ở đây ta dùng PROM loại TPB28L42 512×8 làm bộ giải mã. PROM này có 9 đầu vào địa chỉ, 8 đầu ra dữ liệu. Để thực hiện giải mã PROM phải được lập trình chứa các mẫu bit sao cho với mỗi mẫu bit có một đầu ra dữ liệu bằng 0 để chọn một vi mạch nhớ EPROM. Dưới đây là mẫu bit lập trình cho PROM và sơ đồ nối ghép bộ giải mã PROM TPB28L42 với bộ nhớ 64 KB EPROM.

Các đầu vào										Các đầu ra							
\overline{G}	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	O ₀	O ₁	O ₂	O ₃	O ₄	O ₅	O ₆	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
Các tổ hợp còn lại										1	1	1	1	1	1	1	1



Giải mã địa chỉ cho bộ nhớ 64Kx8 (gồm 8 EPROM 2764) trong hệ vi xử lý 8088

Thuận lợi chính khi dùng PROM làm bộ giải mã là có thể thay đổi dải địa chỉ một cách dễ dàng. Ngoài ra, vì chỉ có 8 trong 512 ô nhớ cần được lập trình, các ô nhớ còn lại đều có logic 1 nên tiết kiệm thời gian cho nhà sản xuất.

6. Các bộ giải mã lập trình được PLD

Gần đây người ta sử dụng các thiết bị logic lập trình được (PLD, Programmable Logic Device) làm các bộ giải mã địa chỉ. Có 3 loại PLD: PLA (Programmable Logic Array), PAL (Programmable Array Logic) và GAL (Gate Array Logic). Các thiết bị này xuất hiện vào giữa thập kỷ 1970 nhưng gần đây mới được dùng trong các hệ thống nhớ và thiết kế số.

III - PHỐI GHÉP 8088 VỚI BỘ NHỚ

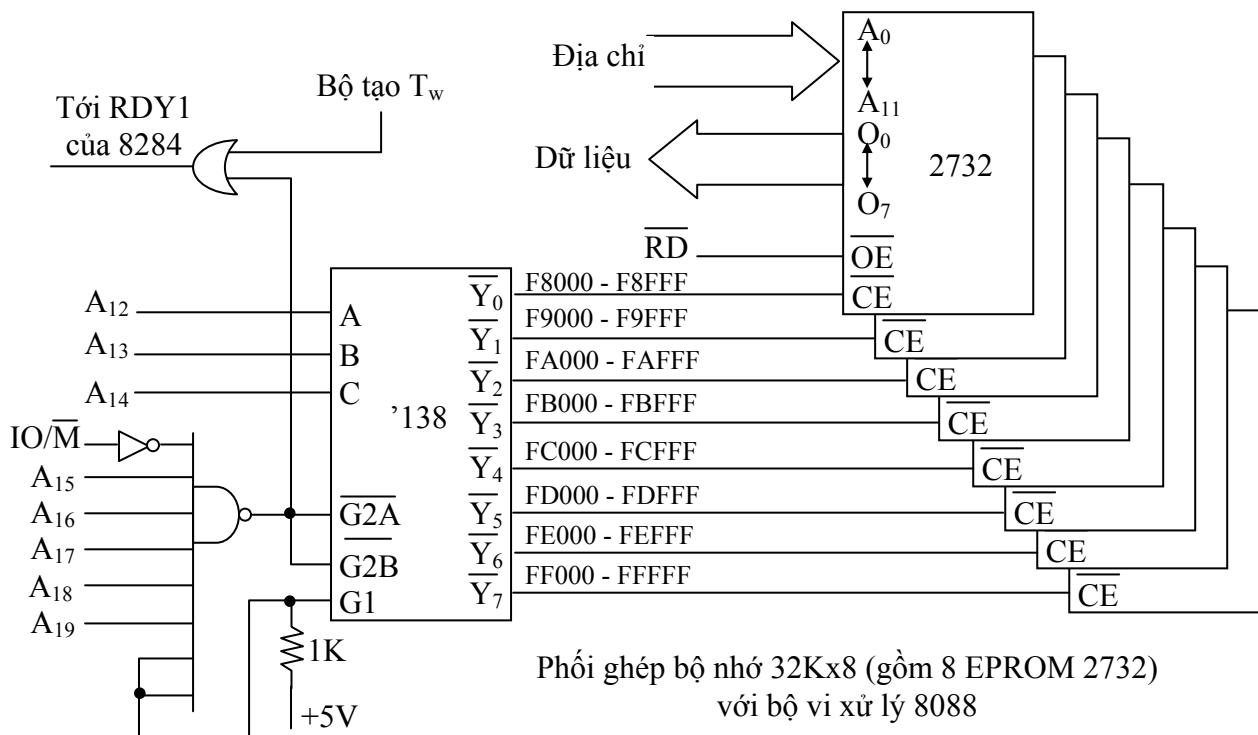
Bộ vi xử lý 8088 có bus dữ liệu 8 bit nên rất thích hợp khi nối với các vi mạch nhớ 8 bit. Nếu thời gian truy nhập của các bộ nhớ nhỏ hơn 420 ns thì việc phối ghép 8088 với bộ nhớ đơn giản chỉ là việc giải mã địa chỉ cho bộ nhớ. Tuy nhiên, nếu thời gian truy nhập của các bộ nhớ lớn hơn 420 ns thì ngoài việc giải mã địa chỉ cho bộ nhớ còn phải chèn thêm các chu kỳ đợi.

