

NỘI DUNG MÔN HỌC

Chương 1: Hệ đếm

Chương 2: Hàm Boole và cổng logic

Chương 3: Mạch logic tổ hợp

Chương 4: Mạch logic tuần tự

Chương 5: Bộ nhớ bán dẫn

Chương 3 – MẠCH LOGIC TỔ HỢP

3.1. Khái niệm chung

3.2. Phân tích mạch logic tổ hợp

3.3. Thiết kế mạch logic tổ hợp

3.4. Mạch mã hóa – giải mã

3.5. Mạch hợp kênh – phân kênh

3.6. Mạch số học

3.7. Mạch so sánh

3.8. Mạch tạo/ kiểm tra chẵn lẻ

3.9. Mạch tạo/ kiểm tra mã Hamming

Mạch tổ hợp (1)

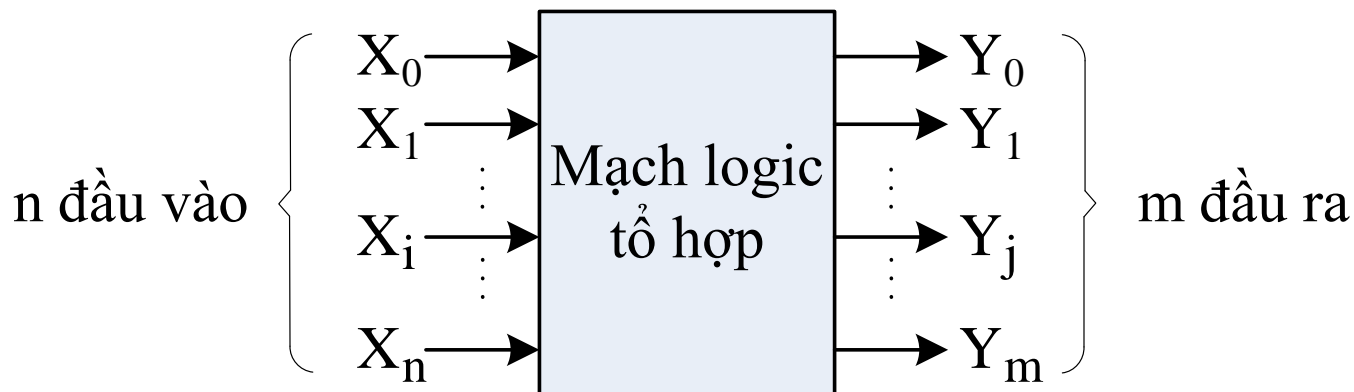
➤ *Đặc điểm:*

- Trị số của tín hiệu đầu ra ở thời điểm đang xét chỉ phụ thuộc vào tổ hợp các giá trị tín hiệu đầu vào.
- Được cấu trúc nên từ các cổng logic.

➤ *Phương pháp biểu diễn chức năng logic của mạch tổ hợp:*

- Hàm logic (thường áp dụng cho vi mạch cỡ nhỏ - SSI)
- Bảng trạng thái (vi mạch cỡ vừa – MSI)
- Bảng Karnaugh
- Đồ thị dạng xung ...

Mạch tổ hợp (2) – Sơ đồ khối tổng quát



➤ Hệ phương trình tổng quát:

$$\begin{cases} Y_0 = f_0(X_0, X_1, \dots, X_n) \\ Y_1 = f_1(X_0, X_1, \dots, X_n) \\ \vdots \\ Y_m = f_m(X_0, X_1, \dots, X_n) \end{cases}$$

Chương 3 – MẠCH LOGIC TỔ HỢP

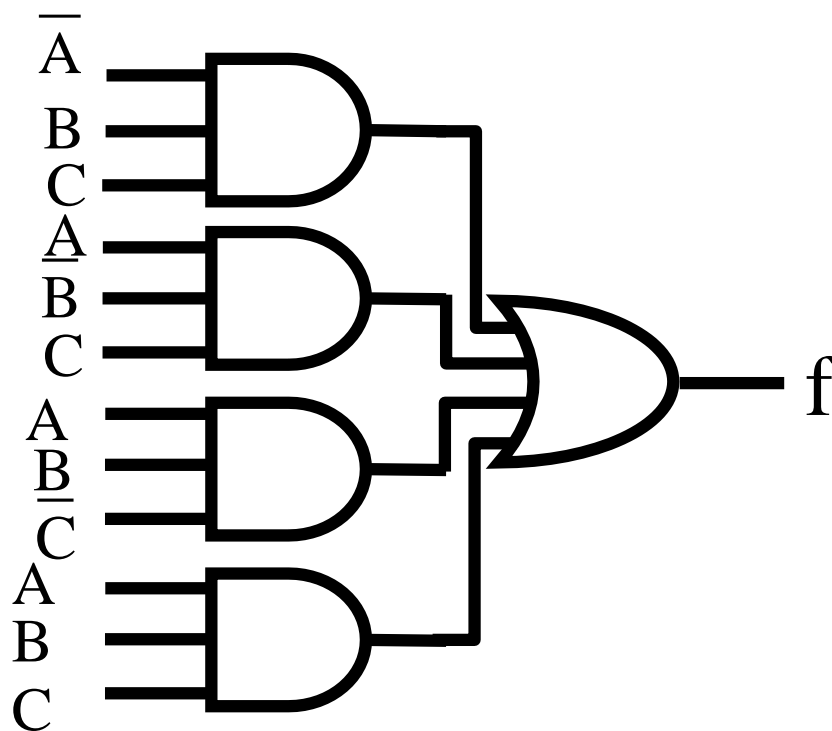
- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp**
- 3.3. Thiết kế mạch logic tổ hợp
- 3.4. Mạch mã hóa – giải mã
- 3.5. Mạch hợp kênh – phân kênh
- 3.6. Mạch số học
- 3.7. Mạch so sánh
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ
- 3.9. Mạch tạo/ kiểm tra mã Hamming

Phân tích mạch logic tổ hợp (1)

- Phân tích mạch logic tổ hợp là từ sơ đồ cho trước xác định chức năng, dạng sóng, tính năng kỹ thuật,... của mạch. Từ đó có thể rút gọn, chuyển đổi dạng thực hiện của mạch điện để có được lời giải tối ưu.
- Phương pháp:
 - Với mạch đơn giản: tiến hành lập bảng trạng thái, viết biểu thức, rút gọn, tối ưu (nếu cần) và vẽ lại mạch điện.
 - Với mạch phức tạp: tiến hành phân đoạn mạch để viết biểu thức, sau đó rút gọn, tối ưu (nếu cần) và vẽ lại mạch điện.

Phân tích mạch logic tổ hợp (2) – Ví dụ

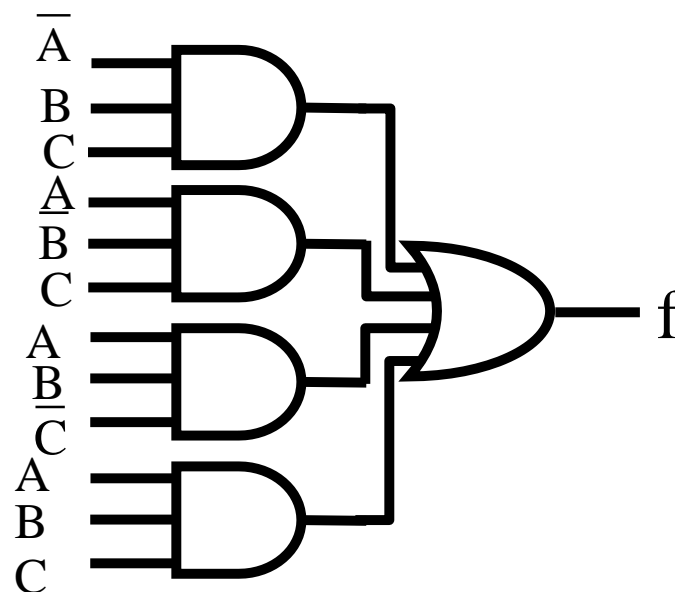
➤ Phân tích mạch điện sau, nêu chức năng của mạch:



Phân tích mạch logic tổ hợp (3) - VD

Bước 1: Lập Bảng trạng thái:

A	B	C	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Bước 2: Viết biểu thức:

$$f(A, B, C) = \sum(3, 5, 6, 7)$$

Phân tích mạch logic tổ hợp (4) - VD

Bước 3: Rút gọn: $f(A, B, C) = \sum(3, 5, 6, 7)$

BC	00	01	11	10
A				
0	0	0	1	0
1	0	1	1	1

$f_1 = AC$ (blue box around 01, 11 in row A=1)
 $f_2 = AB$ (red oval around 11, 10 in row A=1)
 $f_3 = BC$ (dashed red box around 11 in row A=0 and 11 in row A=1)

$$f = AB + BC + AC$$

Bước 4: Tối ưu về dạng toàn NAND:

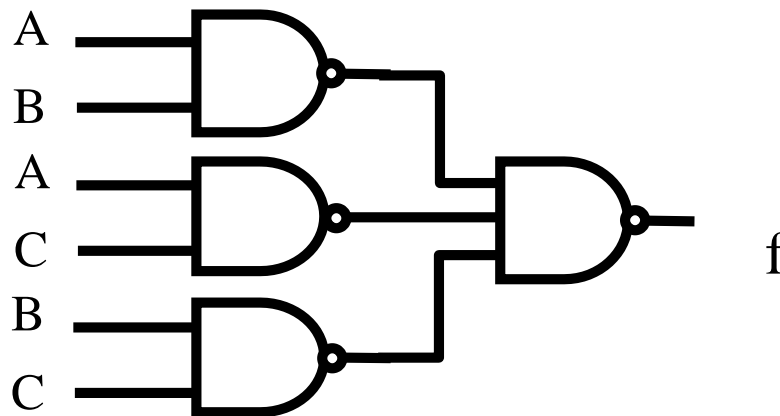
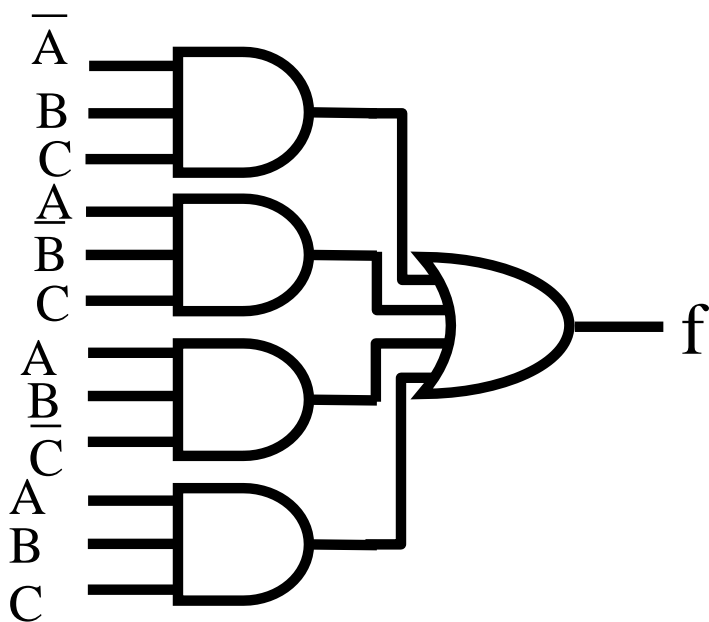
$$f = AB + BC + AC$$

$$\overline{\overline{f}} = \overline{\overline{AB + BC + AC}}$$

$$f = \overline{\overline{AB} \cdot \overline{BC} \cdot \overline{AC}}$$

Phân tích mạch logic tổ hợp (5) - VD

Bước 5: Vẽ mạch điện dạng tối ưu:



Ví dụ 2-1

3-7

Cho hàm logic: $F(A, B, C, D) = \sum(0, 2, 5, 6, 7, 8, 10, 13, 15)$

a) Viết biểu thức tối giản của hàm

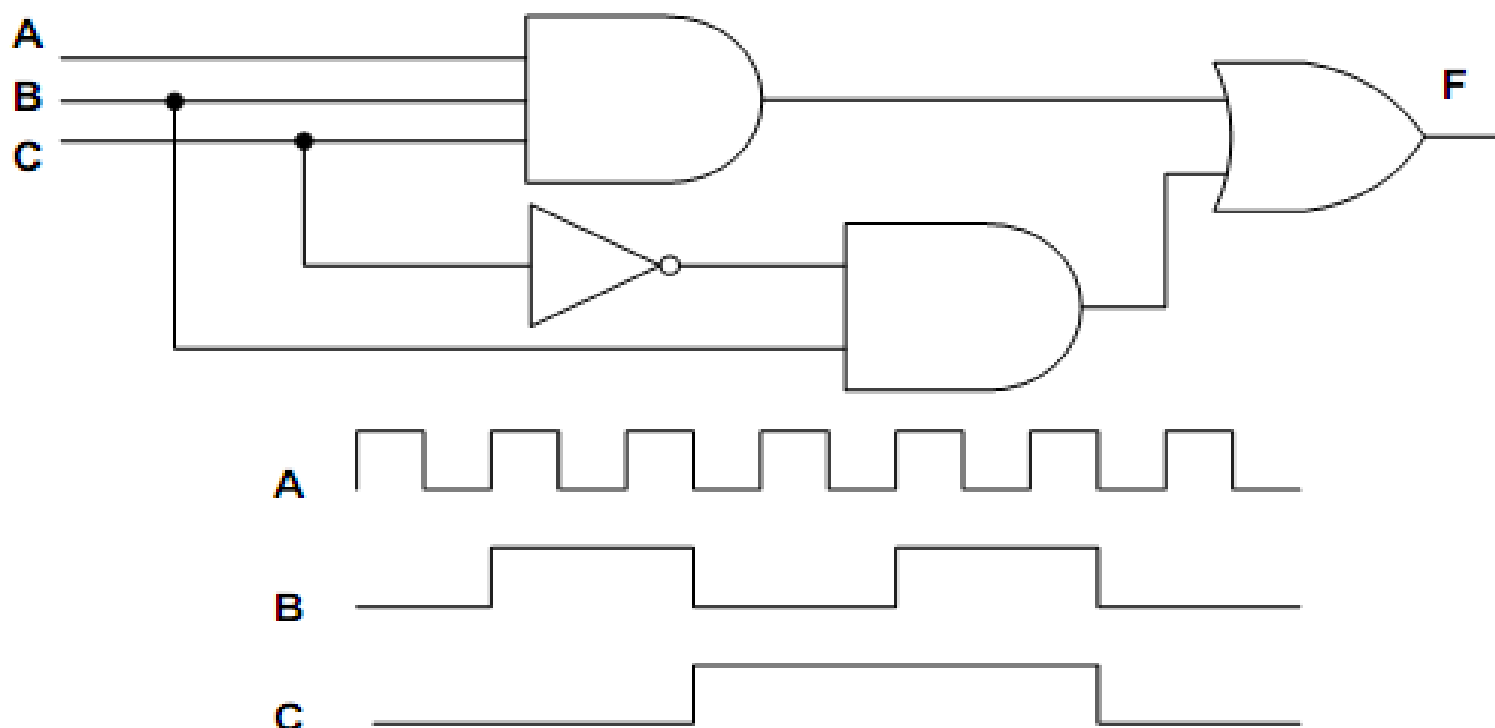
b) Thực hiện hàm bằng 1 mạch tối ưu toàn NOR 2 lối vào

$$\begin{aligned}\bar{F} &= \bar{B}D + B\bar{C}\bar{D} + AB\bar{D} = \bar{B}D + \bar{D}(B\bar{C} + AB) = \bar{B}D + \bar{D}(\overline{B\bar{C} + AB}) = \\ &= \overline{\overline{B} + \overline{D}} + \overline{\overline{D} + \overline{B\bar{C} + AB}} = \overline{\overline{B} + \overline{D}} + \overline{\overline{D} + \overline{B} + \overline{C} + \overline{A} + \overline{B}} = \\ &\Rightarrow F = \overline{\overline{\overline{B} + \overline{D}} + \overline{\overline{D} + \overline{B} + \overline{C} + \overline{A} + \overline{B}}}\end{aligned}$$

$$F = \overline{\overline{\overline{B} + \overline{D}} + \overline{\overline{D} + \overline{B} + \overline{C} + \overline{A} + \overline{B}}}$$

Ví dụ 2-2

3-11 Cho mạch logic và giản đồ xung sau hình 3-31. Hãy vẽ hàm ra F.



Hình 3-31. Sơ đồ logic và giản đồ xung bài 3-11.

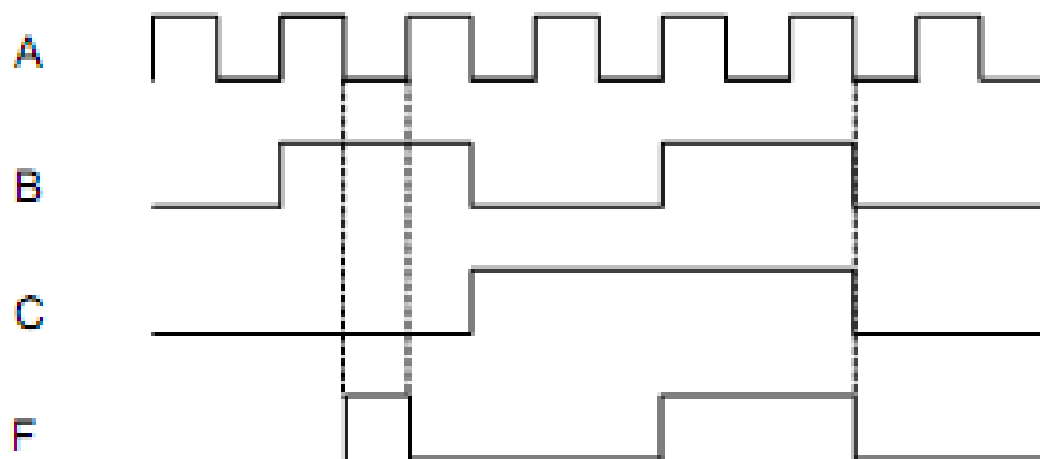
Ví dụ 2-2

$$F = ABC + B\bar{C}$$

Từ đó, lập được bảng trạng thái sau:

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Bảng trạng thái bài 3-11.



