



25
SOICT

YEARS ANNIVERSARY

ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

ĐIỆN TỬ CHO CÔNG NGHỆ THÔNG TIN

Electronics for Information Technology

IT3420

Đỗ Công Thuần

Bộ môn Kỹ thuật Máy tính

Email: thuandc@soict.hust.edu.vn

Thông tin chung

- Tên học phần: **Điện tử cho Công nghệ thông tin**
- Mã học phần: IT3420
- Khối lượng: 2 (2-1-0-4)
- Lý thuyết/Bài tập: 30/15 tiết
- Đánh giá: 50% - 50%
- Tài liệu học tập:
 - Lecture slides
 - Textbooks
 - *Introductory Circuit Analysis* (2015), 10th – 13th ed., Robert L. Boylestad
 - *Electronic Device and Circuit Theory* (2013), 11th ed., Robert L. Boylestad, Louis Nashelsky
 - *Microelectronics Circuit Analysis and Design* (2006), 4th ed., Donald A. Neamen
 - *Digital Electronics: Principles, Devices and Applications* (2007), Anil K. Maini

Nội dung

- Khái niệm chung về ĐT cho CNTT
- Chương 1: Linh kiện thụ động và ứng dụng
- Chương 2: Linh kiện bán dẫn và ứng dụng
- Chương 3: Khuếch đại thuật toán
- Chương 4: Cơ sở lý thuyết mạch số
- Chương 5: Các cổng logic cơ bản
- Chương 6: Mạch tổ hợp
- Chương 7: Mạch dãy

Chương 6:

Các cổng logic cơ bản

1. Khái niệm
2. Một số hệ tổ hợp cơ bản
 - Các mạch số học cơ bản
 - Bộ chọn kênh
 - Bộ mã hóa
 - Bộ phân kênh
 - Bộ giải mã

Bài giảng có sử dụng hình vẽ, text từ các tài liệu tham khảo:

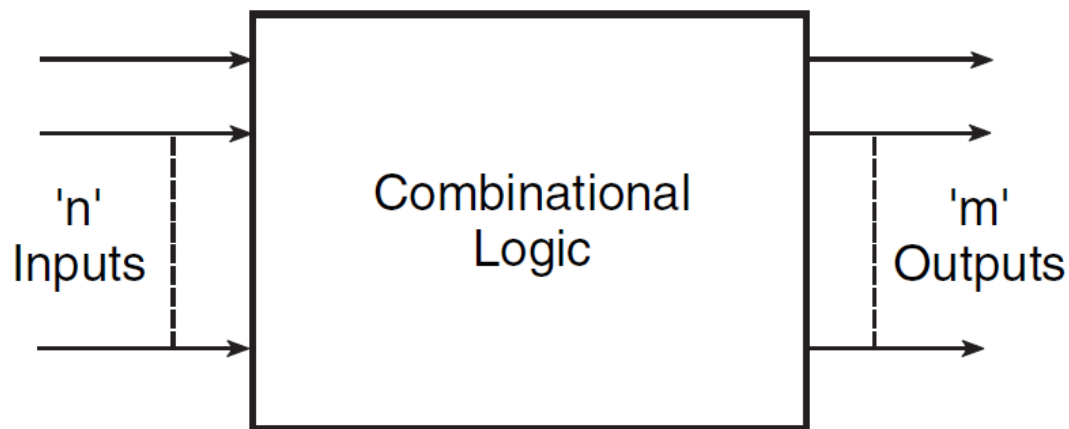
***Digital electronics: Principles, Devices, and Applications, Anil Kumar Maini 2007
John Wiley & Sons***

Mục tiêu

- Nắm vững phương pháp thiết kế mạch tổ hợp
- Hiểu và thiết kế được các mạch tổ hợp cơ bản bao gồm các mạch cộng/trừ/so sánh, các bộ mã hoá/giải mã, phân kênh/chọn kênh
- Xây dựng các mạch logic sử dụng bộ phân kênh/mã hoá

Khái niệm

- Hệ tổ hợp là hệ mà tín hiệu ra chỉ phụ thuộc vào tín hiệu vào tại thời điểm hiện tại
- Hệ tổ hợp còn được gọi là hệ không có nhớ
- Hệ tổ hợp chỉ cần thực hiện bằng những phần tử logic cơ bản



Các bước để xây dựng một hệ tổ hợp

1. Xác định yêu cầu bài toán
2. Xác định các biến đầu vào và đầu ra
3. Mô tả mối quan hệ giữa các biến đầu vào và đầu ra
4. Xây dựng bảng thật đáp ứng được các yêu cầu của đầu vào và đầu ra
5. Viết hàm Boolean cho các biến đầu ra dựa trên các biến đầu vào
6. Tối thiểu hóa hàm Boolean
7. Thực hiện mạch theo hàm tối thiểu hóa Boolean

Các quy tắc khi triển khai phần cứng

1. Sử dụng số lượng cổng ít nhất, với các cổng sử dụng đầu vào tối thiểu nhất
2. Số lượng kết nối là ít nhất, thời gian trễ truyền là nhỏ nhất
3. Hạn chế trong thời gian chuyển mạch của các cổng không nên bị bỏ qua.