1. Phối ghép 8088 với EPROM

Loại EPROM được dùng phổ biến hiện nay là 2732 4K×8. EPROM 2732 có 12 đầu vào địa chỉ với dung lượng 4 KB.

Để minh họa việc phối ghép EPROM với 8088 ta dùng bộ giải mã 74LS138. Với 8 đầu ra, bộ giải mã 74LS138 có thể giải mã cho 8 vi mạch nhớ, nếu dùng EPROM loại 2732 4K×8 thì tổng dung lượng bộ nhớ được giải là $8 \times 4KB = 32 KB$.

Như ta đã biết, bộ vi xử lý 8088 làm việc với tần số đồng hồ 5 MHz có thời gian truy nhập bộ nhớ là 420 ns. Vi mạch nhớ EPROM 2732 có thời gian truy nhập bộ nhớ là 450 ns, cộng với thời gian trễ của bộ giải mã (12 ns), tổng cộng là 462 ns, chậm hơn bộ vi xử lý. Bởi vậy ta phải chèn thêm chu kỳ đợi. Việc này được thực hiện bằng cách dùng một cổng NAND vừa tạo ra tín hiệu cho phép bộ giải mã hoạt động vừa tạo ra tín hiệu cho bộ tạo chu kỳ đợi. Với một chu kỳ đợi được chèn vào, thời gian truy nhập bộ nhớ của 8088 sẽ tăng lên 620 ns, đủ để truy nhập bộ nhớ EPROM 2732 ngay cả khi có trễ qua bộ giải mã. Sơ đồ nối ghép EPROM 2732 với 8088 như sau:

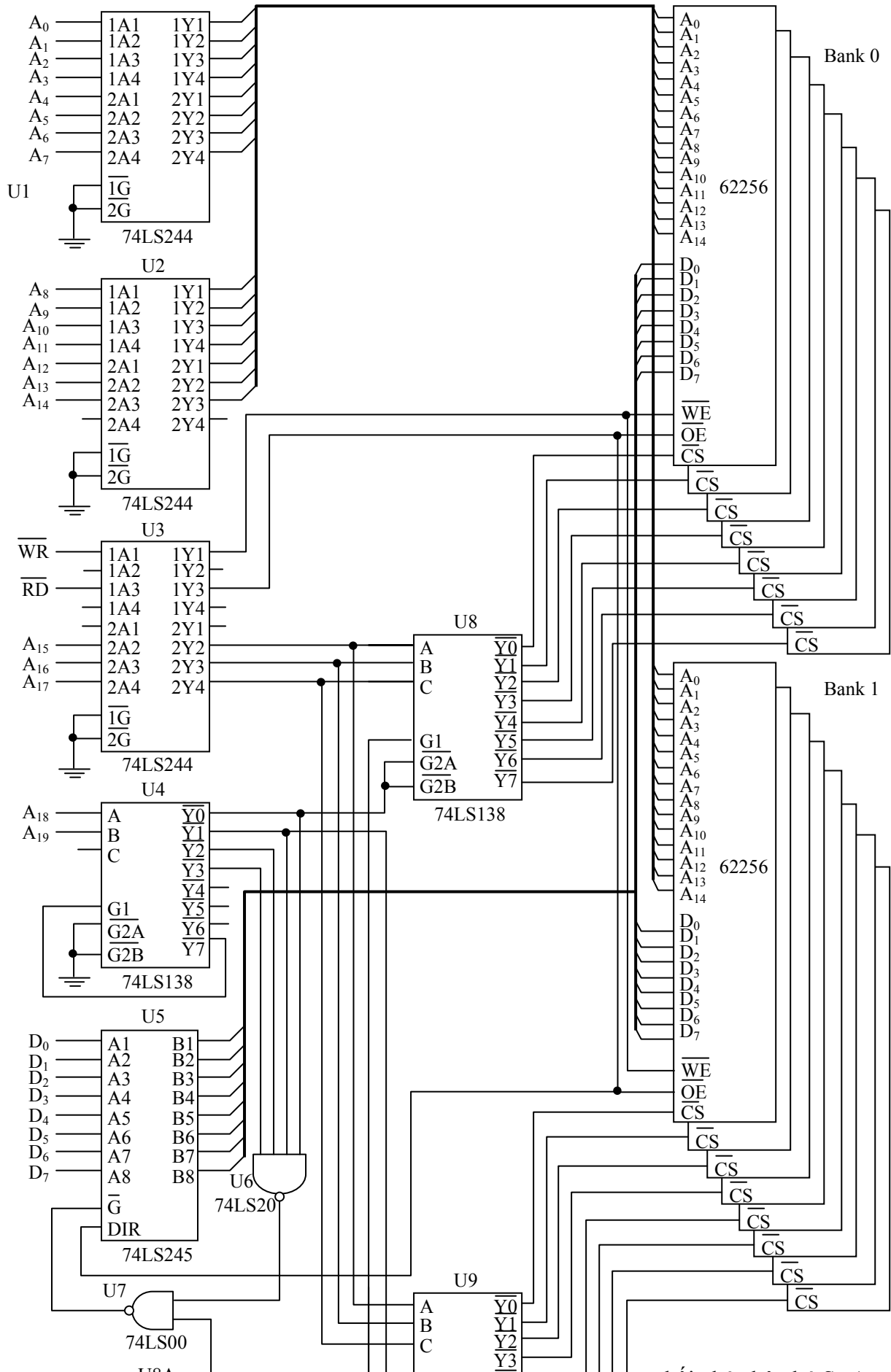


Khoảng địa chỉ mà bộ giải mã địa chỉ trong sơ đồ trên gán cho bộ nhớ EPROM là từ F8000H đến FFFFFH, chiếm 32 KB phần trên đỉnh trong toàn thể không gian nhớ 1 MB của 8088. Phần địa chỉ dành cho EPROM vì địa chỉ FFFF0H là địa chỉ mà 8088 thực hiện lệnh sau khi khởi động cứng.

2. Phối ghép RAM với 8088

Phối ghép RAM với 8088 dễ hơn EPROM vì RAM có thời gian truy nhập nhanh nên không cần chèn thêm chu kỳ đợi. Phần bộ nhớ lý tưởng dành cho RAM là đáy của không gian nhớ 1 MB, đây là nơi chứa các vector ngắt.

Để minh họa ta lấy 16 vi mạch nhớ SRAM 62256 32K×8 phối ghép với 8088.



Mạch phối ghép trên sử dụng ba bộ giải mã 74LS138, hai bộ U8 và U9 để chọn 16 vi mạch nhớ RAM 62256 khác nhau, một bộ U4 để chọn bộ giải mã U8 và U9. Địa chỉ bộ nhớ 20 bit bắt đầu với 00 chọn U8, bắt đầu với 01 chọn U9. Các chân còn lại của U4 dự trữ để mở rộng bộ nhớ. Sau này nếu muốn thêm các khối RAM 256K×8 thì chỉ cần thêm RAM và các bộ giải mã.

Dải địa chỉ bộ nhớ được xác định như sau: khi bit địa chỉ $A_{19}A_{18} = 00$ thì bộ giải mã U8 được chọn, do đó địa chỉ của các ô nhớ trong khối Bank 0 có dạng

00XX XXXX XXXX XXXX XXXX

Từ đây suy ra dải địa chỉ của khối nhớ Bank 0 là từ

0000 0000 0000 0000 0000 = 00000H

đến

0011 1111 1111 1111 1111 = 3FFFFH

Còn khi bit địa chỉ $A_{19}A_{18} = 01$ thì bộ giải mã U9 được chọn, do đó địa chỉ của các ô nhớ trong khối Bank 1 có dạng

01XX XXXX XXXX XXXX XXXX

Từ đây suy ra dải địa chỉ của khối nhớ Bank 1 là từ

0100 0000 0000 0000 0000 = 40000H

đến

0111 1111 1111 1111 1111 = 7FFFFH

Như vậy dải địa chỉ của cả bộ nhớ 512 KB là từ 00000H đến 7FFFFH.