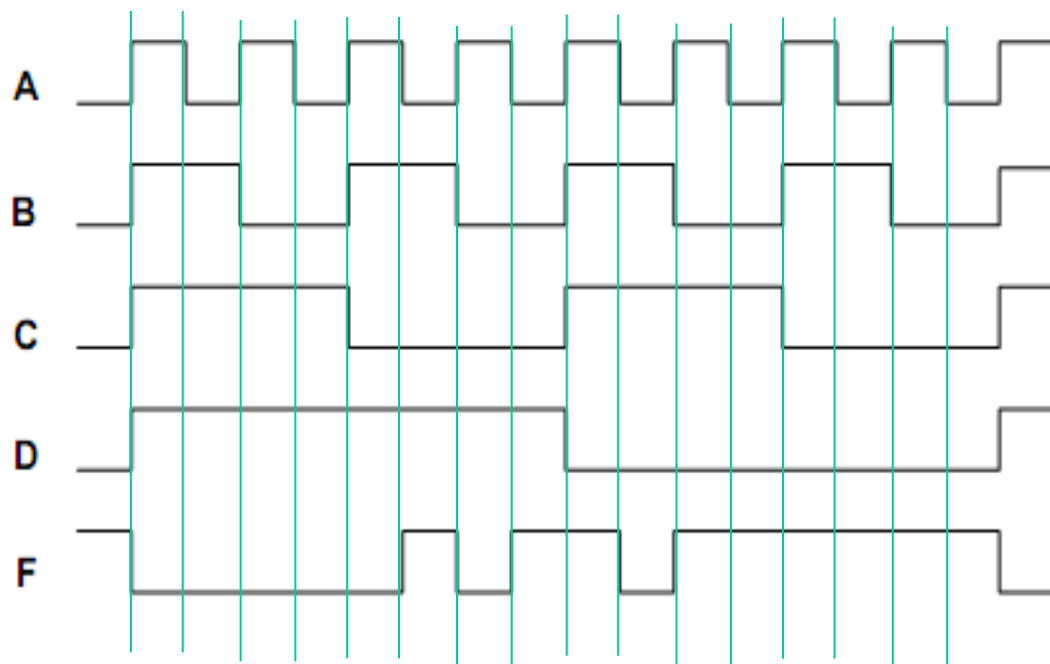
Hình 3-32. Giản đồ xung

Ví dụ 2-3

3-14

Cho giản đồ xung hình 3-34:

- Viết biểu thức hàm F theo dạng chuẩn minterm và Maxterm.
- Rút gọn hàm F và vẽ mạch thực hiện chỉ dùng cổng NAND 2 lối vào



Hình 3-34. Giản đồ xung bài 3-14

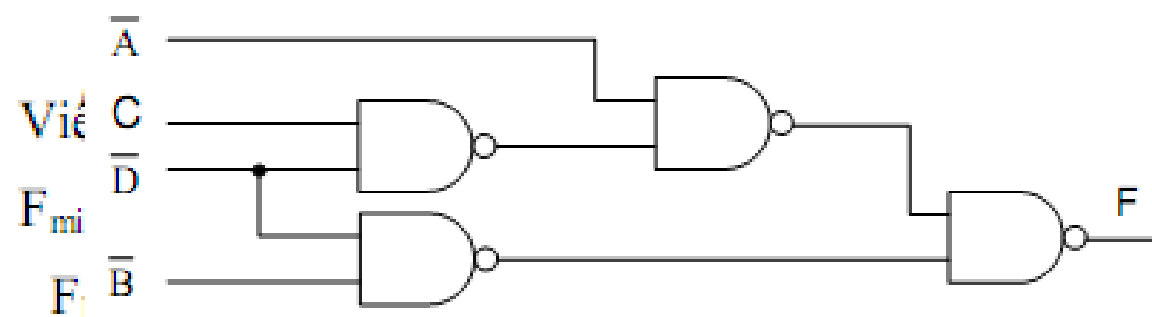
Bảng trạng thái

D	C	B	A	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

a) Từ bảng trạng thái, viết biểu thức theo dạng chuẩn minterm và Maxterm

$$F_{\text{minterm}} = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}D + \bar{A}BC\bar{D} + \bar{A}BCD + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + A\bar{B}C\bar{D} + A\bar{B}CD + AB\bar{C}\bar{D} + AB\bar{C}D + ABC\bar{D} + ABCD$$

$$F_{\text{Maxterm}} = (A + \bar{B} + \bar{C} + D)(\bar{A} + B + C + \bar{D})(\bar{A} + B + \bar{C} + \bar{D})(\bar{A} + \bar{B} + C + D)(\bar{A} + \bar{B} + C + \bar{D}) \times$$



Bảng trạng thái

D	C	B	A	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0

CD \ AB	00	01	11	10
00	1	1	1	1
01	1	1	1	0
11	0	0	0	0
10	1	0	0	1

b) Rút gọn hàm F và đưa về dạng toàn NAND

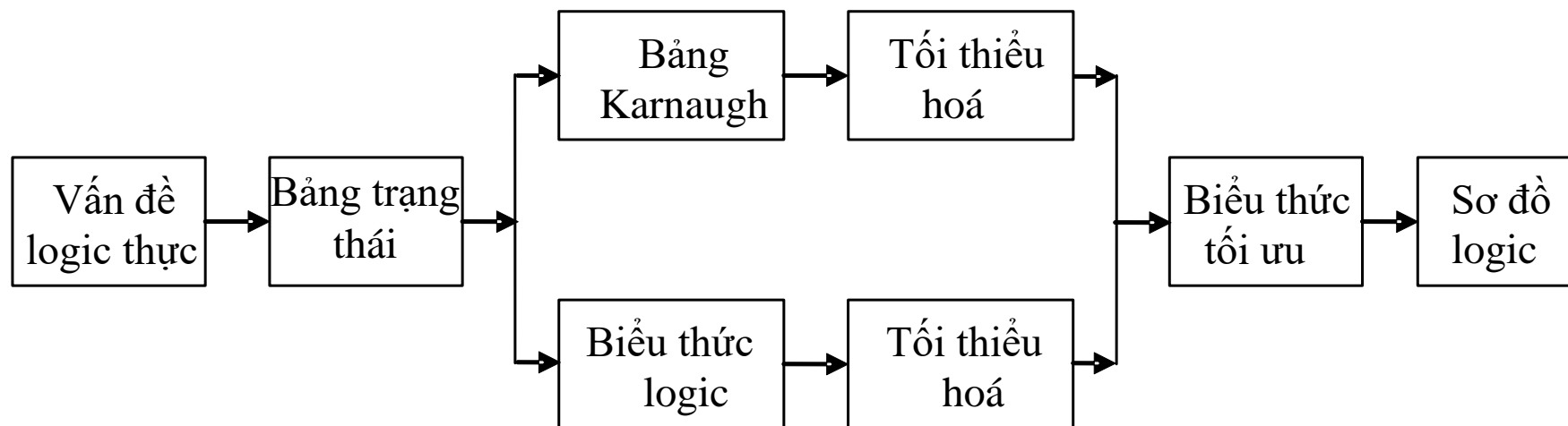
$$F = \bar{A}\bar{C} + \bar{A}D + \bar{B}\bar{D} = \bar{A}(\bar{C} + D) + \bar{B}\bar{D} = \bar{A}(\overline{\bar{C} + D}) + \bar{B}\bar{D} = \bar{A}\bar{C}\bar{D} + \bar{B}\bar{D} = \overline{\overline{\bar{A}\bar{C}\bar{D}} + \overline{\bar{B}\bar{D}}} = \overline{\overline{\bar{A}\bar{C}\bar{D}} \cdot \overline{\bar{B}\bar{D}}} = \overline{\overline{\bar{A}\bar{C}\bar{D}} \cdot \overline{\bar{B}\bar{D}}}$$

Chương 3 – MẠCH LOGIC TỔ HỢP

- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp
- 3.3. Thiết kế mạch logic tổ hợp**
- 3.4. Mạch mã hóa – giải mã
- 3.5. Mạch hợp kênh – phân kênh
- 3.6. Mạch số học
- 3.7. Mạch so sánh
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ
- 3.9. Mạch tạo/ kiểm tra mã Hamming

Thiết kế mạch logic tổ hợp

- Thiết kế mạch logic tổ hợp là bài toán từ yêu cầu (chức năng, dạng sóng, tính năng kỹ thuật, ...) xây dựng sơ đồ mạch thực hiện (ngược với bài toán phân tích).
- Phương pháp:



Chương 3 – MẠCH LOGIC TỔ HỢP

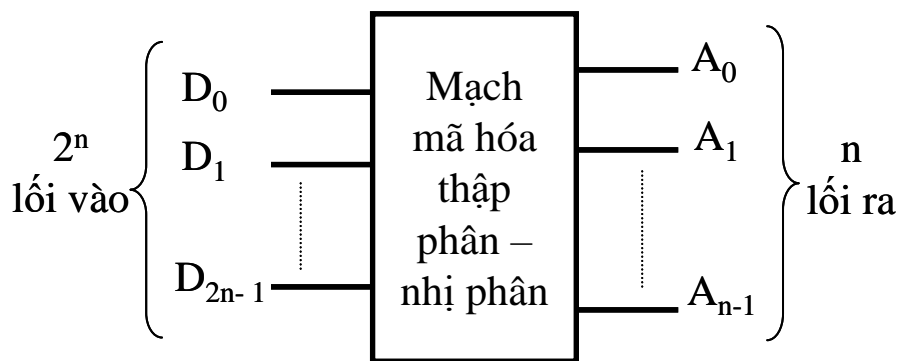
- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp
- 3.3. Thiết kế mạch logic tổ hợp
- 3.4. Mạch mã hóa – giải mã**
- 3.5. Mạch hợp kênh – phân kênh
- 3.6. Mạch số học
- 3.7. Mạch so sánh
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ
- 3.9. Mạch tạo/ kiểm tra mã Hamming

3.4.1. Mạch mã hóa – Khái niệm

- Mã hóa: là dùng văn tự, ký hiệu (hay mã) để biểu thị một đối tượng nào đó (tin tức).
- Mục đích: để tin tức truyền đi tốt hơn, cải thiện một số đặc tính như: độ tin cậy, tốc độ truyền tin, dung lượng kênh, tính bảo mật, ...
- Bộ mã hóa: thực hiện nhiệm vụ mã hóa. Một số bộ mã hóa:
 - Bộ mã hóa nhị phân
 - Bộ mã hóa BCD
 - Bộ mã hóa ưu tiên
 - ...

3.4.1. Mạch mã hóa: Mã hóa thập phân – nhị phân (1)

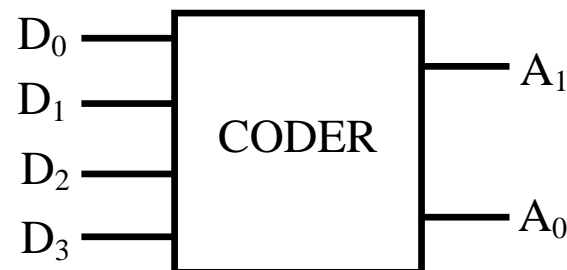
- Chức năng: mã hóa từ dạng thập phân đầu vào thành dạng nhị phân đầu ra.
- Bộ mã hóa N tín hiệu thập phân đầu vào có số bit đầu ra n thỏa mãn: $N \leq 2^n$
- Sơ đồ khối tổng quát:



3.4.1. Mạch mã hóa: Mã hóa thập phân – nhị phân (2)

➤ Bài toán: Thiết kế mạch mã hóa thập phân – nhị phân 2 bit.

➤ Sơ đồ khối:



➤ Bảng trạng thái:

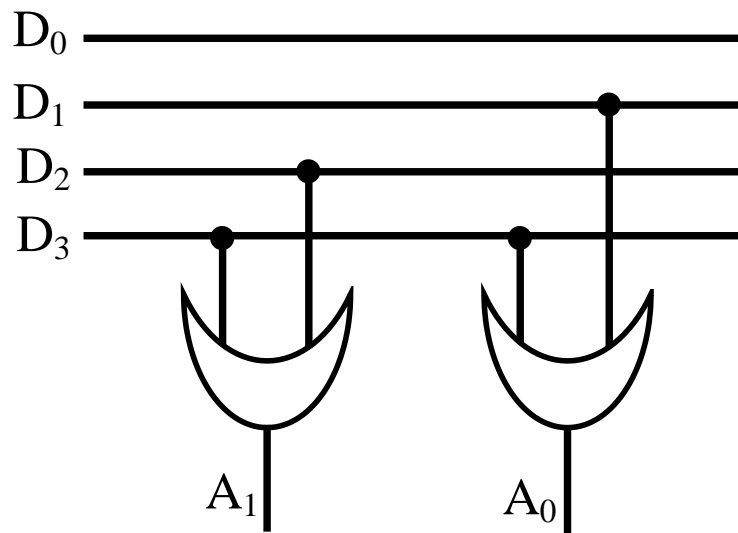
D0	D1	D2	D3	A1	A0
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

3.4.1. Mạch mã hóa: Mã hóa thập phân – nhị phân (3)

➤ Biểu thức hàm ra:

$$\begin{aligned} A_0 &= \overline{D_0} \cdot D_1 \cdot \overline{D_2} \cdot \overline{D_3} + \overline{D_0} \cdot \overline{D_1} \cdot \overline{D_2} \cdot D_3 \\ A_1 &= \overline{D_0} \cdot \overline{D_1} \cdot D_2 \cdot \overline{D_3} + \overline{D_0} \cdot \overline{D_1} \cdot \overline{D_2} \cdot D_3 \end{aligned} \Rightarrow \begin{aligned} A_0 &= D_1 + D_3 \\ A_1 &= D_2 + D_3 \end{aligned}$$

➤ Mạch điện:



3.4.1. Mạch mã hóa: Mã hóa thập phân – NBCD (1)

➤ *Chức năng:* Mã hóa 10 ký hiệu thập phân bằng 4 bit nhị phân.

➤ *Sơ đồ khối:*



➤ *Bảng trạng thái:*

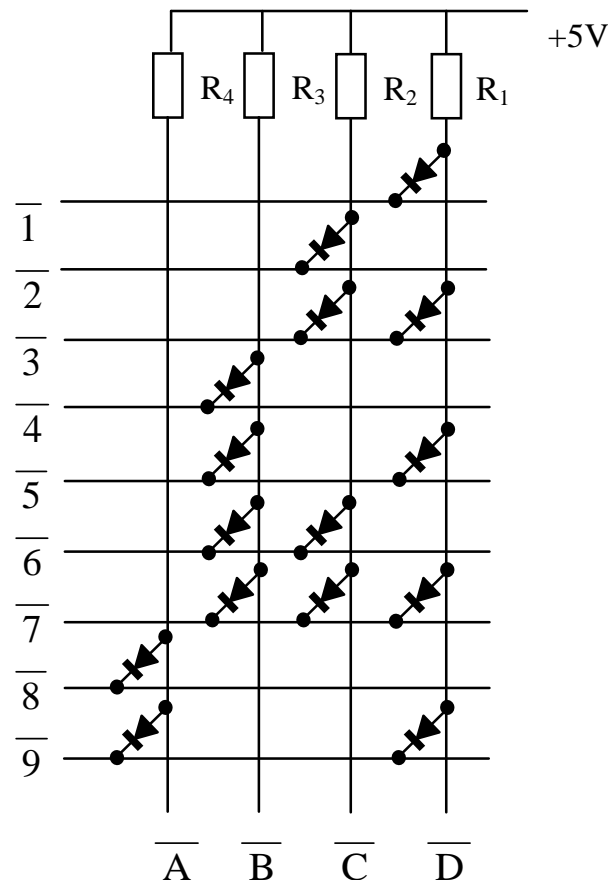
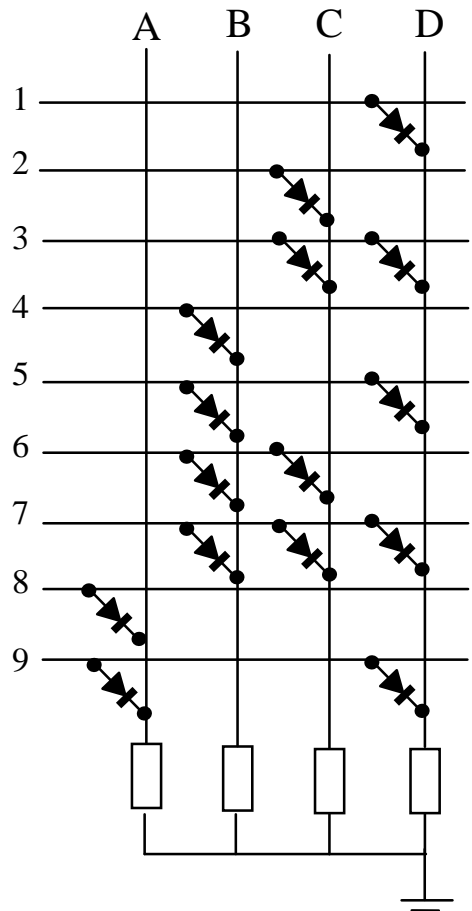
➤ *Biểu thức hàm ra:*

$$\begin{aligned}
 A &= 8 + 9 && = \Sigma (8, 9) \\
 B &= 4 + 5 + 6 + 7 && = \Sigma (4, 5, 6, 7) \\
 C &= 2 + 3 + 6 + 7 && = \Sigma (2, 3, 6, 7) \\
 D &= 1 + 3 + 5 + 7 + 9 && = \Sigma (1, 3, 5, 7, 9)
 \end{aligned}$$

Vào thập phân	Ra BCD 8 4 2 1 A B C D
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1

3.4.1. Mạch mã hóa: Mã hóa thập phân – NBCD (2)

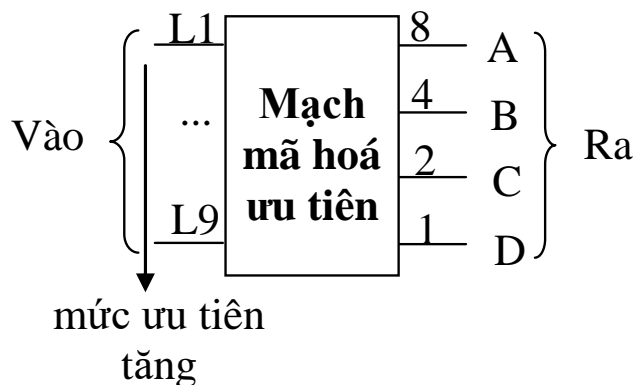
➤ Mạch điện (dạng ma trận OR): ➤ Mạch điện (dạng ma trận AND):



3.4.1. Mạch mã hóa: Mã hóa ưu tiên TP – NBCD (1)

➤ Khi có nhiều đầu vào tác động đồng thời, chỉ mã hoá tín hiệu vào có mức ưu tiên cao nhất. Mức ưu tiên do người thiết kế mạch xác định.