Một số hệ tổ hợp cơ bản

- Các mạch số học cơ bản
- Bộ chọn kênh
- Bộ mã hóa
- Bộ phân kênh
- Bộ giải mã

Các mạch số học cơ bản

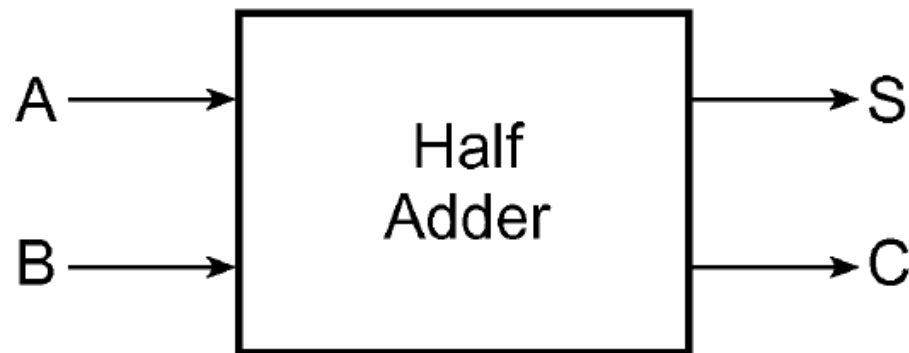
- a. Bộ cộng
- b. Bộ trừ
- c. Bộ so sánh

a. Bộ cộng

- Chức năng: thực hiện phép cộng giữa 2 số nhị phân.
- Gồm có:
 - Bộ bán tổng (Half-Adder)
 - Bộ cộng đầy đủ (Full-Adder)
 - Bộ cộng nhiều bit

a. Bộ cộng \rightarrow Bộ bán tổng (Half-Adder)

- Thực hiện phép cộng giữa 2 bit thấp nhất của phép cộng 2 số nhị phân.
- Sơ đồ khối:



a. Bộ cộng \rightarrow Bộ bán tổng (Half-Adder)

- Bảng thật:

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

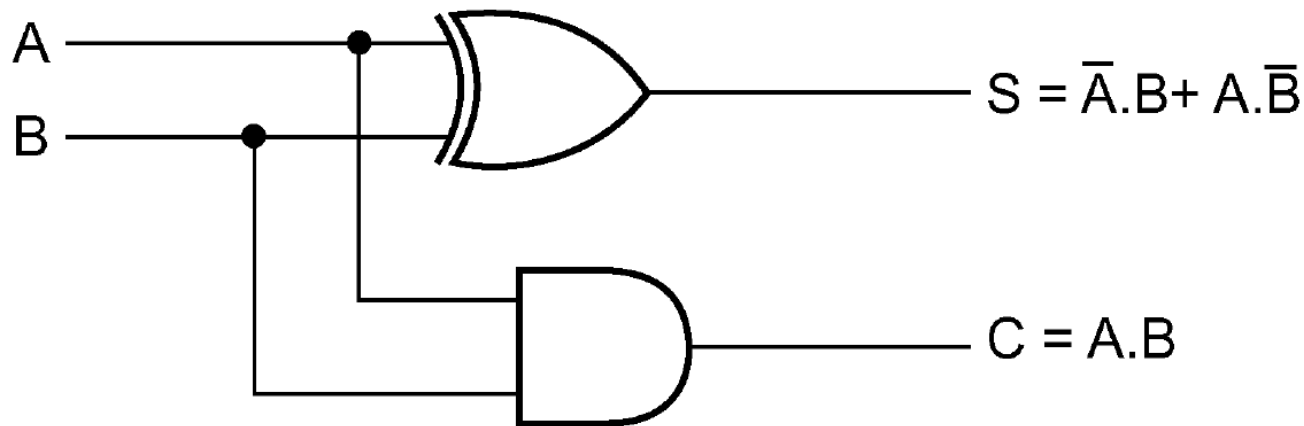
- Biểu thức đầu ra phụ thuộc đầu vào:

$$\text{SUM } S = A.\bar{B} + \bar{A}.B$$

$$\text{CARRY } C = A.B$$

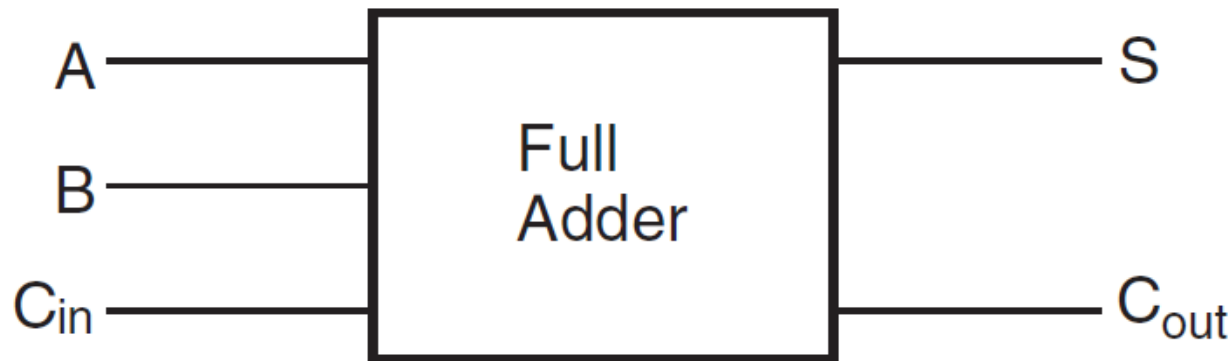
a. Bộ cộng \rightarrow Bộ bán tổng (Half-Adder)