➤ Sơ đồ khối:



➤ Bảng trạng thái:

Vào thập phân									Ra BCD			
1	2	3	4	5	6	7	8	9	A	B	C	D
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1
x	1	0	0	0	0	0	0	0	0	0	1	0
x	x	1	0						0	0	1	1
			1						0	1	0	0
				1					0	1	0	1
					1				0	1	1	0
						1			0	1	1	1
							1		1	0	0	0
								1	1	0	0	1

3.4.1. Mạch mã hóa: Mã hóa ưu tiên TP – NBCD (2)

➤ Biểu thức:

$$D = "1" \text{ nếu } \begin{cases} L9 = "1" \\ L7 = "1" \text{ và } L8 \text{ bằng } "0" \\ L5 = "1" \text{ và } L6,8 \text{ bằng } "0" \\ L3 = "1" \text{ và } L4,6,8 \text{ bằng } "0" \\ L1 = "1" \text{ và } L2,4,6,8 \text{ bằng } "0" \end{cases}$$

$$\Rightarrow D = 1.\bar{2}.\bar{4}.\bar{6}.\bar{8} + 3.\bar{4}.\bar{6}.\bar{8} + 5.\bar{6}.\bar{8} + 7.\bar{8} + 9$$

$$C = 2.\bar{4}.\bar{5}.\bar{8}.\bar{9} + 3.\bar{4}.\bar{5}.\bar{8}.\bar{9} + 6.\bar{8}.\bar{9} + 7.\bar{8}.\bar{9}$$

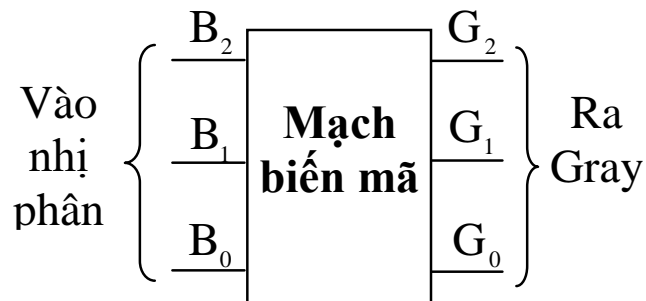
$$B = 4.\bar{8}.\bar{9} + 5.\bar{8}.\bar{9} + 6.\bar{8}.\bar{9} + 7.\bar{8}.\bar{9}$$

$$A = 8 + 9$$

3.4.2. Mạch biến mã (1)

➤ *Bài toán:* Thiết kế mạch biến mã từ nhị phân 3 bit sang Gray.

➤ *Sơ đồ khối:*



➤ *Bảng trạng thái:*

➤ *Biểu thức:*

$$G_0 = \sum(1, 2, 5, 6)$$

$$G_1 = \sum(2, 3, 4, 5)$$

$$G_2 = \sum(4, 5, 6, 7)$$

B2	B1	B0	G2	G1	G0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

3.4.2. Mạch biến mã (2)

➤ Rút gọn: $G_0 = \sum(1,2,5,6)$; $G_1 = \sum(2,3,4,5)$; $G_2 = \sum(4,5,6,7)$

G0

B1B0	00	01	11	10
B2				
0	0	1	0	1
1	0	1	0	1

$$G_0 = \overline{B_1} \cdot B_0 + B_1 \cdot \overline{B_0}$$

G2

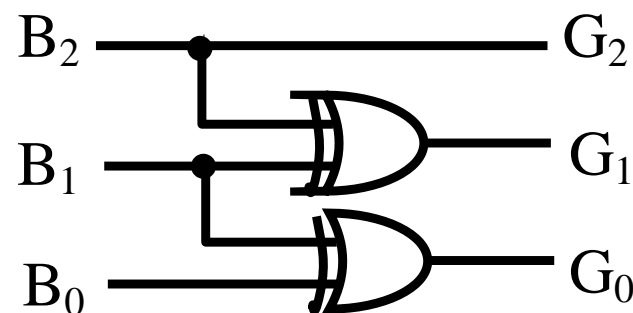
B1B0	00	01	11	10
B2				
0	0	0	0	0
1	1	1	1	1

$$G_2 = B_2$$

G1

B1B0	00	01	11	10
B2				
0	0	0	1	1
1	1	1	0	0

$$G_1 = B_2 \cdot \overline{B_1} + \overline{B_2} \cdot B_1$$



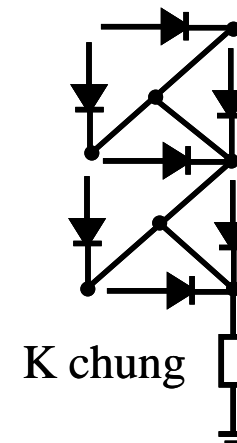
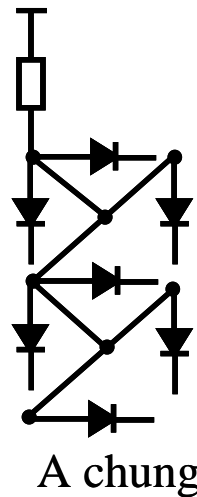
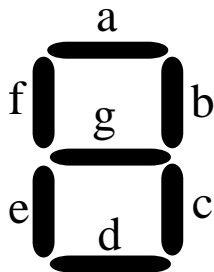
3.4.3. Mạch giải mã – Khái niệm

- Giải mã: là quá trình ngược lại của mã hóa, chuyển từ dạng mã sang tin tức.
- Bộ giải mã: thực hiện nhiệm vụ giải mã.
- Một số bộ giải mã:
 - Mạch giải mã BCD - 7 đoạn
 - Mạch giải mã nhị phân
 - ...

3.4.3. Mạch giải mã: BCD \rightarrow LED 7 đoạn (1)

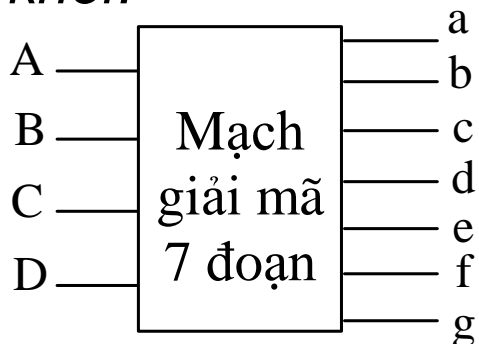
➤ *Dụng cụ 7 đoạn:*

- Dùng để hiển thị chữ số của một hệ đếm bất kỳ.
- Cấu tạo: gồm 7 đoạn làm bằng vật liệu có khả năng phát sáng (LED, LCD,...).
- Có hai loại LED 7 đoạn: Anode chung và Kathode chung.



3.4.3. Mạch giải mã: BCD \rightarrow LED 7 đoạn (2)

➤ Sơ đồ khối:



➤ Bảng trạng thái: (A chung)

➤ Biểu thức:

$$a = \sum (1,4)$$

$$b = \sum (5,6)$$

$$c = \sum (2)$$

$$d = \sum (1,4,7)$$

$$e = \sum (1,3,4,5,7,9)$$

$$f = \sum (1,2,3,7)$$

$$g = \sum (0,1,7)$$

	A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	1	1	0	0	1	1	1	1
2	0	0	1	0	0	0	1	0	0	1	0
3	0	0	1	1	0	0	0	0	1	1	0
4	0	1	0	0	1	0	0	1	1	0	0
5	0	1	0	1	0	1	0	0	1	0	0
6	0	1	1	0	0	1	0	0	0	0	0
7	0	1	1	1	0	0	0	1	1	1	1
8	1	0	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	1	0	0

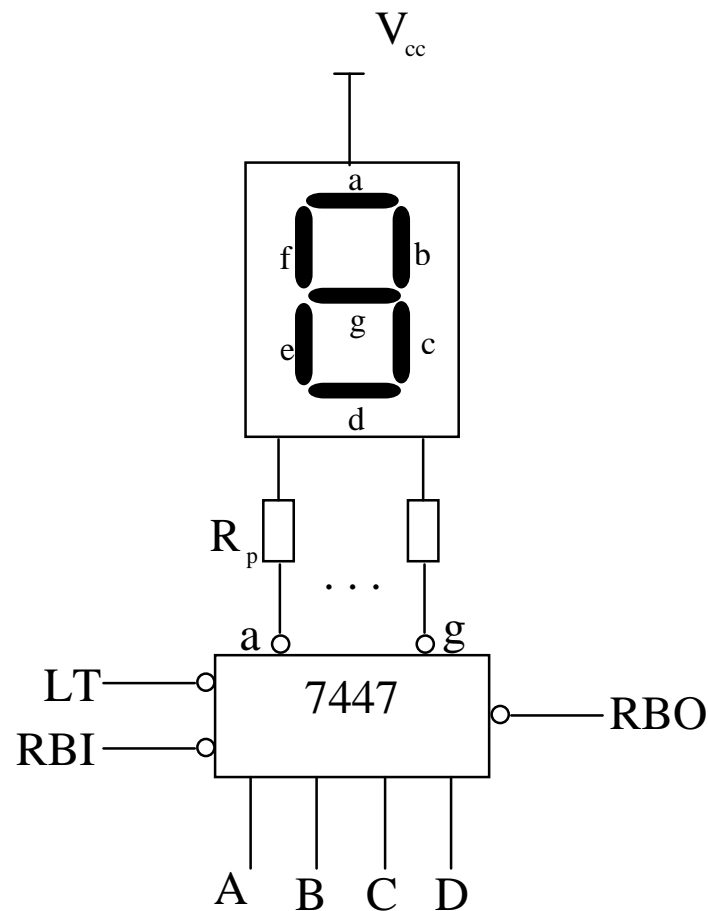
3.4.3. Mạch giải mã: BCD \rightarrow LED 7 đoạn (3)

➤ IC giải mã 7 đoạn:

TTL: A chung: 7447, 74247 (đầu ra ở mức tích cực thấp)

K chung: 7448 (đầu ra ở mức tích cực cao)

CMOS: 4511



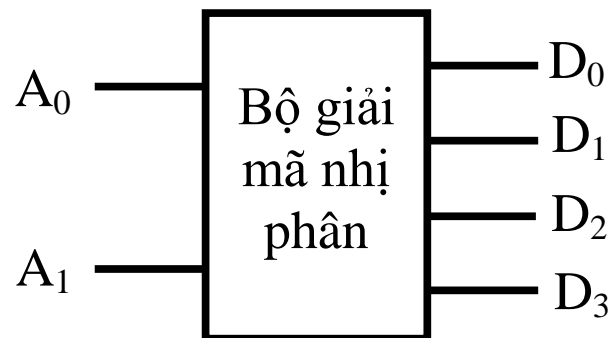
3.4.3. Mạch giải mã: Mạch giải mã nhị phân (1)

- Chức năng: giải mã từ dạng nhị phân đầu vào thành dạng thập phân đầu ra.
- Lựa chọn duy nhất một đầu ra ứng với một tổ hợp nhị phân đầu vào.
 - Mức tích cực cao: đầu ra được chọn bằng '1', các đầu ra còn lại bằng '0'.
 - Mức tích cực thấp: đầu ra được chọn bằng '0', các đầu ra còn lại bằng '1'.
- Bộ giải mã nhị phân n đầu vào có số đầu ra: $N = 2^n$

3.4.3. Mạch giải mã: Mạch giải mã nhị phân (2)

➤ Bài toán 1: Thiết kế mạch giải mã nhị phân 2 vào – 4 ra.

➤ Sơ đồ khối:



➤ Bảng trạng thái:

➤ Biểu thức:

$$D_0 = \overline{A_1} \cdot \overline{A_0}$$

$$D_1 = \overline{A_1} \cdot A_0$$

$$D_2 = A_1 \cdot \overline{A_0}$$

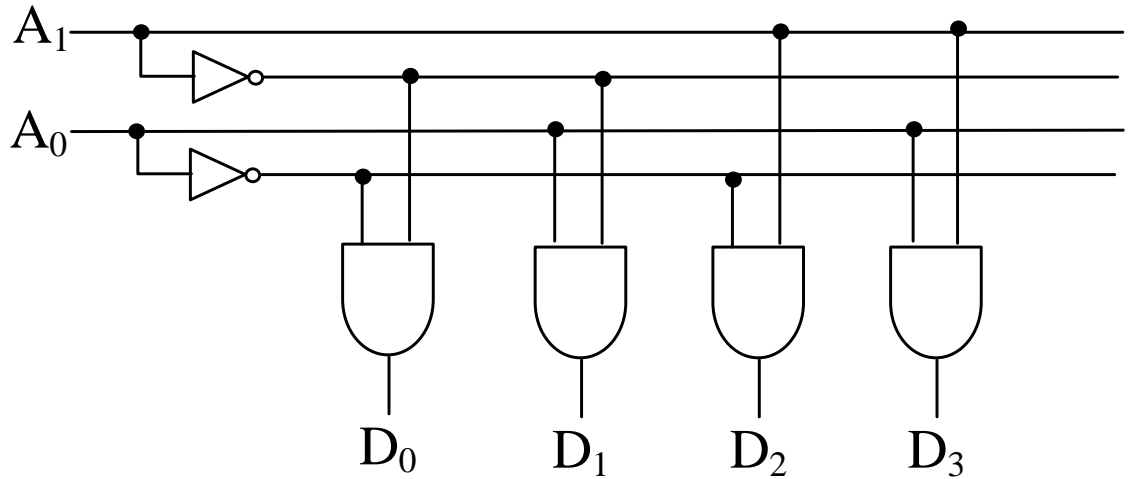
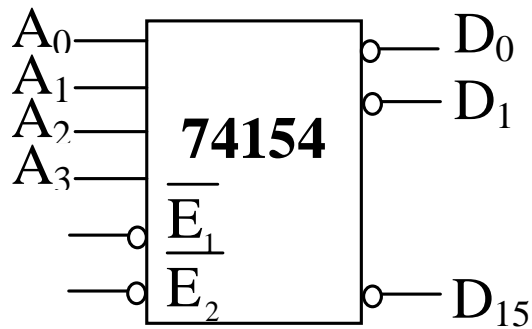
$$D_3 = A_1 \cdot A_0$$

A1	A0	D0	D1	D2	D3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

3.4.3. Mạch giải mã: Mạch giải mã nhị phân (3)

➤ Mạch điện:

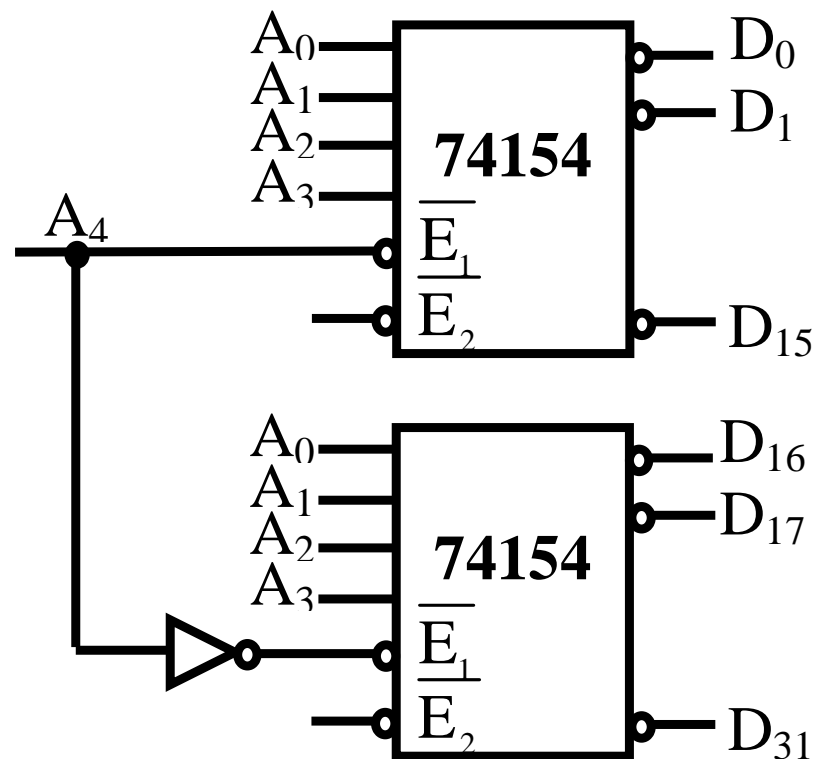
➤ IC thực tế: 74154



3.4.3. Mạch giải mã: Mạch giải mã nhị phân (4)

- Mở rộng dung lượng mạch giải mã nhị phân: Sử dụng IC 74154 thực hiện bộ giải mã nhị phân 5 lối vào địa chỉ.

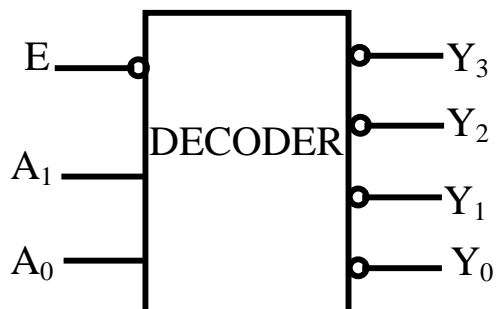
Sơ đồ khối:



3.4.3. Mạch giải mã: Mạch giải mã nhị phân (5)

➤ Bài toán 2: Cho mạch điện có BTT sau, nêu chức năng của mạch:

- Sơ đồ khối:



- Biểu thức hàm ra:

- Chức năng:

E	A1	A0	Y0	Y1	Y2	Y3
1	x	x	1	1	1	1
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

$$Y_0 = \overline{\overline{E}} \cdot \overline{\overline{A_1}} \cdot \overline{\overline{A_0}}$$

$$Y_1 = \overline{\overline{E}} \cdot \overline{\overline{A_1}} \cdot A_0$$

$$Y_2 = \overline{\overline{E}} \cdot A_1 \cdot \overline{\overline{A_0}}$$

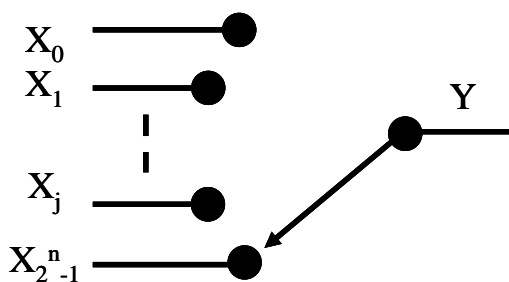
$$Y_3 = \overline{\overline{E}} \cdot A_1 \cdot A_0$$

Chương 3 – MẠCH LOGIC TỔ HỢP

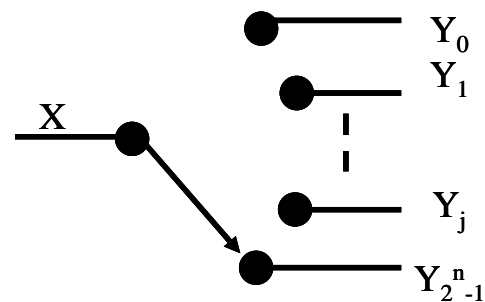
- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp
- 3.3. Thiết kế mạch logic tổ hợp
- 3.4. Mạch mã hóa – giải mã
- 3.5. Mạch hợp kênh – phân kênh**
- 3.6. Mạch số học
- 3.7. Mạch so sánh
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ
- 3.9. Mạch tạo/ kiểm tra mã Hamming

Mạch hợp kênh – phân kênh – Khái niệm

- Hợp kênh (ghép kênh): ghép nhiều kênh dữ liệu ở đầu vào thành một kênh duy nhất ở đầu ra.
- Mỗi kênh dữ liệu đầu vào có một địa chỉ xác định.
- Việc lựa chọn ghép kênh dữ liệu nào tới đầu ra phụ thuộc vào địa chỉ được chọn.
- Phân kênh (tách kênh): tách từ một kênh dữ liệu đầu vào thành nhiều kênh ở đầu ra (ngược của quá trình hợp kênh).



(a). Bộ hợp kênh (MUX)



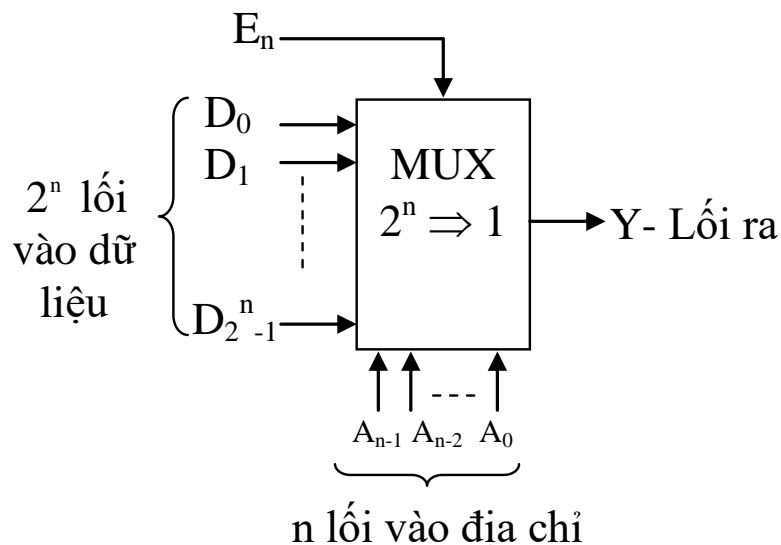
(b). Bộ phân kênh (DEMUX)

3.5.1. Mạch hợp kênh – MUX (1)

➤ *Chức năng:* lựa chọn nối một đầu vào dữ liệu tới đầu ra.

➤ *Sơ đồ khối:*

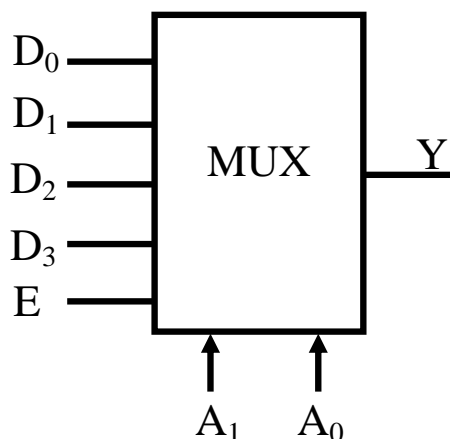
- Gồm: 2^n lối vào dữ liệu, n lối vào địa chỉ, 1 lối vào chọn mạch E và 1 lối ra.
- Tùy theo giá trị của n lối vào địa chỉ mà lối ra sẽ bằng một trong những giá trị ở lối vào (D_j).
- Nếu giá trị thập phân của n lối vào địa chỉ bằng j thì $Y = D_j$.



3.5.1. Mạch hợp kênh – MUX (2)

➤ *Bài toán:* Thiết kế MUX 2 lối vào địa chỉ, một lối vào điều khiển mức tích cực cao.

➤ *Sơ đồ khối:*



E	A1	A0	Y
0	x	x	0
1	0	0	D0
1	0	1	D1
1	1	0	D2
1	1	1	D3

➤ *Bảng trạng thái:*

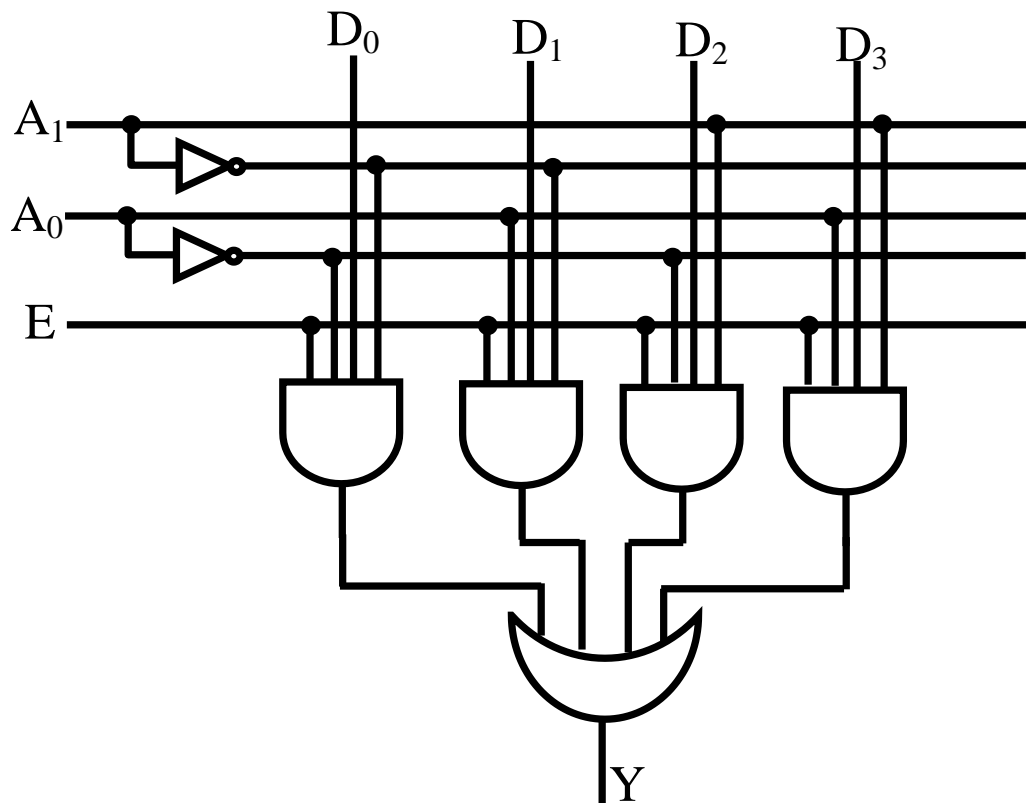
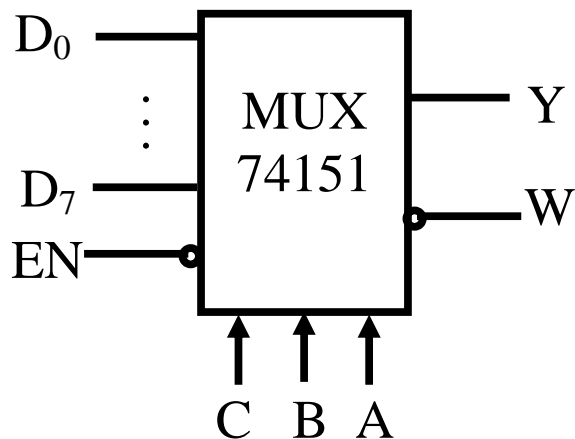
➤ *Biểu thức:*

$$Y = E.(\overline{A_1}.\overline{A_0}.D_0 + \overline{A_1}.A_0.D_1 + A_1.\overline{A_0}.D_2 + A_1.A_0.D_3)$$

3.5.1. Mạch hợp kênh – MUX (3)

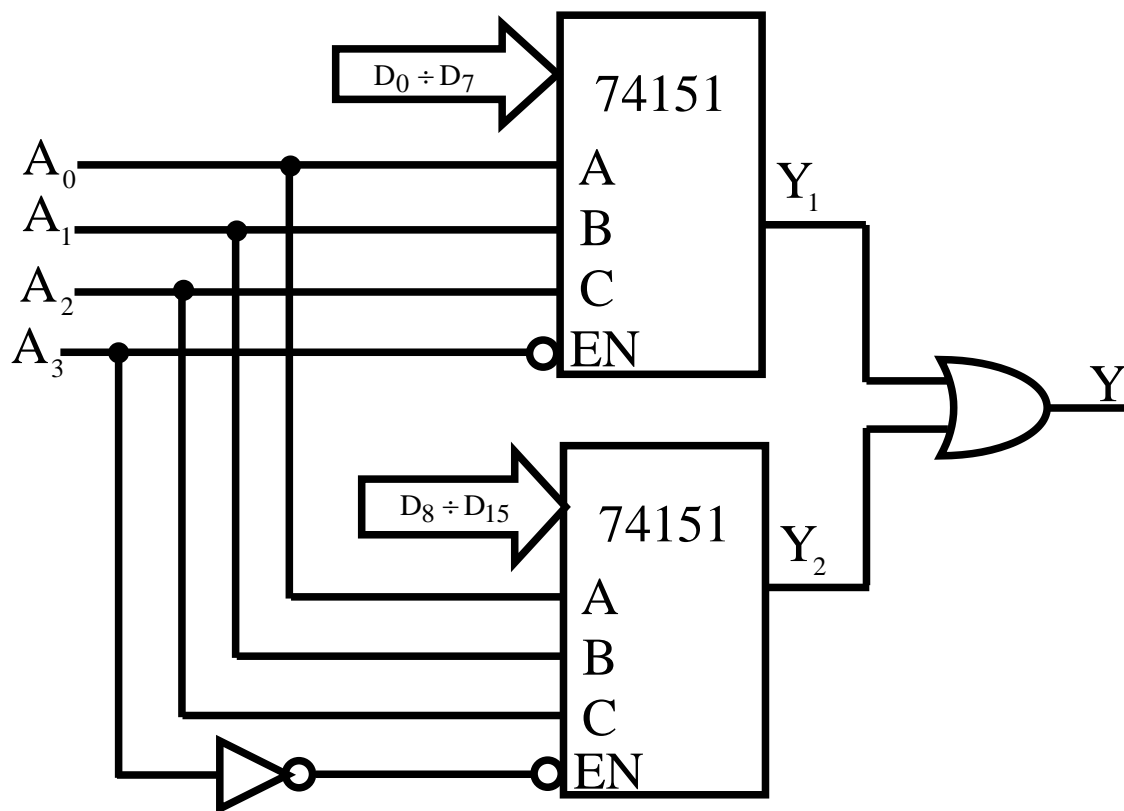
➤ Mạch điện:

➤ IC thực tế: 74151



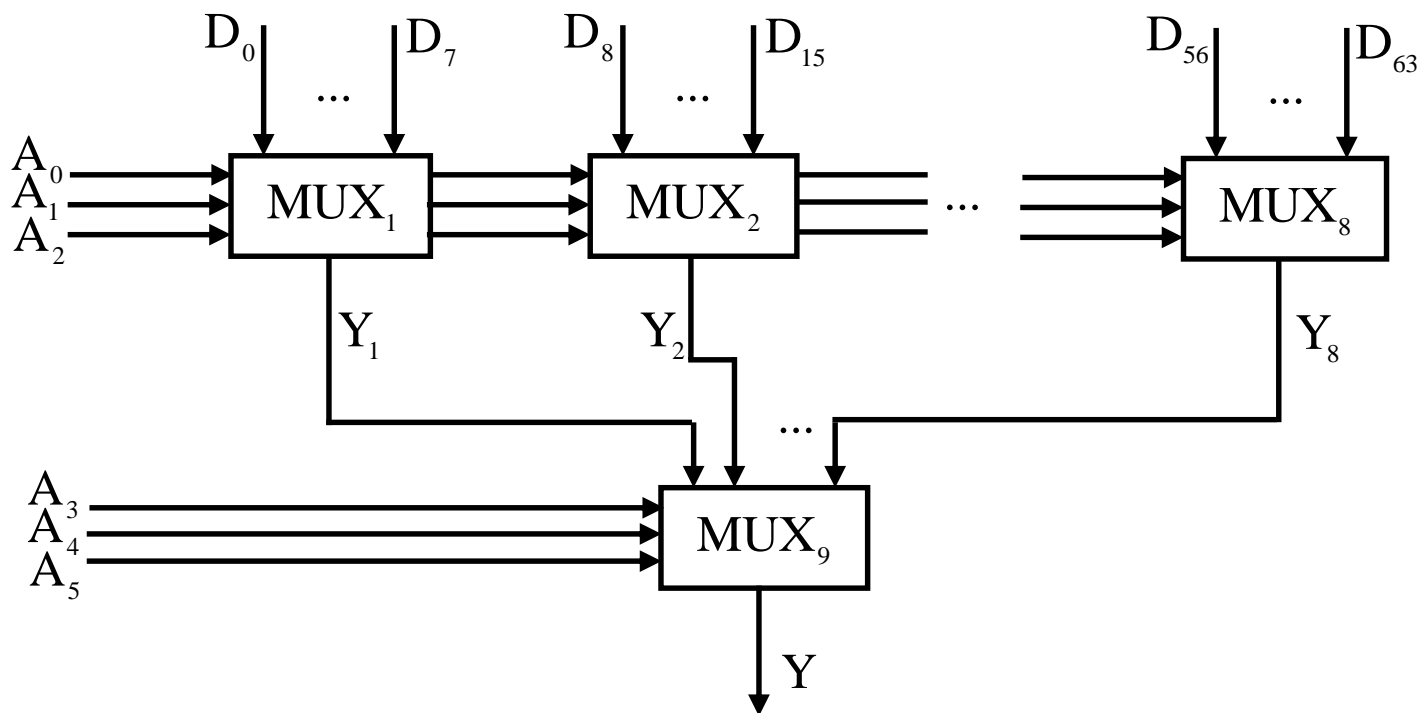
3.5.1. Mạch hợp kênh – MUX (4)

- Mở rộng dung lượng bộ hợp kênh: Sử dụng IC 74151 thực hiện MUX 16 vào – 1 ra.



3.5.1. Mạch hợp kênh – MUX (5)

- Mở rộng dung lượng bộ hợp kênh: Sử dụng IC 74151 thực hiện MUX 64 vào – 1 ra.

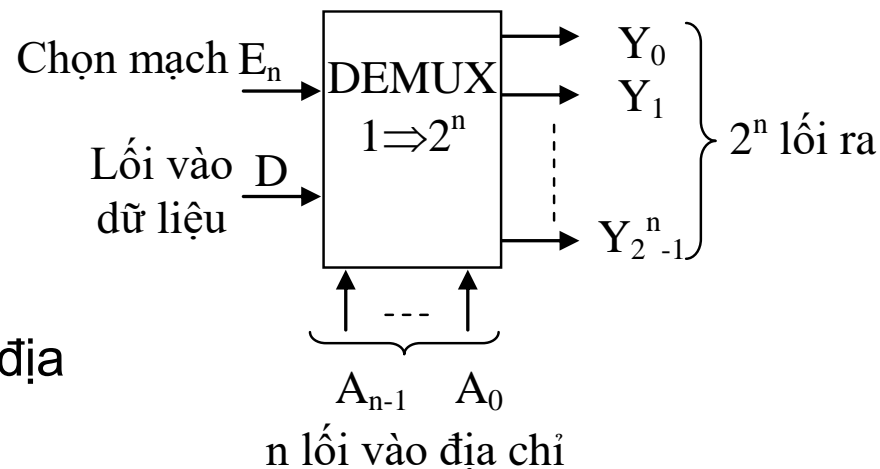


3.5.2. Mạch phân kênh – DEMUX (1)

➤ *Chức năng:* lựa chọn nối một đầu ra với đầu vào dữ liệu.

➤ *Sơ đồ khối:*

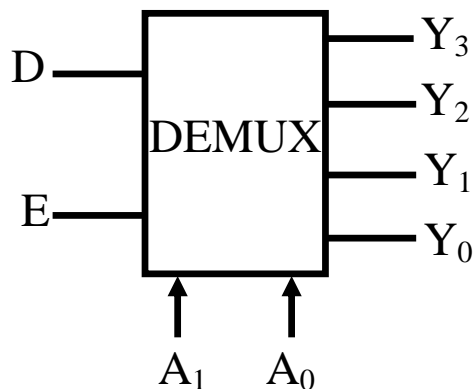
- Gồm: 1 lối vào dữ liệu D , n lối vào địa chỉ, 1 lối vào chọn mạch E và 2^n lối ra.
- Tùy theo giá trị của n lối vào địa chỉ mà một trong số các lối ra sẽ bằng giá trị ở lối vào (D).
- Nếu giá trị thập phân của n lối vào địa chỉ bằng j thì $Y_j = D$.



3.5.2. Mạch phân kênh – DEMUX (2)

➤ *Bài toán:* Thiết kế DEMUX hai lối vào địa chỉ, một lối vào điều khiển mức tích cực cao.