- Sơ đồ mạch: sử dụng một cổng XOR cho S và một cổng AND cho C.



a. Bộ cộng \rightarrow Bộ cộng đầy đủ (Full-Adder)

- Thực hiện phép cộng giữa 2 bit bất kỳ của phép cộng 2 số nhị phân.
- Sơ đồ khối:



a. Bộ cộng \rightarrow Bộ cộng đầy đủ (Full-Adder)

- Bảng thật:

A	B	C_{in}	SUM (S)	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

- Biểu thức đầu ra phụ thuộc đầu vào:

$$S = \overline{A}.\overline{B}.C_{in} + \overline{A}.B.\overline{C}_{in} + A.\overline{B}.\overline{C}_{in} + A.B.C_{in}$$

$$C_{out} = \overline{A}.B.C_{in} + A.\overline{B}.C_{in} + A.B.\overline{C}_{in} + A.B.C_{in}$$

a. Bộ cộng \rightarrow Bộ cộng đầy đủ (Full-Adder)

- Tối thiểu hóa hàm Boolean:
 - Lập bìa Các-nô cho C_{out} và Sum:

AB \ C_{in}	$\overline{C_{in}}$	C_{in}
$\overline{A}\overline{B}$		
$\overline{A}B$		1
AB	1	1
$A\overline{B}$		1

AB \ C_{in}	$\overline{C_{in}}$	C_{in}
$\overline{A}\overline{B}$		1
$\overline{A}B$	1	
AB		1
$A\overline{B}$	1	

- Hàm tối thiểu hóa:

$$C_{out} = A.B + C_{in}.(\overline{A}.B + A.\overline{B})$$

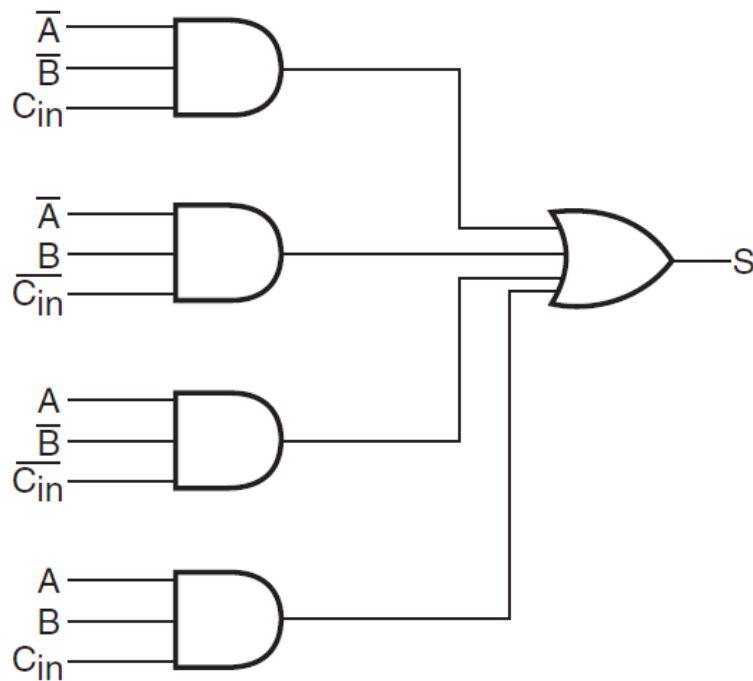
$$S = \overline{C_{in}}.(\overline{A}.B + A.\overline{B}) + C_{in}.(A.B + \overline{A}.\overline{B})$$

$$S = \overline{C_{in}}.(\overline{A}.B + A.\overline{B}) + C_{in}.(\overline{\overline{A}.B + A.\overline{B}})$$

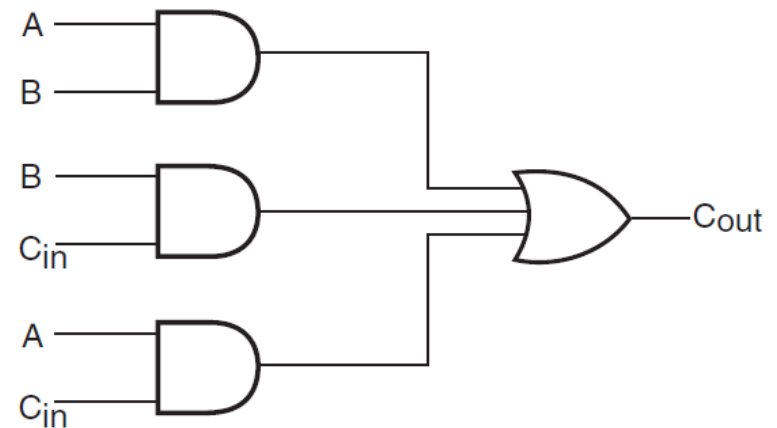
a. Bộ cộng → Bộ cộng đầy đủ (Full-Adder)

- Sơ đồ mạch:

$$S = \overline{C_{in}} \cdot (\overline{A} \cdot B + A \cdot \overline{B}) + C_{in} \cdot (A \cdot B + \overline{A} \cdot \overline{B})$$



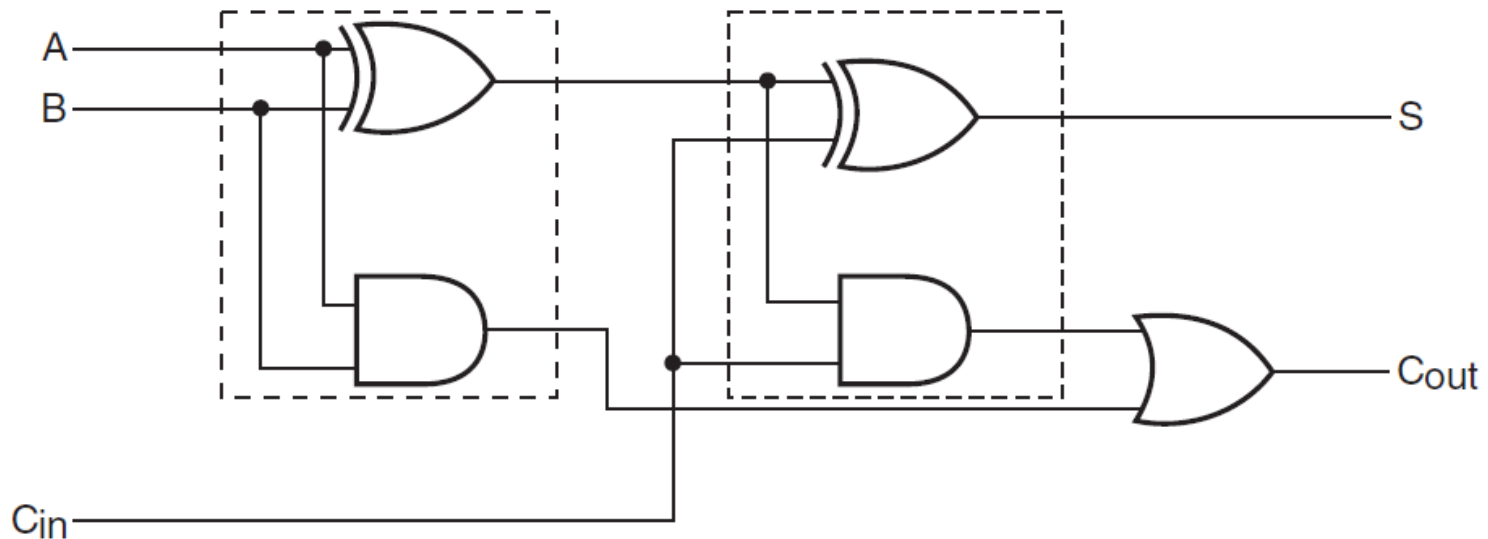
$$C_{out} = A \cdot B + C_{in} \cdot (\overline{A} \cdot B + A \cdot \overline{B})$$



a. Bộ cộng \rightarrow Bộ cộng đầy đủ (Full-Adder)

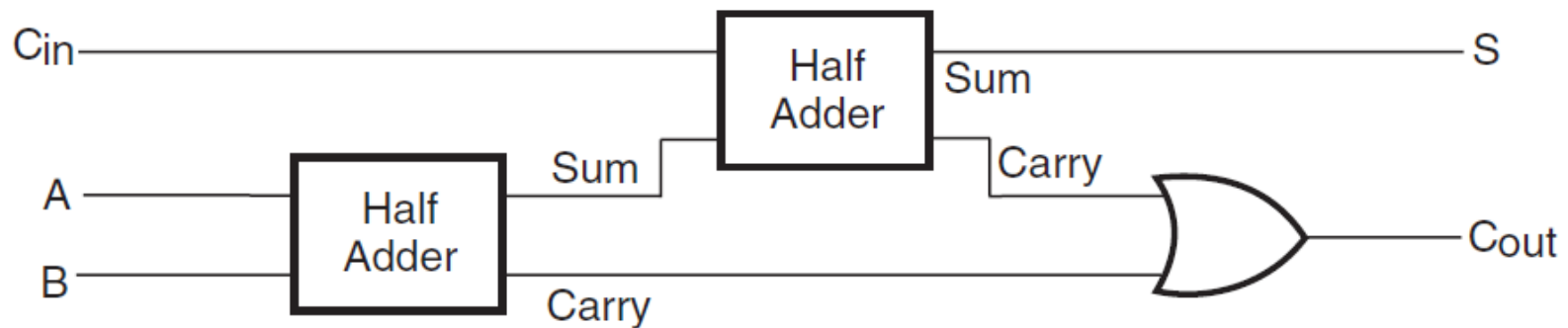
$$S = \overline{C}_{in} \cdot (\overline{A} \cdot B + A \cdot \overline{B}) + C_{in} \cdot (\overline{\overline{A} \cdot B + A \cdot \overline{B}})$$

$$C_{out} = A \cdot B + C_{in} \cdot (\overline{A} \cdot B + A \cdot \overline{B})$$