➤ *Sơ đồ khối:*



Bảng trạng thái:

E	A1	A0	Y0	Y1	Y2	Y3
0	x	x	0	0	0	0
1	0	0	D	0	0	0
1	0	1	0	D	0	0
1	1	0	0	0	D	0
1	1	1	0	0	0	D

➤ *Biểu thức:*

$$Y_0 = E \cdot \overline{A_1} \cdot \overline{A_0} \cdot D$$

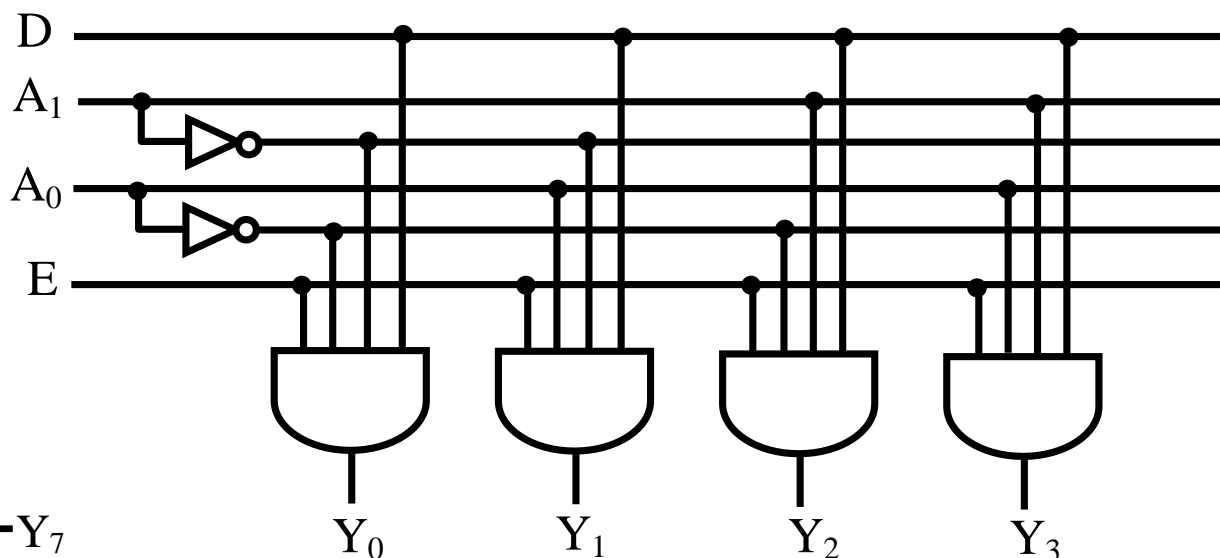
$$Y_1 = E \cdot \overline{A_1} \cdot A_0 \cdot D$$

$$Y_2 = E \cdot A_1 \cdot \overline{A_0} \cdot D$$

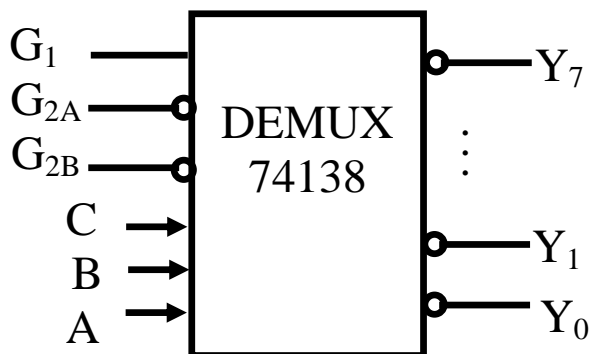
$$Y_3 = E \cdot A_1 \cdot A_0 \cdot D$$

3.5.2. Mạch phân kênh – DEMUX (3)

➤ Mạch điện:

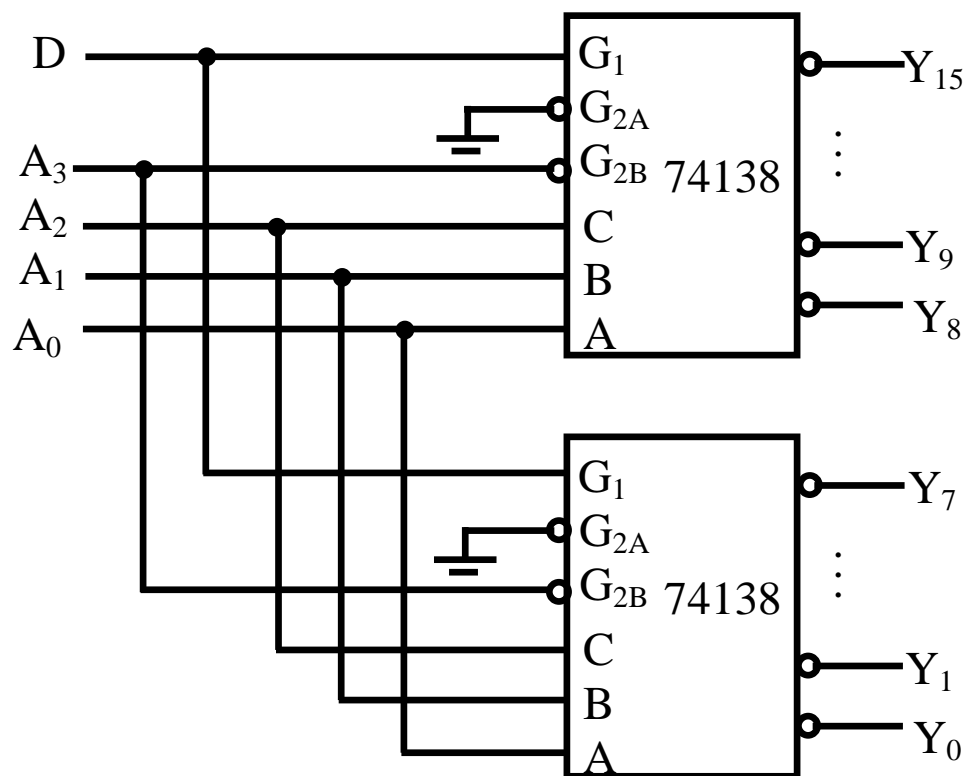


➤ IC thực tế: 74138



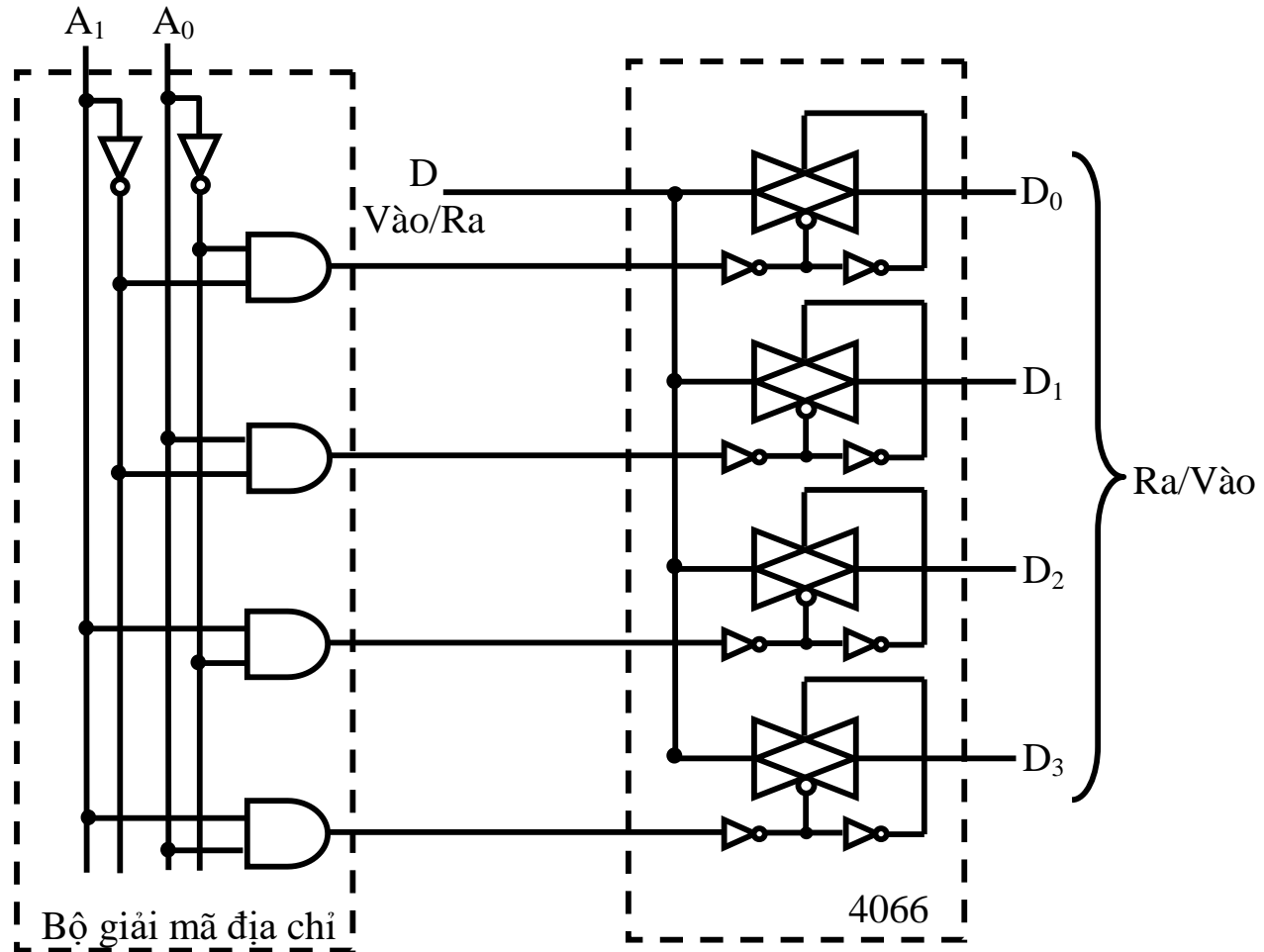
3.5.2. Mạch phân kênh – DEMUX (4)

- Mở rộng dung lượng bộ phân kênh: Sử dụng IC 74138 thực hiện DEMUX 1 vào – 16 ra.



3.5.3. Mạch hợp kênh - phân kênh hỗn hợp

➤ Sơ đồ:

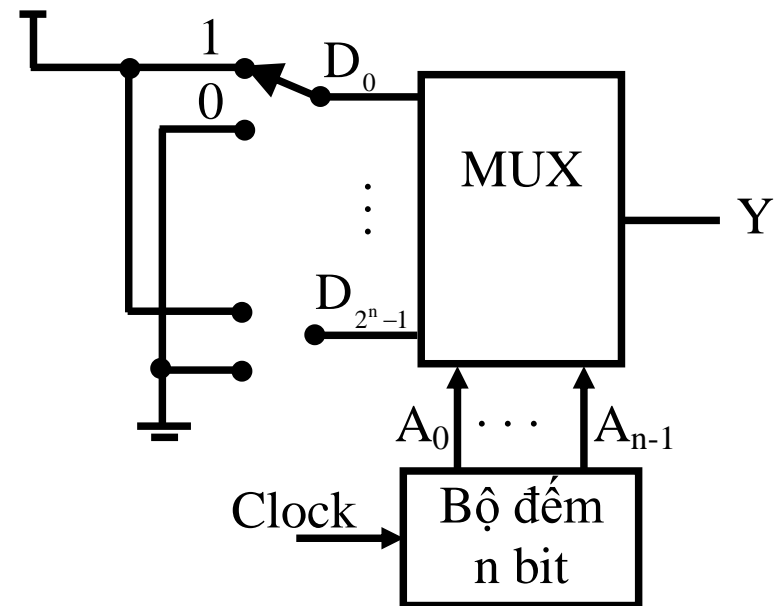


3.5.4. Một số ứng dụng của bộ hợp kênh – phân kênh (1)

- Chuyển đổi luồng dữ liệu từ nối tiếp sang song song và ngược lại.
 - Từ nối tiếp sang song song: 1 đầu vào – nhiều đầu ra, sử dụng IC phân kênh.
 - Từ song song sang nối tiếp: nhiều đầu vào – 1 đầu ra, sử dụng IC hợp kênh.

3.5.4. Một số ứng dụng của bộ hợp kênh – phân kênh (2)

- Tạo dãy nhị phân tuần hoàn: sử dụng MUX để tạo dãy nhị phân tuần hoàn.
- Thay đổi tính chất của dãy:
 - Thay đổi giá trị đầu vào dữ liệu MUX.
 - Thay đổi tần số bộ đếm.



3.5.4. Một số ứng dụng của bộ hợp kênh – phân kênh (3)

- Tạo hàm logic: sử dụng bộ hợp kênh hoặc bộ giải mã địa chỉ chỉ có thể thực hiện được các hàm logic.
- Ví dụ: Thực hiện hàm logic sau $f(A,B,C) = \sum(2,3,5,6)$
sử dụng:
 - a. Bộ giải mã địa chỉ (Decoder)
 - b. Bộ hợp kênh 3 lối vào địa chỉ (Mux 8 – 1)
 - c. Bộ hợp kênh 2 lối vào địa chỉ (Mux 4 – 1)

3.5.4. (4) – Ví dụ: Tạo hàm logic

➤ Bảng trạng thái hàm f : $f(A, B, C) = \sum(2, 3, 5, 6)$

A	B	C	f
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

3.5.4. (5) – Ví dụ: a. Sử dụng Decoder

➤ A, B, C là 3 đầu vào của Decoder.

➤ Bảng trạng thái:

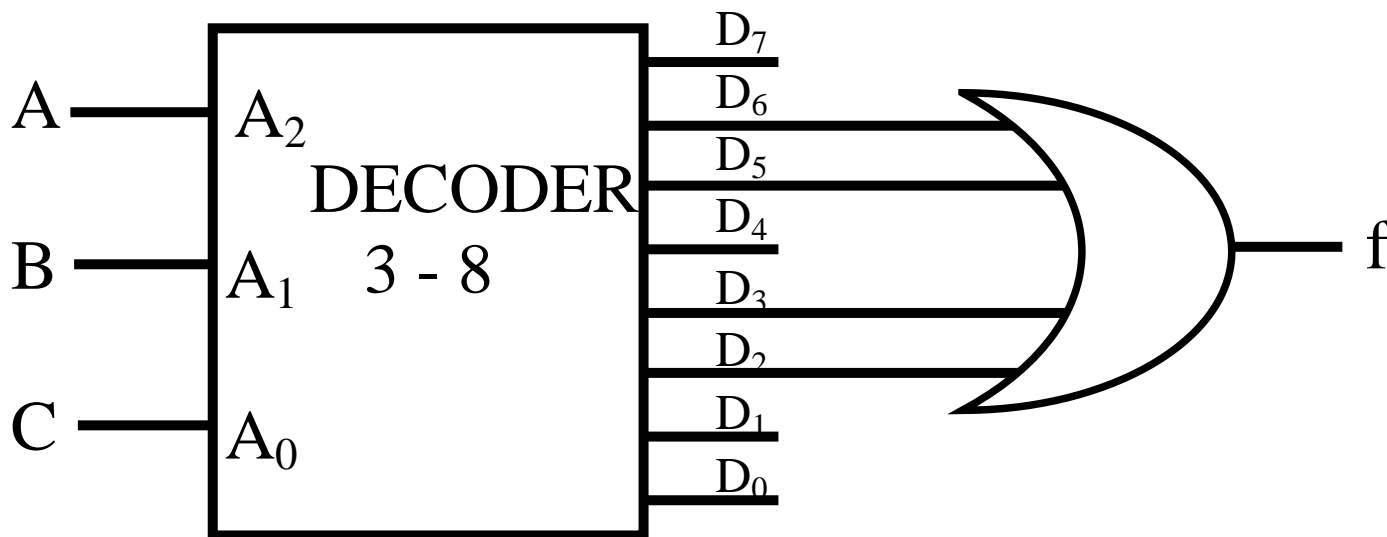
A	B	C	f	Decoder 3 - 8
0	0	0	0	D0 = 1
0	0	1	0	D1 = 1
0	1	0	1	D2 = 1
0	1	1	1	D3 = 1
1	0	0	0	D4 = 1
1	0	1	1	D5 = 1
1	1	0	1	D6 = 1
1	1	1	0	D7 = 1

$$\Rightarrow f(A, B, C) = D_2 + D_3 + D_5 + D_6$$

3.5.4. (6) – Ví dụ: a. Sử dụng Decoder

$$f(A, B, C) = D_2 + D_3 + D_5 + D_6$$

➤ Sơ đồ khối:



3.5.4. (7) – Ví dụ: b. Sử dụng Mux 8 - 1

➤ A, B, C là 3 đầu vào địa chỉ của Mux.

➤ Bảng trạng thái:

A	B	C	f = Y	Mux 8 - 1
0	0	0	0	Y = D0
0	0	1	0	Y = D1
0	1	0	1	Y = D2
0	1	1	1	Y = D3
1	0	0	0	Y = D4
1	0	1	1	Y = D5
1	1	0	1	Y = D6
1	1	1	0	Y = D7

3.5.4. (8) – Ví dụ: b. Sử dụng Mux 8 - 1

➤ Sơ đồ khối:

$D_0 = 0$

$D_1 = 0$

$D_2 = 1$

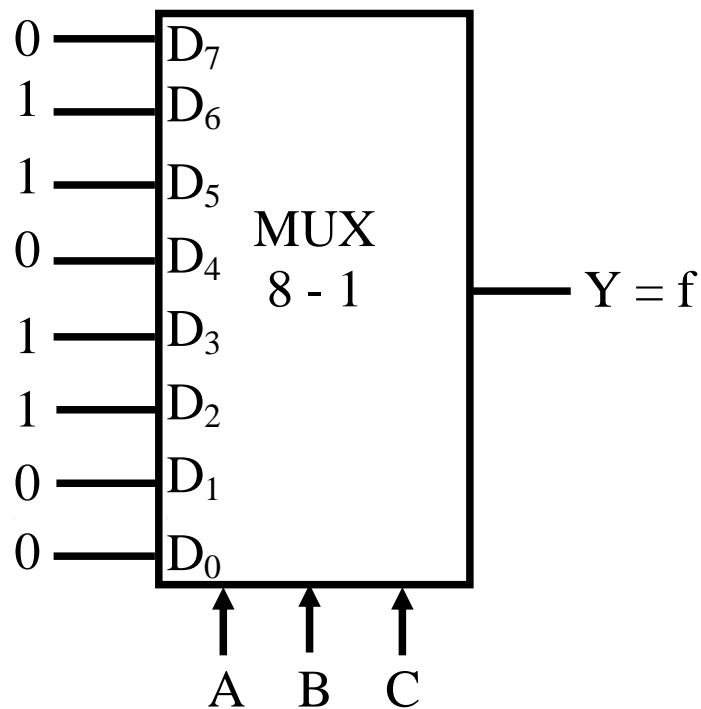
$D_3 = 1$

$D_4 = 0$

$D_5 = 1$

$D_6 = 1$

$D_7 = 0$



3.5.4. (9) – Ví dụ: c. Sử dụng Mux 4 - 1

- Chọn A, B là 2 đầu vào địa chỉ của Mux.
- C là đầu vào dữ liệu.
- Bảng trạng thái:

A	B	C	f = Y	Mux 4 - 1
0	0	0	0	Y = D0
0	0	1	0	
0	1	0	1	Y = D1
0	1	1	1	
1	0	0	0	Y = D2
1	0	1	1	
1	1	0	1	Y = D3
1	1	1	0	

3.5.4. (10) – Ví dụ: c. Sử dụng Mux 4 - 1

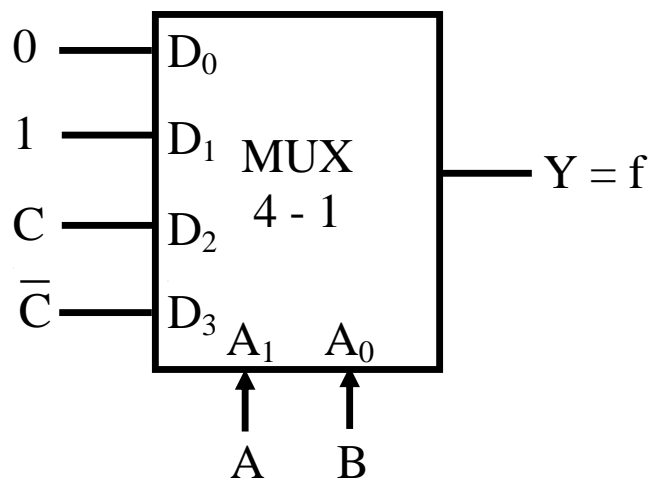
➤ Sơ đồ khối:

$$D_0 = 0$$

$$D_1 = 1$$

$$D_2 = C$$

$$D_3 = \bar{C}$$

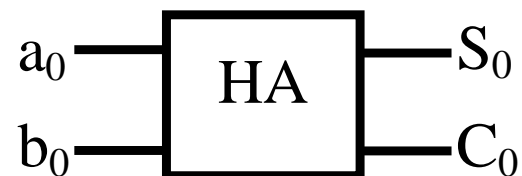


Chương 3 – MẠCH LOGIC TỔ HỢP

- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp
- 3.3. Thiết kế mạch logic tổ hợp
- 3.4. Mạch mã hóa – giải mã
- 3.5. Mạch hợp kênh – phân kênh
- 3.6. Mạch số học**
- 3.7. Mạch so sánh
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ
- 3.9. Mạch tạo/ kiểm tra mã Hamming