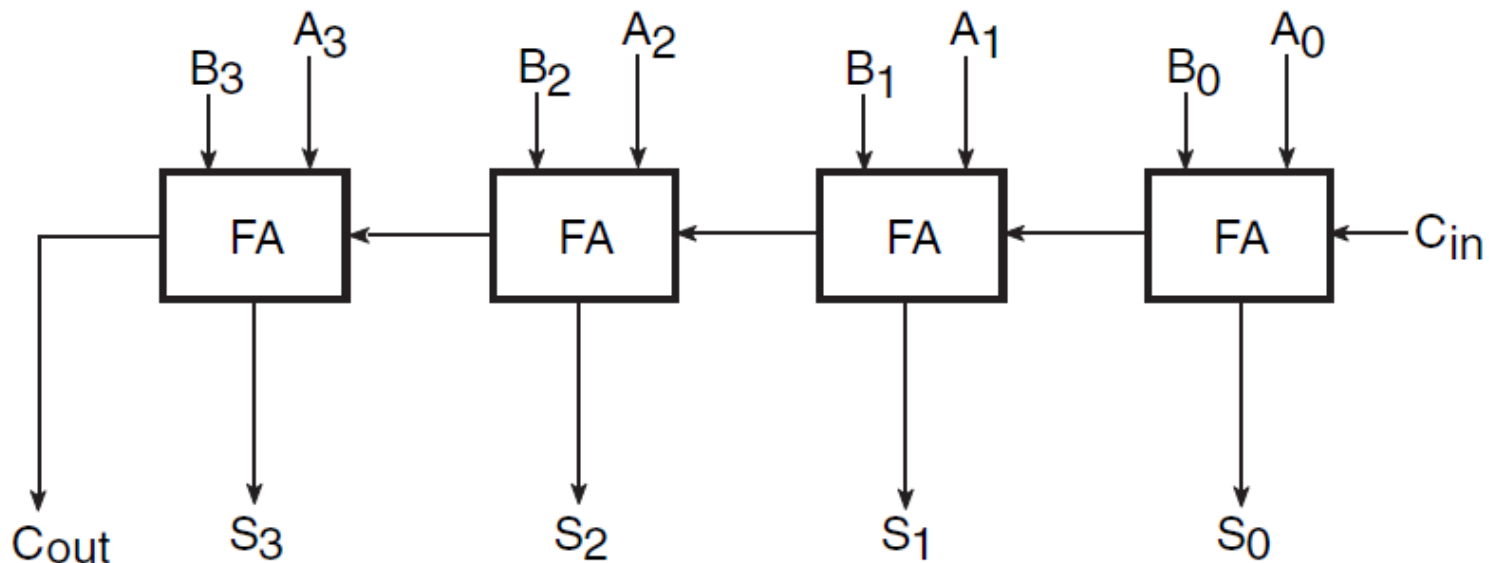
a. Bộ cộng \rightarrow Bộ cộng đầy đủ (Full-Adder)

- Xây dựng bộ full-adder từ các bộ half-adder



a. Bộ cộng \rightarrow Bộ cộng nhiều bit

- Chức năng: Đây là bộ cộng 2 số nhị phân 4 bit, kết quả nhận được là 1 số nhị phân 5 bit.
- Sơ đồ:



Các mạch số học cơ bản

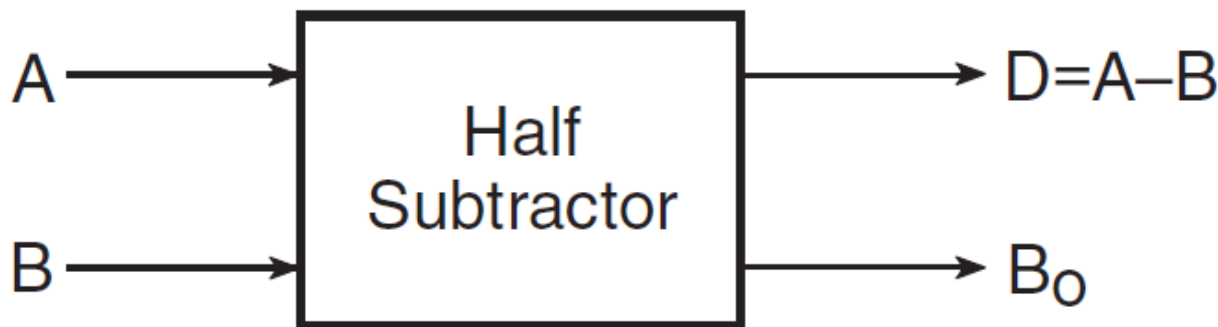
- a. Bộ cộng
- b. Bộ trừ
- c. Bộ so sánh

b. Bộ trừ

- Chức năng: thực hiện phép trừ giữa 2 số nhị phân.
- Bao gồm:
 - Bộ bán hiệu
 - Bộ trừ đầy đủ
 - Bộ trừ cộng

b. Bộ trừ \rightarrow Bộ bán hiệu (Half-Subtractor)

- Dùng để thực hiện phép trừ giữa 2 bit thấp nhất trong phép trừ giữa 2 số nhị phân
- Sơ đồ khối:
 - D: hiệu
 - B_0 : bit mượn



b. Bộ trừ → Bộ bán hiệu (Half-Subtractor)

- Bảng thật:

A	B	D	B ₀
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

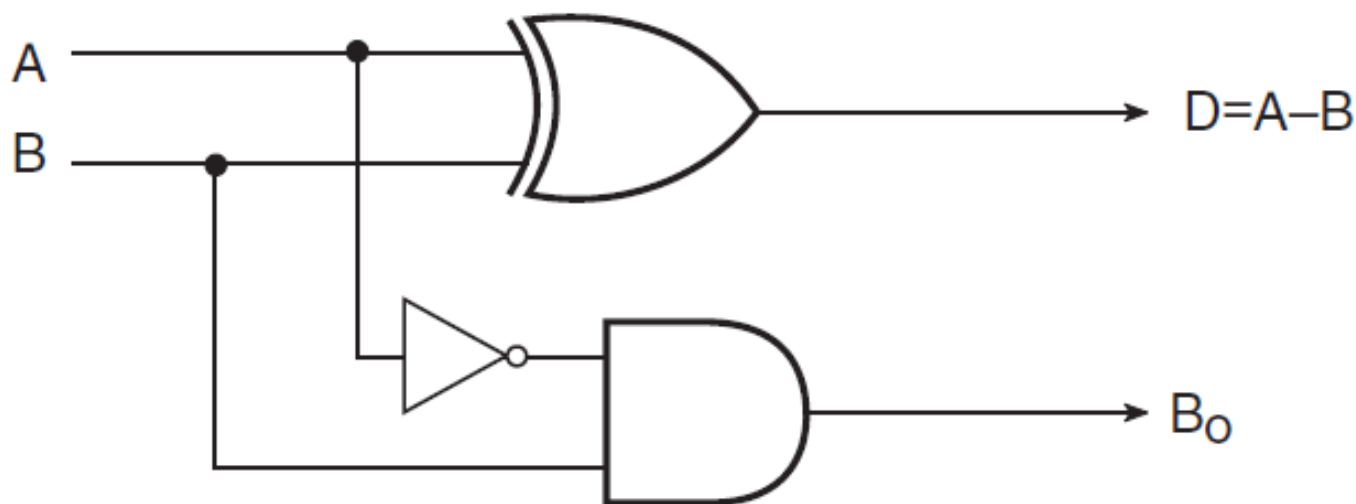
- Biểu thức đầu ra phụ thuộc đầu vào:

$$D = \overline{A}.B + A.\overline{B}$$

$$B_0 = \overline{A}.B$$

b. Bộ trừ → Bộ bán hiệu (Half-Subtractor)

- Sơ đồ mạch: sử dụng một cổng XOR cho D và một cổng AND với đầu vào A đảo cho B_0 .

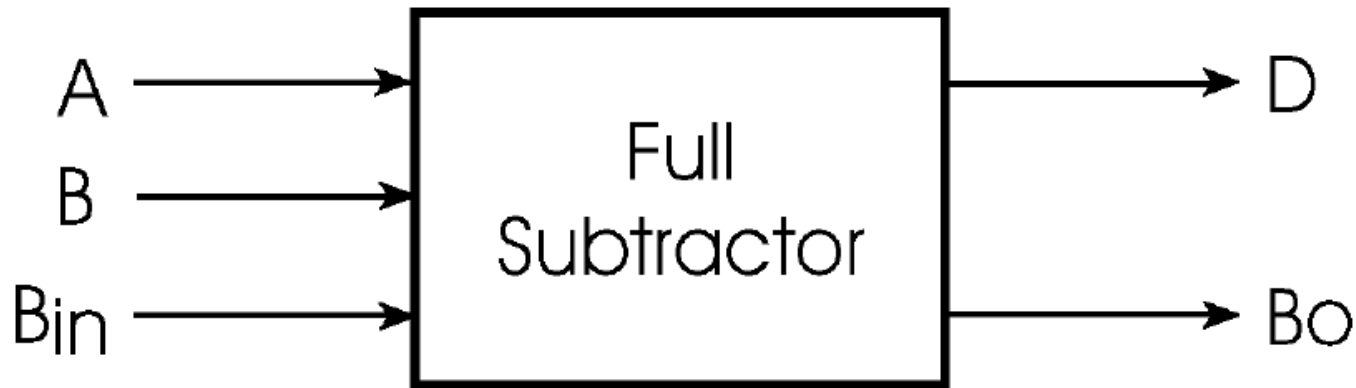


$$D = \bar{A}.B + A.\bar{B}$$

$$B_0 = \bar{A}.B$$

b. Bộ trừ → Bộ trừ đầy đủ (Full-Subtractor)

- Dùng để thực hiện phép trừ giữa 2 bit bất kỳ trong phép trừ 2 số nhị phân.
- Sơ đồ khối:



b. Bộ trừ → Bộ trừ đầy đủ (Full-Subtractor)

- Bảng thật:

Minuend (A)	Subtrahend (B)	Borrow In (B _{in})	Difference (D)	Borrow Out (B _o)
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

- Biểu thức đầu ra phụ thuộc đầu vào:

$$D = \overline{A}.\overline{B}.B_{in} + \overline{A}.B.\overline{B}_{in} + A.\overline{B}.\overline{B}_{in} + A.B.B_{in}$$

$$B_o = \overline{A}.\overline{B}.B_{in} + \overline{A}.B.\overline{B}_{in} + \overline{A}.B.B_{in} + A.B.B_{in}$$

b. Bộ trừ → Bộ trừ đầy đủ (Full-Subtractor)

- Tối thiểu hóa hàm Boolean:
 - Lập bìa Các-nô cho D và B_{out} :

AB \ B_{in}		
	$\overline{B_{in}}$	B_{in}
$\overline{A}\overline{B}$		1
$\overline{A}B$	1	
AB		1
$A\overline{B}$	1	