3.6.1. Mạch tổng (1) – Mạch bán tổng

➤ Sơ đồ khối:



➤ Bảng trạng thái:

a0	b0	S0	C0
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

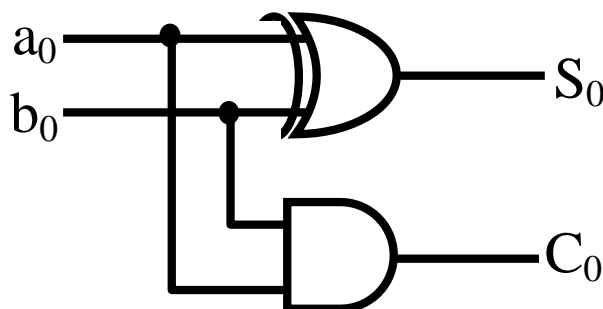
3.6.1. Mạch tổng (2) – Mạch bán tổng

➤ *Biểu thức:*

$$S_0 = a_0 \oplus b_0$$

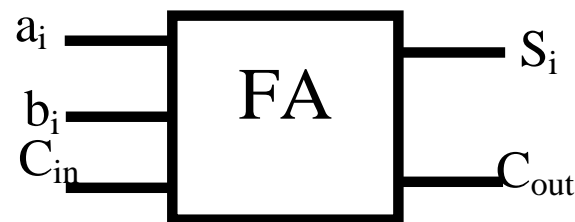
$$C_0 = a_0 \cdot b_0$$

➤ *Mạch điện:*



3.6.1. Mạch tổng (3) – Mạch tổng toàn phần

➤ Sơ đồ khối:



➤ Bảng trạng thái:

C_{in}	a_i	b_i	S_i	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

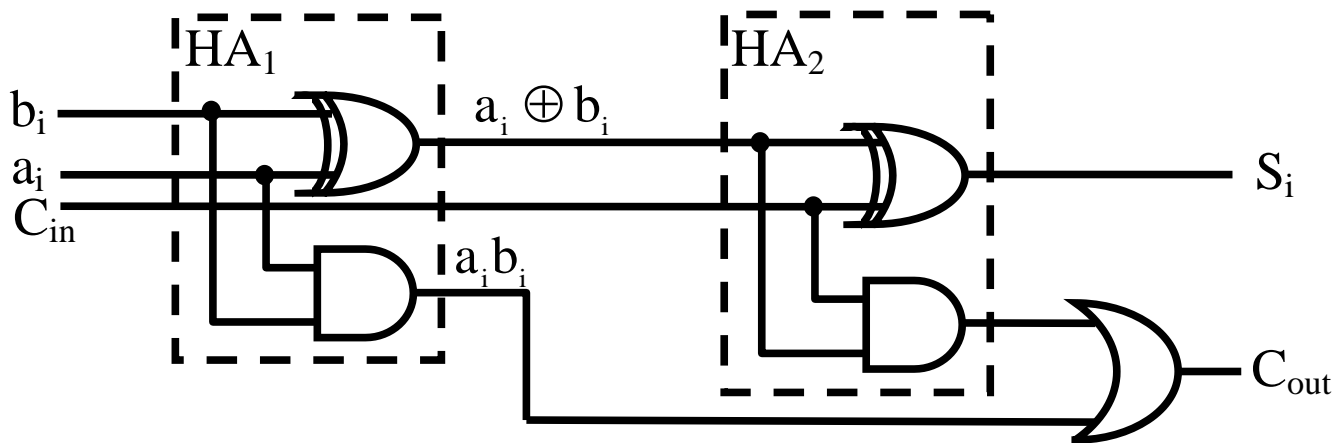
3.6.1. Mạch tổng (4) – Mạch tổng toàn phần

➤ *Biểu thức:*

$$S_i = \overline{a_i} \overline{b_i} \overline{C_{in}} + \overline{a_i} \overline{b_i} C_{in} + \overline{a_i} b_i \overline{C_{in}} + a_i \overline{b_i} \overline{C_{in}} + a_i b_i \overline{C_{in}} + \overline{a_i} b_i C_{in} + a_i \overline{b_i} C_{in} + a_i b_i C_{in} = a_i \oplus b_i \oplus C_{in}$$

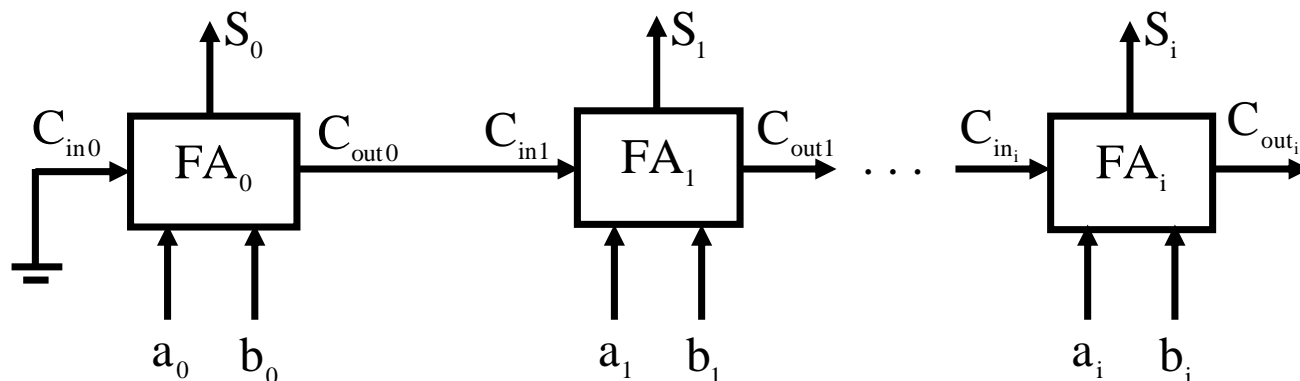
$$C_{out} = \overline{a_i} b_i \overline{C_{in}} + \overline{a_i} b_i C_{in} + a_i \overline{b_i} \overline{C_{in}} + a_i \overline{b_i} C_{in} + a_i b_i \overline{C_{in}} + a_i b_i C_{in} = a_i b_i + (a_i \oplus b_i) C_{in}$$

➤ *Mạch điện:*

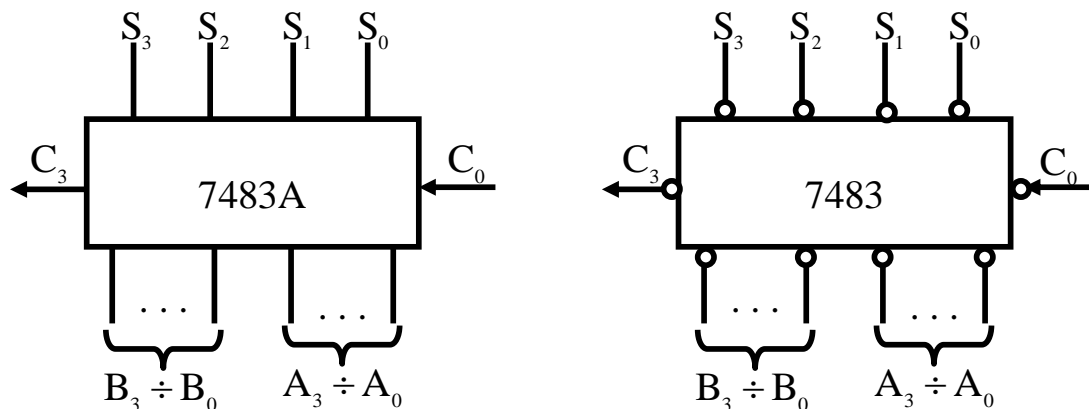


3.6.1. Mạch tổng (5) – Mạch cộng nhị phân song song

➤ Sơ đồ khối:



➤ IC thực tế: 7483, 7483A



3.6.1. Mạch tổng (6) – Mạch cộng nhớ nhanh

➤ Bộ cộng FA một bit:

$$C_i = (A_i \oplus B_i)C_{i-1} + A_i B_i ; S_i = A_i \oplus B_i \oplus C_{i-1}$$

Đặt: $A_i \oplus B_i = P_i$ và $A_i \cdot B_i = G_i$

Ta có: $C_i = P_i C_{i-1} + G_i ; S_i = P_i \oplus C_{i-1}$

➤ Bộ cộng FA 4 bit:

$$C_0 = P_0 C_{-1} + G_0$$

$$C_1 = P_1 C_0 + G_1 = P_1 P_0 C_{-1} + P_1 G_0 + G_1$$

$$C_2 = P_2 C_1 + G_2 = P_2 P_1 P_0 C_{-1} + P_2 P_1 G_0 + P_2 G_1 + G_2$$

$$C_3 = P_3 C_2 + G_3 = P_3 P_2 P_1 P_0 C_{-1} + P_3 P_2 P_1 G_0 + P_3 P_2 G_1 + P_3 G_2 + G_3$$

$$S_0 = P_0 \oplus C_{-1}$$

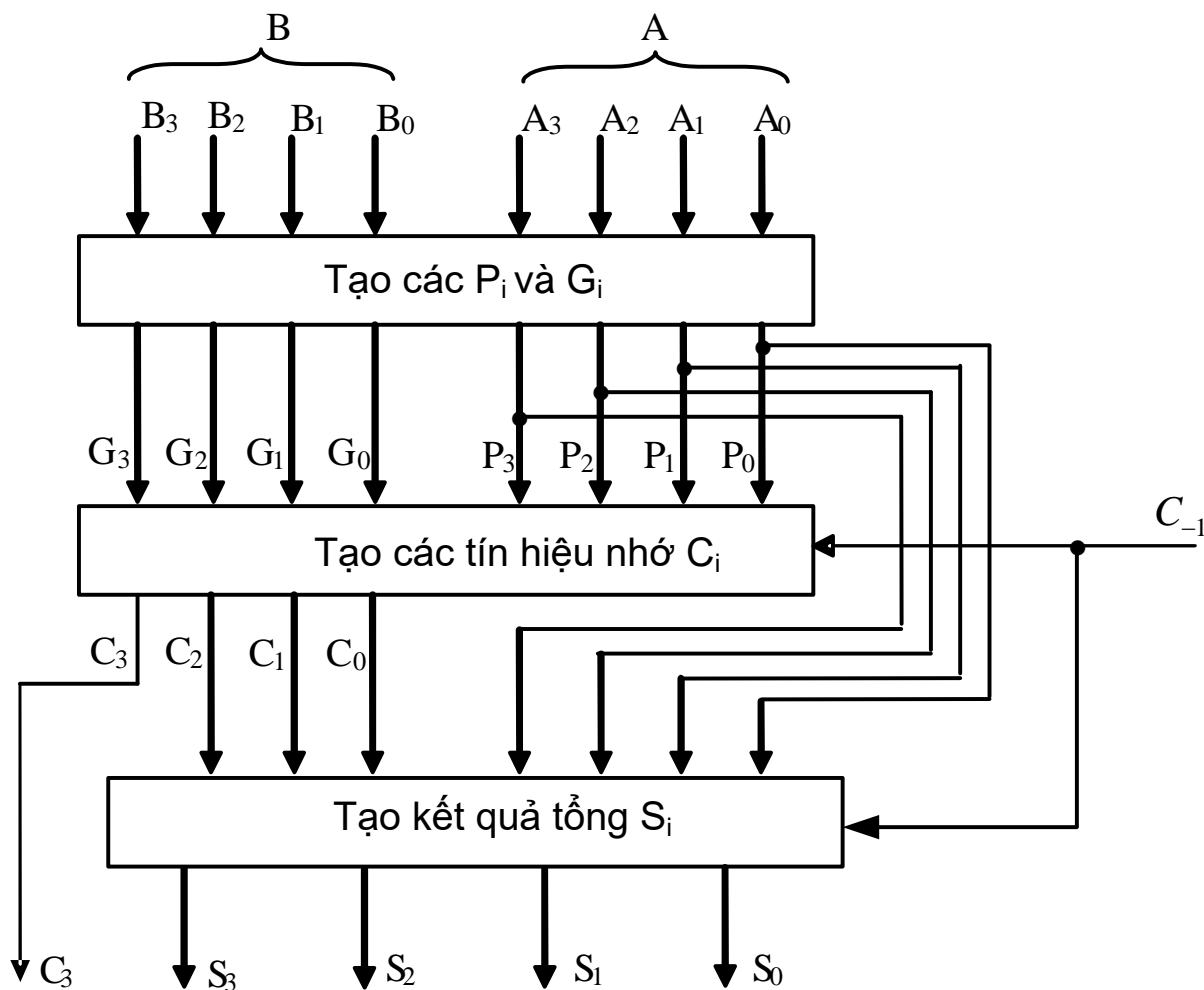
$$S_1 = P_1 \oplus C_0$$

$$S_2 = P_2 \oplus C_1$$

$$S_3 = P_3 \oplus C_2$$

;

3.6.1. Mạch tổng (7) – Mạch cộng nhớ nhanh



3.6.2. Mạch hiệu (1) – Mạch bán hiệu

➤ Sơ đồ khối:



➤ Bảng trạng thái:

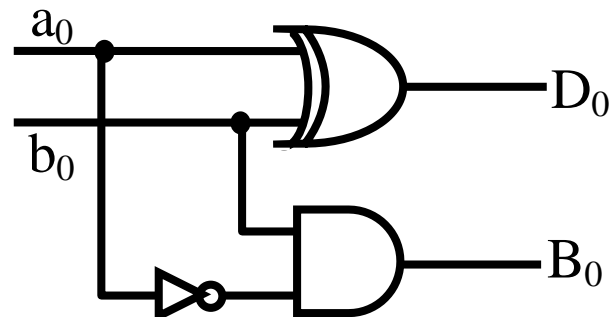
a0	b0	D0	B0
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

3.6.2. Mạch hiệu (2) – Mạch bán hiệu

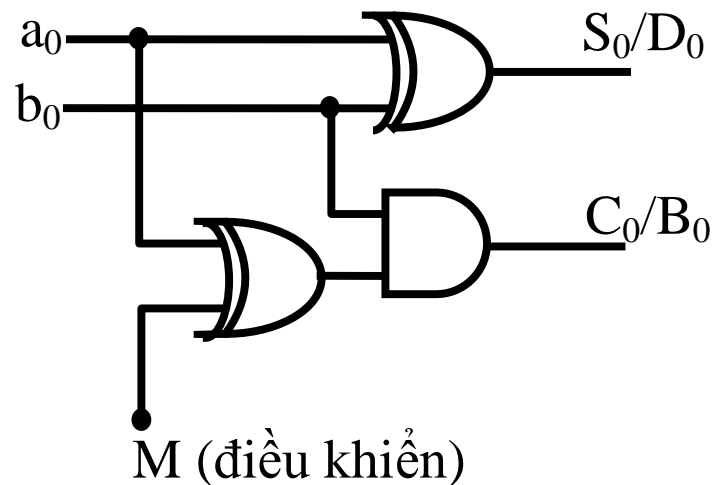
➤ *Biểu thức:*

$$D_0 = a_0 \oplus b_0$$

$$B_0 = \overline{a_0} b_0$$

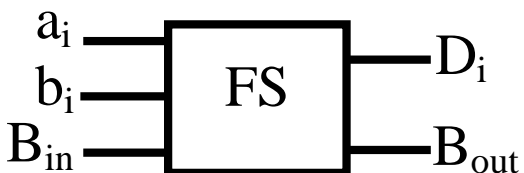


➤ *Mạch điện:*



3.6.2. Mạch hiệu (3) – Mạch hiệu toàn phần

➤ Sơ đồ khối:



➤ Bảng trạng thái:

a_i	b_i	B_{in}	D_i	B_{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

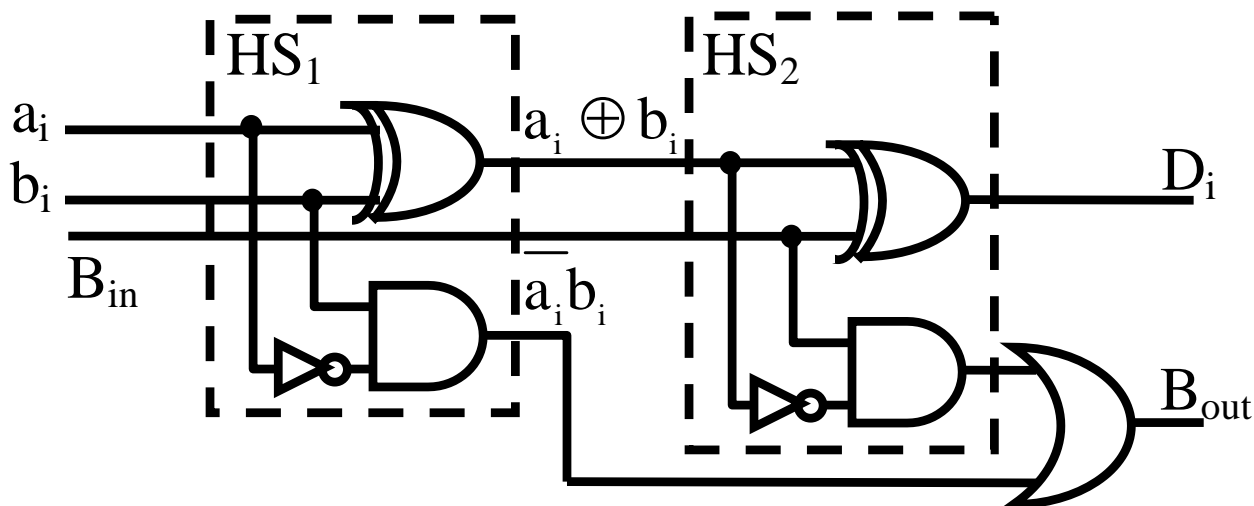
3.6.2. Mạch hiệu (4) – Mạch hiệu toàn phần

➤ *Biểu thức:*

$$D_i = a_i \oplus b_i \oplus B_{in}$$

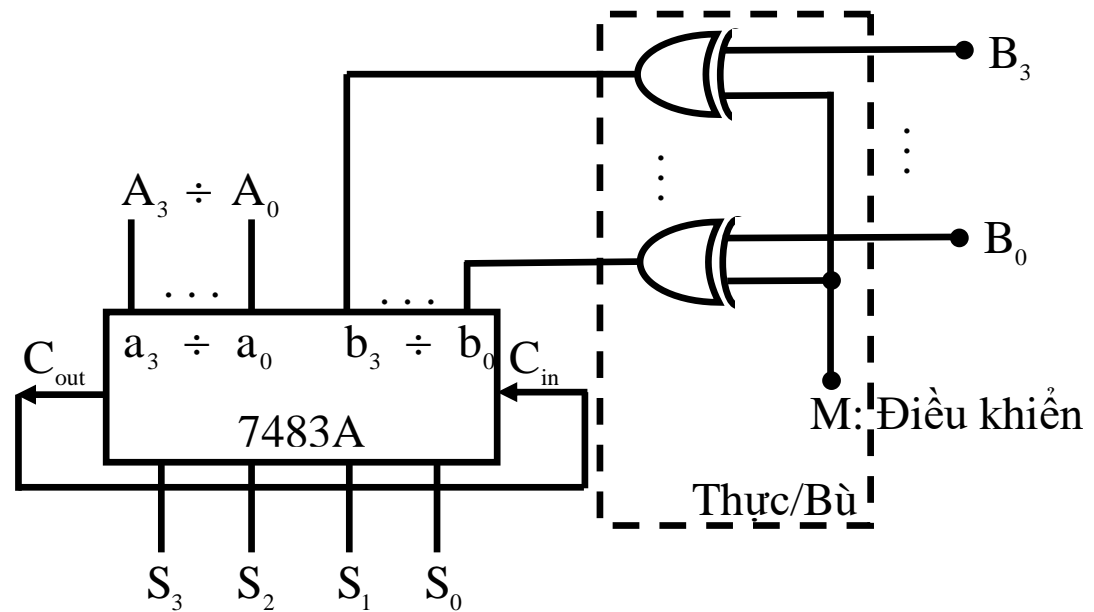
$$B_{out} = \overline{a_i} b_i + (a_i \oplus b_i) B_{in}$$

➤ *Mạch điện:*



3.6.3. Mạch cộng trừ theo phương pháp bù (1) – bù 1

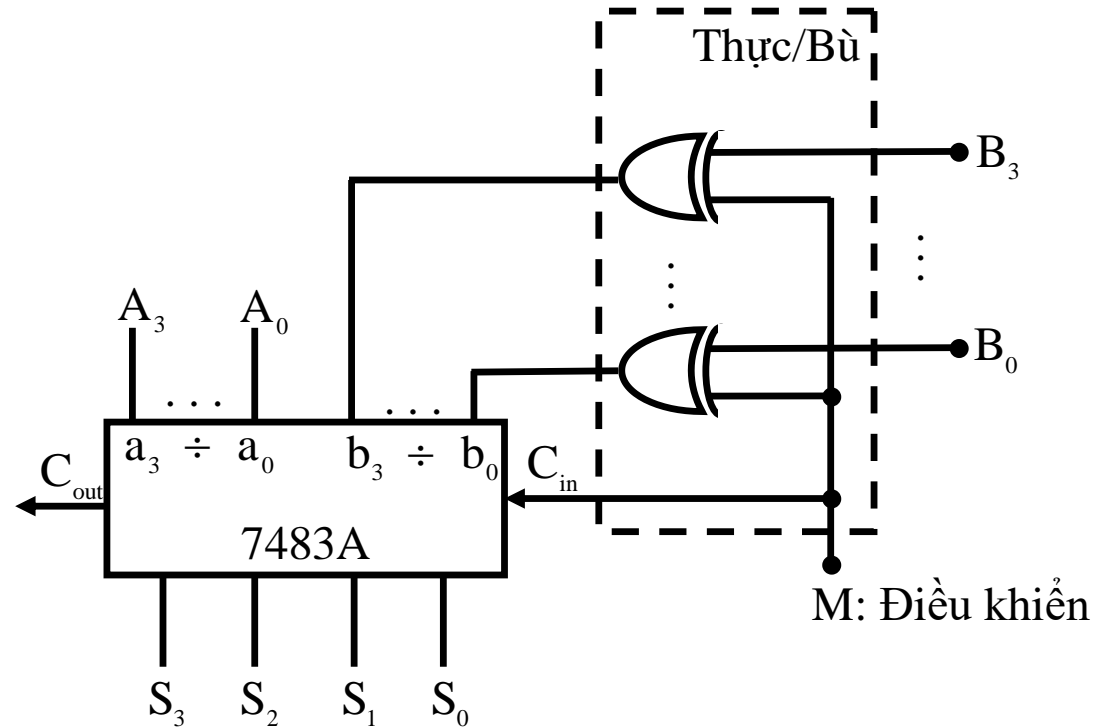
➤ Sơ đồ:



➤ Hoạt động:

3.6.3. Mạch cộng trừ theo phương pháp bù (2) – bù 2

➤ Sơ đồ:



➤ Hoạt động:

3.6.4. Mạch cộng số NBCD (1)

- Trường hợp không phải hiệu chỉnh: Khi kết quả của nhóm đề các tương ứng $S \leq 9$ thì kết quả này là chuẩn BCD.
- Hai trường hợp phải hiệu chỉnh: Khi kết quả của nhóm đề các cùng cấp vượt quá giá trị của số NBCD.

- Khi $9 < S \leq 15$: không có nhớ sang cột tiếp theo.

Hiệu chỉnh bằng cách lấy kết quả trừ đi 10 thập phân (1010) hay cộng với bù 2 của 10 (0110), được kết quả đúng và nhớ 1 lên đề các kế tiếp.

- Khi $15 < S < 19$: có nhớ lên cột tiếp theo.

Hiệu chỉnh bằng cách trừ đi 10 hay cộng với bù 2 của 10 (0110) và nhớ 1 lên đề các kế tiếp.

3.6.4. Mạch cộng số NBCD (2)

➤ Tìm hàm hiệu chỉnh:

- Khi $9 < S \leq 15$:

$$F = S_3S_2 + S_3S_1$$

- Khi $15 < S < 19$: có nhớ.

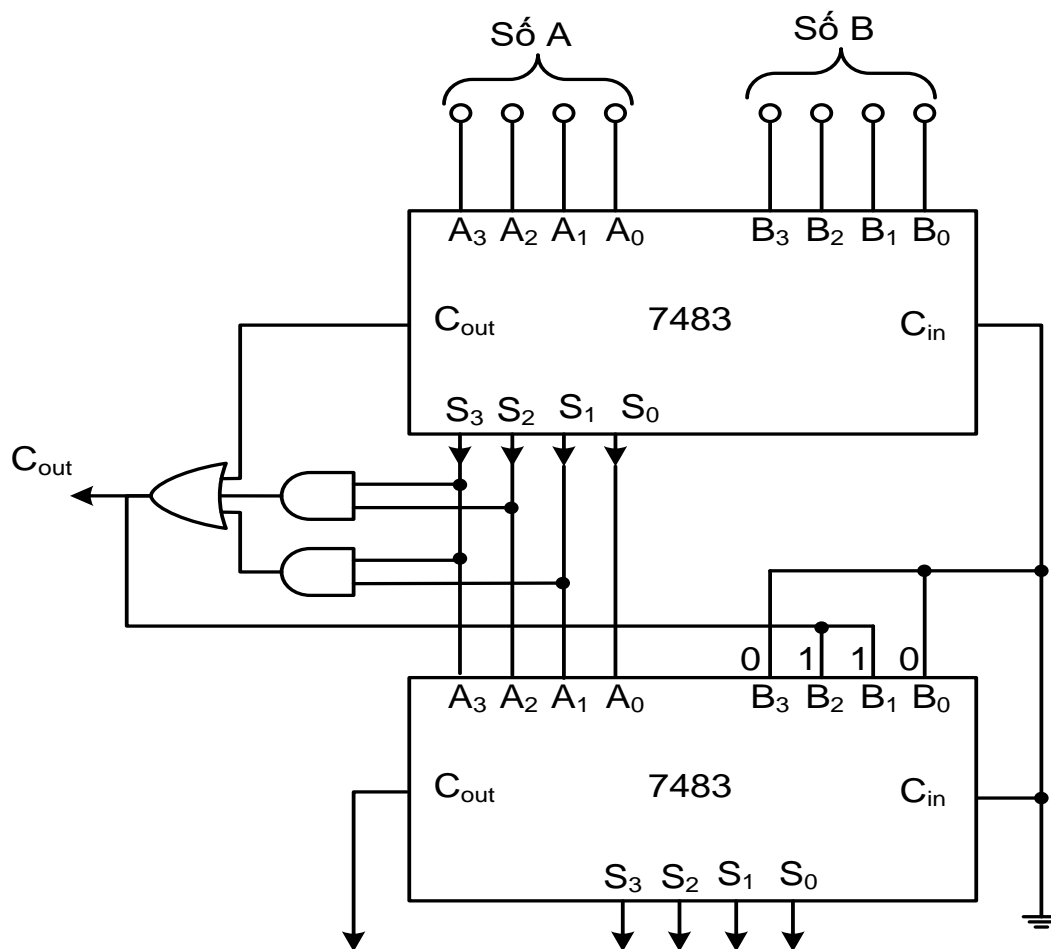
- Hàm hiệu chỉnh chung:

$$F = C_{\text{out}} + S_3S_2 + S_3S_1$$

S_1S_0		00	01	11	10
S_3S_2	00	0	0	0	0
	01	0	0	0	0
	11	1	1	1	1
	10	0	0	1	1

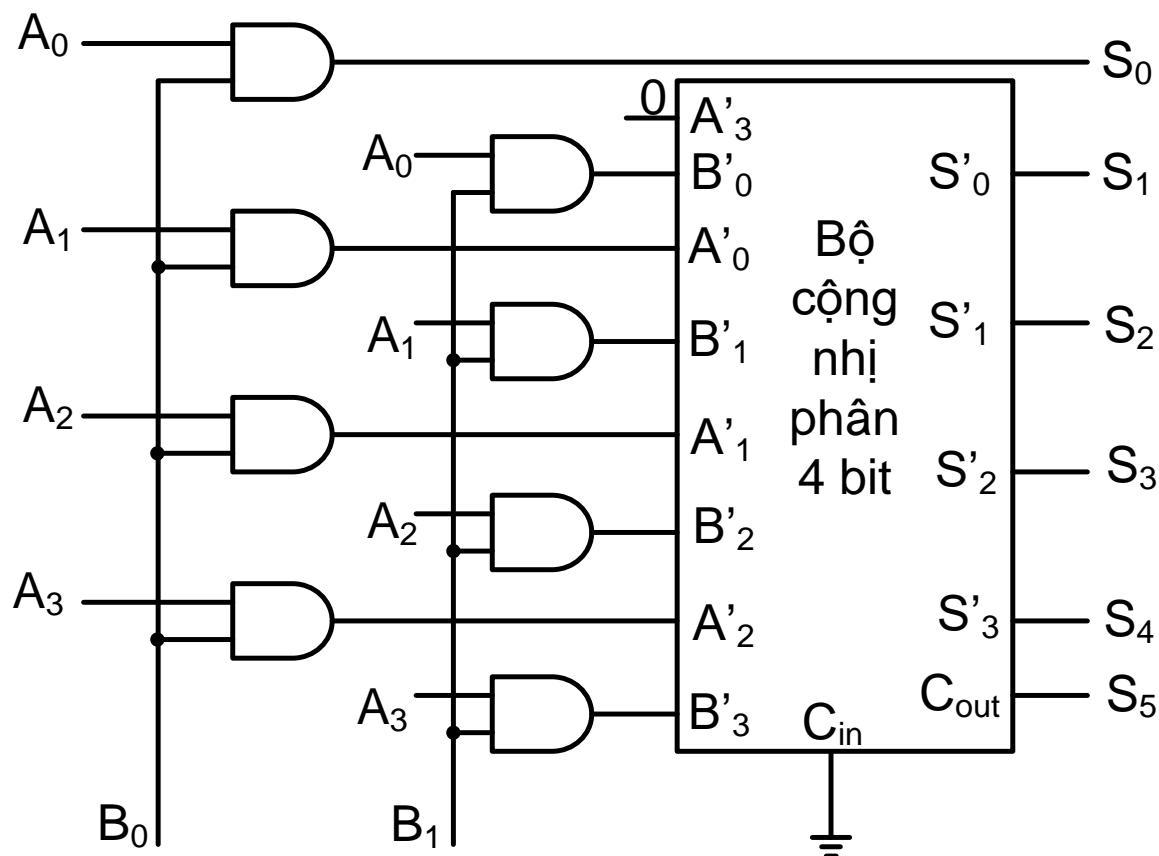
A dashed oval groups the four cells where $S_3S_2 = 11$ (the third row).
 A solid rectangle groups the four cells where $S_3S_1 = 1$ (the third and fourth rows).

3.6.4. Mạch cộng số NBCD (3)



3.6.5. Mạch nhân số nhị phân (1) – Dùng mạch tổ hợp

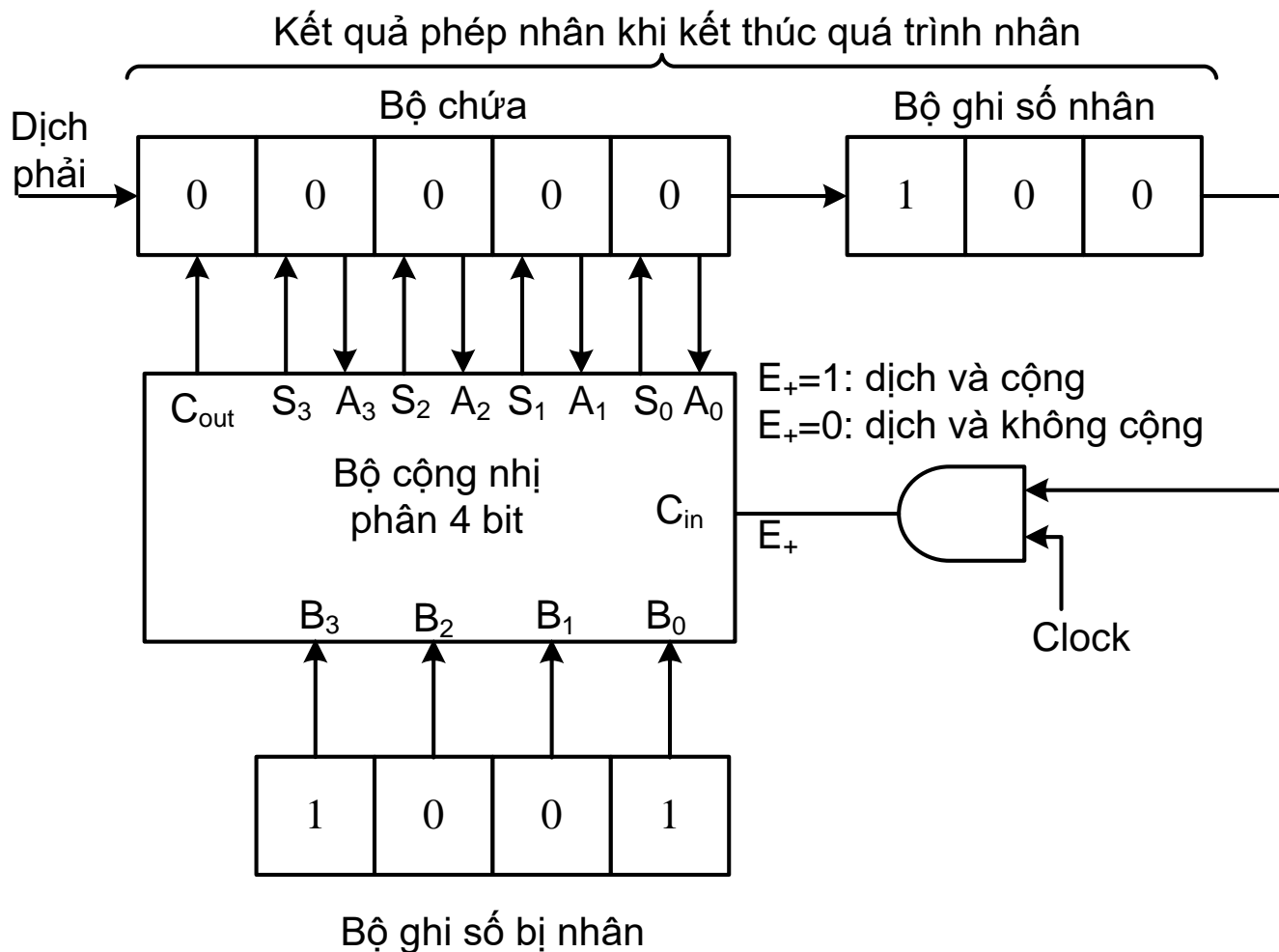
$$A_3A_2A_1A_0 \times B_1B_0 = (A_3A_2A_1A_0 \times B_0) + (A_3A_2A_1A_0 \times B_1 \times 2)$$



3.6.5. Mạch nhân số nhị phân (2) – Dừng mạch tuần tự

- Gồm có các thanh ghi dịch.
 - Thanh ghi chính: ghi cố định số bị nhân.
 - Thanh ghi phụ: chứa các kết quả trung gian trong quá trình thực hiện phép nhân.
- Tại thời điểm ban đầu tất cả các thanh ghi dịch đều bị xóa về 0.
- Sử dụng bộ cộng nhị phân 4 bit và một mạch điều khiển quá trình dịch, cộng.
- Quá trình nhân sẽ dừng lại ở số lần dịch là $(n+1)$ với n là số bit của số nhân.
- Khi kết thúc quá trình nhân thì các thanh ghi làm bộ chứa sẽ chứa kết quả phép nhân.

3.6.5. Mạch nhân số nhị phân (3) – Dùng mạch tuần tự



Chương 3 – MẠCH LOGIC TỔ HỢP

- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp
- 3.3. Thiết kế mạch logic tổ hợp
- 3.4. Mạch mã hóa – giải mã
- 3.5. Mạch hợp kênh – phân kênh
- 3.6. Mạch số học
- 3.7. Mạch so sánh**
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ
- 3.9. Mạch tạo/ kiểm tra mã Hamming

3.7.1. Bộ so sánh bằng nhau

➤ Bộ so sánh bằng nhau 1 bit:

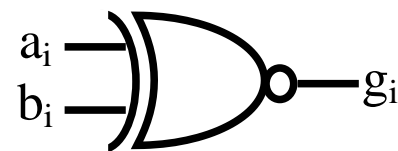
- Bảng trạng thái:

- Biểu thức:

$$g_i = \overline{a_i} \overline{b_i} + a_i b_i = \overline{a_i \oplus b_i}$$

- Mạch điện:

a_i	b_i	g_i
0	0	1
0	1	0
1	0	0
1	1	1



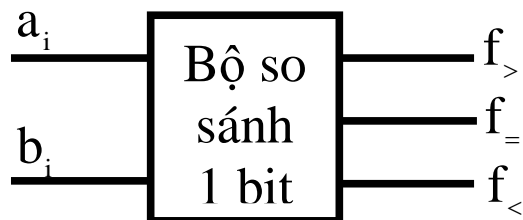
➤ Bộ so sánh bằng nhau 4 bit:

$$A = a_3 a_2 a_1 a_0 \text{ với } B = b_3 b_2 b_1 b_0$$

$$A = B \Leftrightarrow \begin{cases} a_3 = b_3 \\ a_2 = b_2 \\ a_1 = b_1 \\ a_0 = b_0 \end{cases}$$

3.7.2. Bộ so sánh 1 bit

➤ Sơ đồ khối:



➤ Bảng trạng thái:

a_i	b_i	$f_{<}$	$f_{=}$	$f_{>}$
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

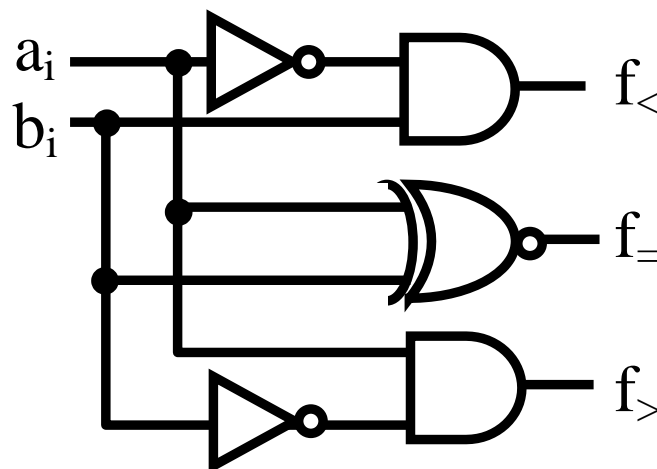
➤ Biểu thức:

$$f_{<} = \overline{a_i} \cdot b_i$$

$$f_{=} = \overline{a_i \oplus b_i}$$

$$f_{>} = a_i \cdot \overline{b_i}$$

➤ Mạch điện:



3.7.3. Bộ so sánh 4 bit (1)

➤ So sánh hai số nhị phân 4 bit $A = a_3a_2a_1a_0$ với $B = b_3b_2b_1b_0$.

$A > B$ khi:

- hoặc $a_3 > b_3$,
- hoặc $a_3 = b_3$, và $a_2 > b_2$,
- hoặc $a_3 = b_3$, và $a_2 = b_2$, và $a_1 > b_1$,
- hoặc $a_3 = b_3$, và $a_2 = b_2$, và $a_1 = b_1$, và $a_0 > b_0$.

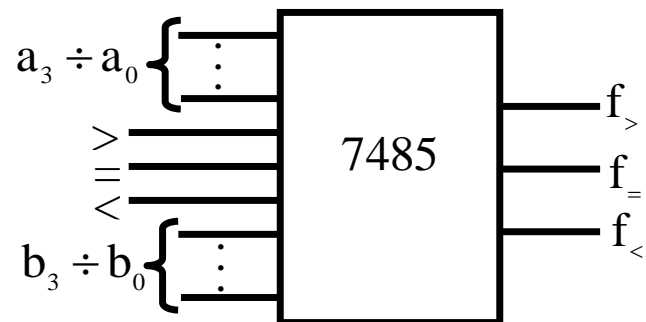
$$f_{>} = a_3 \cdot \overline{b_3} + \overline{a_3} \oplus \overline{b_3} \cdot a_2 \cdot \overline{b_2} +$$

$$\overline{a_3} \oplus \overline{b_3} \cdot \overline{a_2} \oplus \overline{b_2} \cdot a_1 \cdot \overline{b_1} +$$

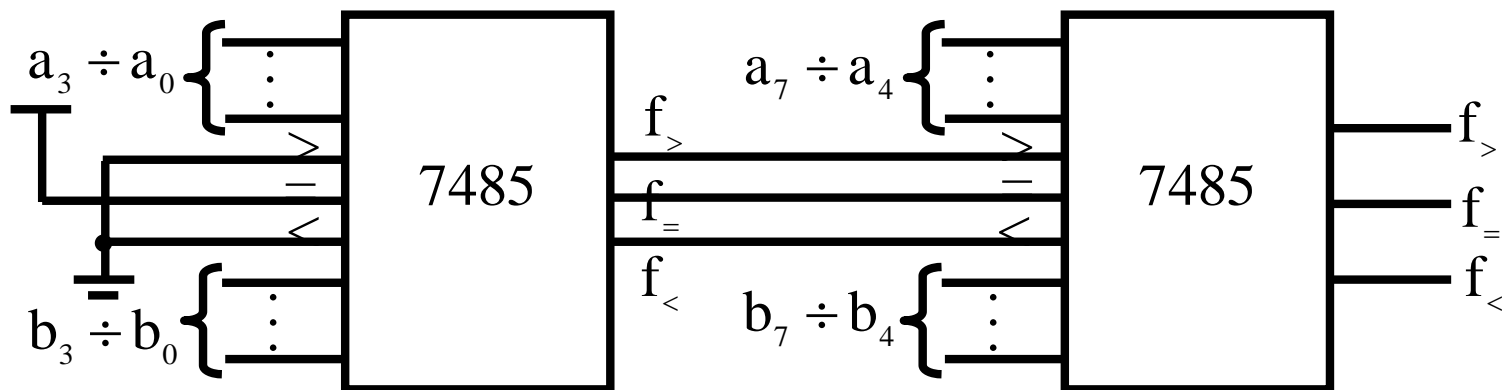
$$\overline{a_3} \oplus \overline{b_3} \cdot \overline{a_2} \oplus \overline{b_2} \cdot \overline{a_1} \oplus \overline{b_1} \cdot a_0 \cdot \overline{b_0}$$

3.7.3. Bộ so sánh 4 bit (2)

➤ IC so sánh: 7485



➤ Mở rộng phạm vi so sánh:



Chương 3 – MẠCH LOGIC TỔ HỢP

- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp
- 3.3. Thiết kế mạch logic tổ hợp
- 3.4. Mạch mã hóa – giải mã
- 3.5. Mạch hợp kênh – phân kênh
- 3.6. Mạch số học
- 3.7. Mạch so sánh
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ**
- 3.9. Mạch tạo/ kiểm tra mã Hamming

3.8.1. Mã chẵn lẻ

➤ Mã chẵn lẻ: là loại mã phát hiện lỗi thông dụng nhất.

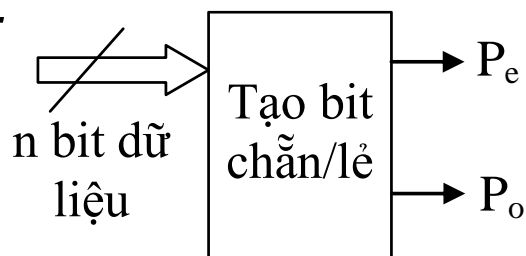
➤ Thiết lập mã chẵn lẻ: thêm một bit chẵn/lẻ (parity bit) vào tổ hợp mã, sao cho:

- Tổng số bit '1' là chẵn (mã chẵn hay tính chẵn)
- Tổng số bit '1' là lẻ (mã lẻ hay tính lẻ)

BCD 8421	BCD 8421chẵn P_C		BCD 8421 lẻ P_L	
0000	0000	0	0000	1
0001	0001	1	0001	0
0010	0010	1	0010	0
0011	0011	0	0011	1
0100	0100	1	0100	0
0101	0101	0	0101	1
0110	0110	0	0110	1
0111	0111	1	0111	0
1000	1000	1	1000	0
1001	1001	0	1001	1

3.8.2. Mạch tạo bit chẵn lẻ

➤ Sơ đồ khối:



➤ Bảng trạng thái:

➤ Biểu thức:

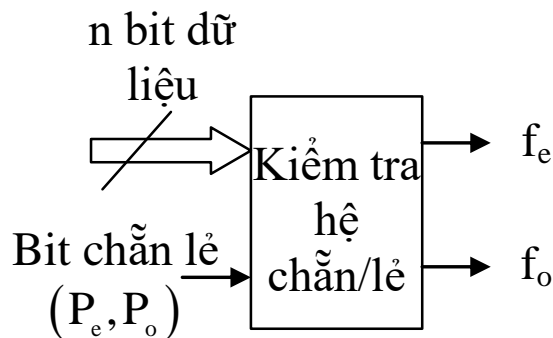
$$p_e = d_1 \oplus d_2 \oplus d_3$$

$$p_o = \overline{p_e} = \overline{d_1 \oplus d_2 \oplus d_3}$$

Vào			Ra	
d_3	d_2	d_1	p_e	p_o
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	0

3.8.3. Mạch kiểm tra chẵn lẻ

➤ Sơ đồ khối:



➤ Bảng trạng thái:

➤ Biểu thức:

$$f_o = d_3 \oplus d_2 \oplus d_1 \oplus p$$

$$f_e = \overline{f_o} = \overline{d_3 \oplus d_2 \oplus d_1 \oplus p}$$

d_3	d_2	d_1	P	f_e	f_o
0	0	0	0	1	0
0	0	0	1	0	1
0	0	1	0	0	1
0	0	1	1	1	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	1	0
0	1	1	1	0	1
1	0	0	0	0	1
1	0	0	1	1	0
1	0	1	0	1	0
1	0	1	1	0	1
1	1	0	0	1	0
1	1	0	1	0	1
1	1	1	0	0	1
1	1	1	1	1	0

Chương 3 – MẠCH LOGIC TỔ HỢP

- 3.1. Khái niệm chung
- 3.2. Phân tích mạch logic tổ hợp
- 3.3. Thiết kế mạch logic tổ hợp
- 3.4. Mạch mã hóa – giải mã
- 3.5. Mạch hợp kênh – phân kênh
- 3.6. Mạch số học
- 3.7. Mạch so sánh
- 3.8. Mạch tạo/ kiểm tra chẵn lẻ
- 3.9. Mạch tạo/ kiểm tra mã Hamming**

3.9.1. Mã Hamming

- Mã Hamming: ghép thêm một số bit kiểm tra P vào thông tin để đảm bảo tính chẵn/lẻ của hệ thống.
- Mã Hamming có khả năng sửa sai một lỗi, có sơ đồ tạo mã và giải mã đơn giản.
- Số bit kiểm tra P và số bit tin tức D phải thỏa mãn biểu thức:

$$D + P + 1 \leq 2^P$$
- Vị trí các bit P: là các bit có thứ tự 2^i tính từ bit có trọng số nhỏ nhất.
- Ví dụ: với số bit tin tức $D = 5$, số bit kiểm tra chẵn/lẻ $P = 4$. Cách sắp xếp vị trí các bit như sau:

9	8	7	6	5	4	3	2	1
D ₅	P ₃	D ₄	D ₃	D ₂	P ₂	D ₁	P ₁	P ₀

(*)

- Bit chẵn lẻ P có nhiệm vụ kiểm tra tính chẵn lẻ của các bit ở các vị trí mà nó chiếm giữ có giá trị 1 tương ứng theo bảng nhị phân (kể cả nó).

3.9.2. Mạch tạo mã Hamming (1)

Bit P_0 kiểm tra tính chẵn/lẻ tại các vị trí :

1, 3, 5, 7, 9, 11, 13, 15

Bit P_1 kiểm tra tính chẵn/lẻ tại các vị trí :

2, 3, 6, 7, 10, 11, 14, 15

Bit P_2 kiểm tra tính chẵn/lẻ tại các vị trí :

4, 5, 6, 7, 12, 13, 14, 15

Bit P_3 kiểm tra tính chẵn/lẻ tại các vị trí :

8, 9, 10, 11, 12, 13, 14, 15

Vị trí	Hệ nhị phân $P_3P_2P_1P_0$
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
10	1 0 1 0
11	1 0 1 1
12	1 1 0 0
13	1 1 0 1
14	1 1 1 0
15	1 1 1 1

3.9.2. Mạch tạo mã Hamming (2) – VD: $D = 5, P = 4$

9	8	7	6	5	4	3	2	1	
D_5	P_3	D_4	D_3	D_2	P_2	D_1	P_1	P_0	(*)

* Tạo mã Hamming chẵn:

$$P_0 \oplus \text{vị trí 3} \oplus \text{vị trí 5} \oplus \text{vị trí 7} \oplus \text{vị trí 9} = 0$$

$$P_0 \oplus D_1 \oplus D_2 \oplus D_4 \oplus D_5 = 0$$

$$P_0 = D_1 \oplus D_2 \oplus D_4 \oplus D_5$$

$$P_1 \oplus D_1 \oplus D_3 \oplus D_4 = 0$$

\Rightarrow

$$P_1 = D_1 \oplus D_3 \oplus D_4$$

$$P_2 \oplus D_2 \oplus D_3 \oplus D_4 = 0$$

$$P_2 = D_2 \oplus D_3 \oplus D_4$$

$$P_3 \oplus D_5 = 0$$

$$P_3 = D_5$$

3.9.2. Mạch tạo mã Hamming (3) – VD: $D = 5, P = 4$

9	8	7	6	5	4	3	2	1	
D_5	P_3	D_4	D_3	D_2	P_2	D_1	P_1	P_0	(*)

* Tạo mã Hamming lẻ:

$$P_0 \oplus D_1 \oplus D_2 \oplus D_4 \oplus D_5 = 1$$

$$P_1 \oplus D_1 \oplus D_3 \oplus D_4 = 1$$

$$P_2 \oplus D_2 \oplus D_3 \oplus D_4 = 1$$

$$P_3 \oplus D_5 = 1$$

\Rightarrow

$$P_0 = \overline{D_1 \oplus D_2 \oplus D_4 \oplus D_5}$$

$$P_1 = \overline{D_1 \oplus D_3 \oplus D_4}$$

$$P_2 = \overline{D_2 \oplus D_3 \oplus D_4}$$

$$P_3 = \overline{D_5}$$

3.9.3. Mạch kiểm tra mã Hamming

Khi nhận được từ mã, tiến hành kiểm tra tính chẵn lẻ trong nhóm bit tại các vị trí:

$$S_0: 1, 3, 5, 7, 9, 11, 13, 15$$

$$S_1: 2, 3, 6, 7, 10, 11, 14, 15$$

$$S_2: 4, 5, 6, 7, 12, 13, 14, 15$$

$$S_3: 8, 9, 10, 11, 12, 13, 14, 15$$

* Kiểm tra mã Hamming: (ví dụ số bit thông tin $D = 5$, số bit kiểm tra $P = 4$)

$$S_0 = \text{vị trí } 1 \oplus \text{vị trí } 3 \oplus \text{vị trí } 5 \oplus \text{vị trí } 7 \oplus \text{vị trí } 9 = 0 \text{ (chẵn)} \quad (= 1 \text{ (lẻ)})$$

$$S_1 = \text{vị trí } 2 \oplus \text{vị trí } 3 \oplus \text{vị trí } 6 \oplus \text{vị trí } 7 = 0 \text{ (chẵn)} \quad (= 1 \text{ (lẻ)})$$

$$S_2 = \text{vị trí } 4 \oplus \text{vị trí } 5 \oplus \text{vị trí } 6 \oplus \text{vị trí } 7 = 0 \text{ (chẵn)} \quad (= 1 \text{ (lẻ)})$$

$$S_3 = \text{vị trí } 8 \oplus \text{vị trí } 9 = 0 \text{ (chẵn)} \quad (= 1 \text{ (lẻ)})$$

- Nếu kết quả thu được khác 0 (chẵn) thì tổ hợp $S_3S_2S_1S_0$ (nhị phân) là vị trí bit lỗi.
- Nếu kết quả thu được khác 1 (lẻ) thì bù 1 của $S_3S_2S_1S_0$ là vị trí bit lỗi.

3.9.4. Ví dụ (1):

Cho từ mã thông tin là 10110, hãy xây dựng từ mã Hamming chẵn.

Giả sử phía thu nhận được chuỗi thông tin là: 110010010.

Kiểm tra lỗi và sửa lỗi (nếu có).

Giải:

-Số bit thông tin: $D = 5$, suy ra số bit kiểm tra: $P = 4$

- Vị trí các bit P :

9	8	7	6	5	4	3	2	1	
D_5	P_3	D_4	D_3	D_2	P_2	D_1	P_1	P_0	(*)

3.9.4. Ví dụ (2) – Xây dựng từ mã Hamming chẵn

➤ Tìm P:

9	8	7	6	5	4	3	2	1
D ₅	P ₃	D ₄	D ₃	D ₂	P ₂	D ₁	P ₁	P ₀
1	P ₃	0	1	1	P ₂	0	P ₁	P ₀

$$P_0 \oplus D_1 \oplus D_2 \oplus D_4 \oplus D_5 = 0$$

$$P_0 = D_1 \oplus D_2 \oplus D_4 \oplus D_5 = 0 \oplus 1 \oplus 0 \oplus 1 = 0$$

$$P_1 \oplus D_1 \oplus D_3 \oplus D_4 = 0$$

$$P_1 = D_1 \oplus D_3 \oplus D_4 = 0 \oplus 1 \oplus 0 = 1$$

$$P_2 \oplus D_2 \oplus D_3 \oplus D_4 = 0$$

$$P_2 = D_2 \oplus D_3 \oplus D_4 = 1 \oplus 1 \oplus 0 = 0$$

$$P_3 \oplus D_5 = 0$$

$$P_3 = D_5 = 1$$

➤ Từ mã Hamming chẵn phát đi:

D ₅	P ₃	D ₄	D ₃	D ₂	P ₂	D ₁	P ₁	P ₀
1	1	0	1	1	0	0	1	0

3.9.4. Ví dụ (3) – Kiểm tra và sửa lỗi

➤ Từ mã nhận được:

b9	b8	b7	b6	b5	b4	b3	b2	b1
1	1	0	0	1	0	0	1	0

➤ Kiểm tra các S:

$$S_0 = b_1 \oplus b_3 \oplus b_5 \oplus b_7 \oplus b_9 = 0 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 0$$

$$S_1 = b_2 \oplus b_3 \oplus b_6 \oplus b_7 = 1 \oplus 0 \oplus 0 \oplus 0 = 1$$

$$S_2 = b_4 \oplus b_5 \oplus b_6 \oplus b_7 = 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

$$S_3 = b_8 \oplus b_9 = 1 \oplus 1 = 0$$

➤ Lỗi ở vị trí 0110 (vị trí 6). Sửa lỗi:

b9	b8	b7	b6	b5	b4	b3	b2	b1
1	1	0	1	1	0	0	1	0

Câu hỏi

- Làm các câu hỏi có nội dung liên quan trong Ngân hàng câu hỏi thi. Trao đổi trên lớp vào buổi học tiếp theo.