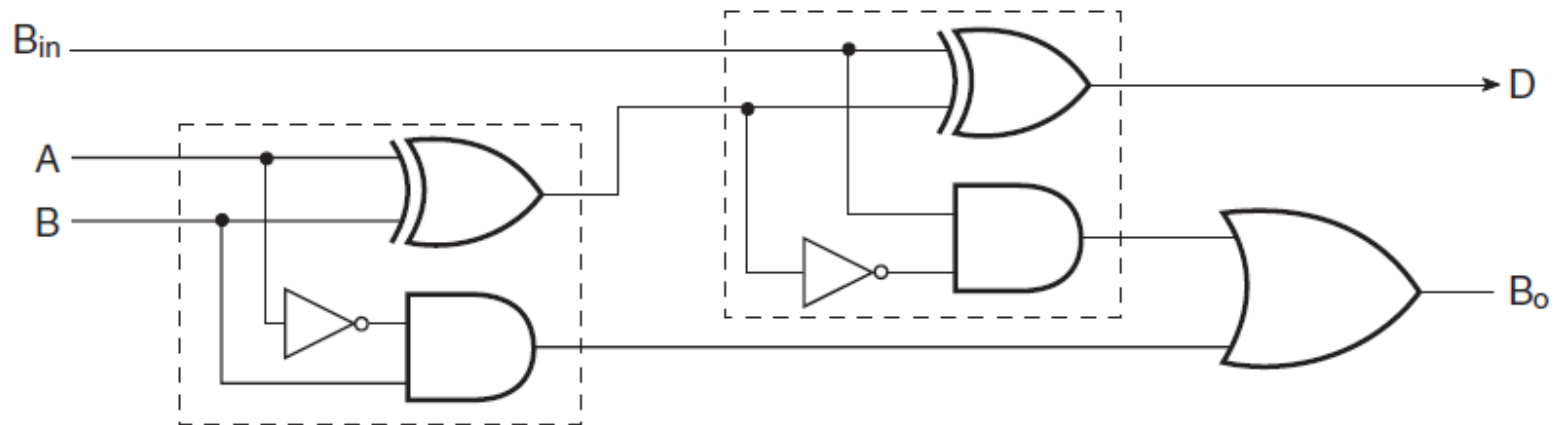
AB \ B_{in}		
	$\overline{B_{in}}$	B_{in}
$\overline{A}\overline{B}$		1
$\overline{A}B$	1	1
AB		1
$A\overline{B}$		

$$B_o = \overline{A}.B + \overline{A}.B_{in} + B.B_{in}$$

$$D = \overline{A}.\overline{B}.B_{in} + \overline{A}.B.\overline{B_{in}} + A.\overline{B}.\overline{B_{in}} + A.B.B_{in}$$

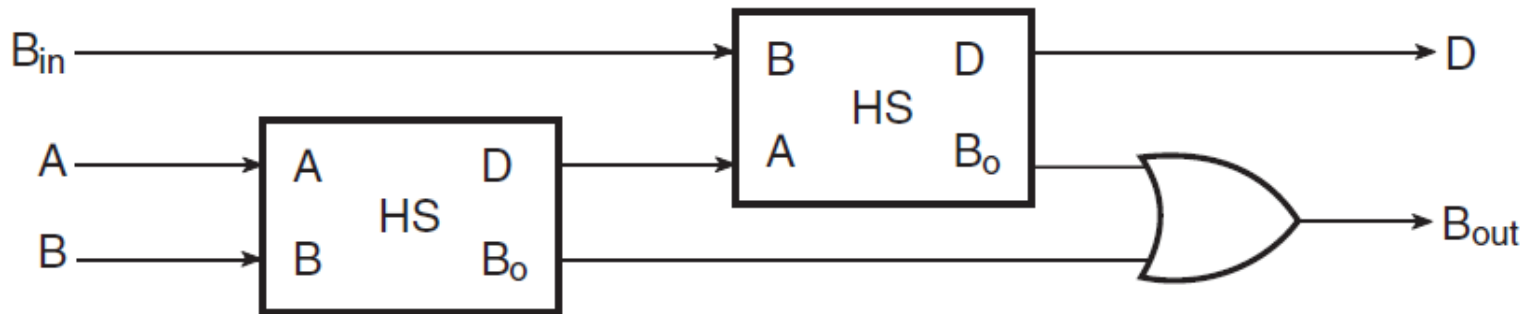
b. Bộ trừ \rightarrow Bộ trừ đầy đủ (Full-Subtractor)

- Sơ đồ mạch:



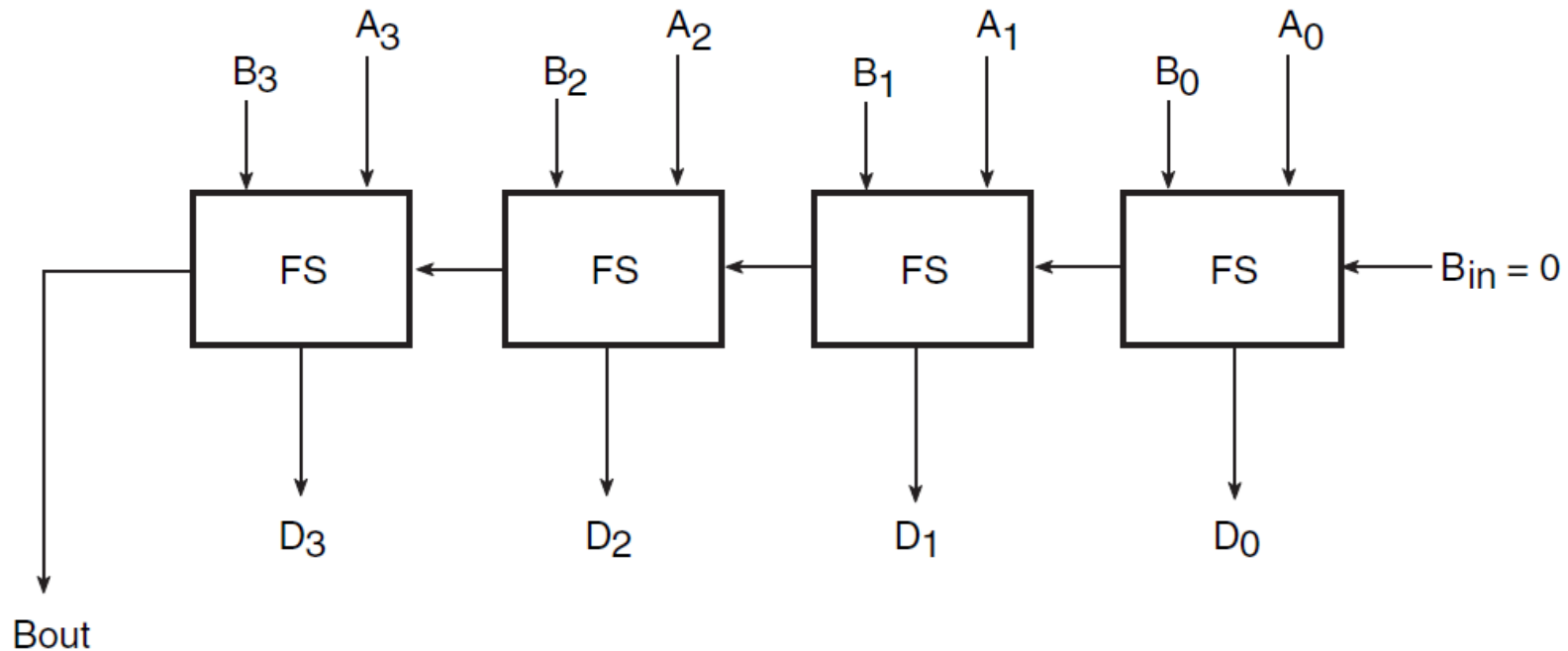
b. Bộ trừ \rightarrow Bộ trừ đầy đủ (Full-Subtractor)

- Xây dựng bộ full-subtractor từ các bộ half-subtractor



b. Bộ trừ \rightarrow Bộ trừ nhiều bit

- Chức năng: Đây là bộ trừ 2 số nhị phân 4 bit, kết quả nhận được là 1 số nhị phân 5 bit.
- Sơ đồ:



b. Bộ trừ \rightarrow Bộ trừ cộng

- Bộ trừ cộng 4 bit:

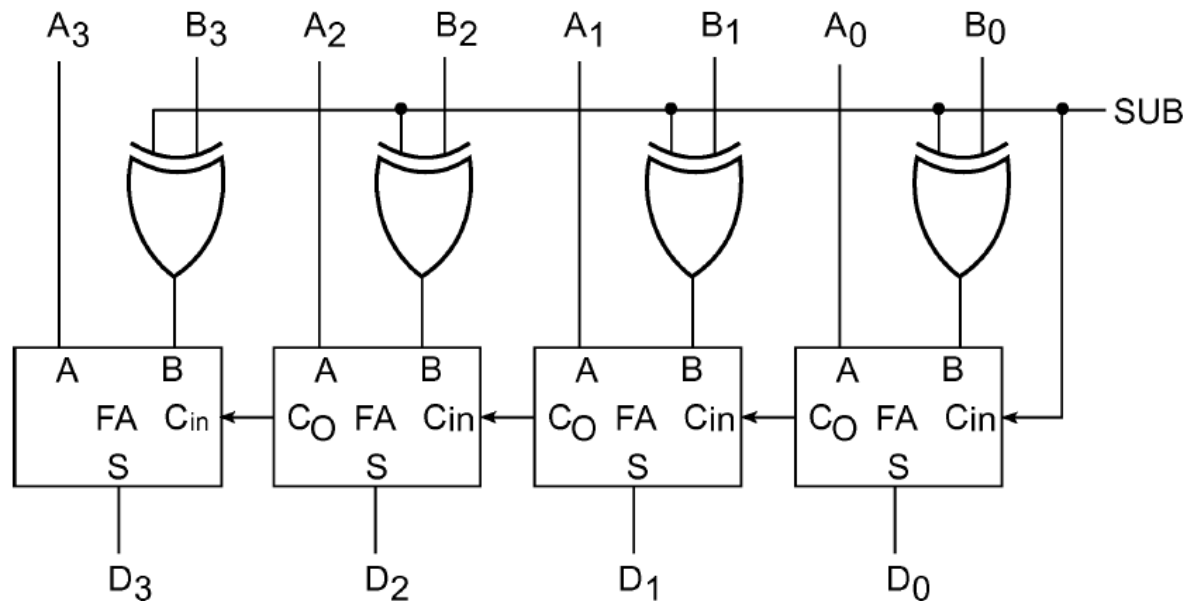
- $SUB = 0$

- $A+B$

- $SUB=1$

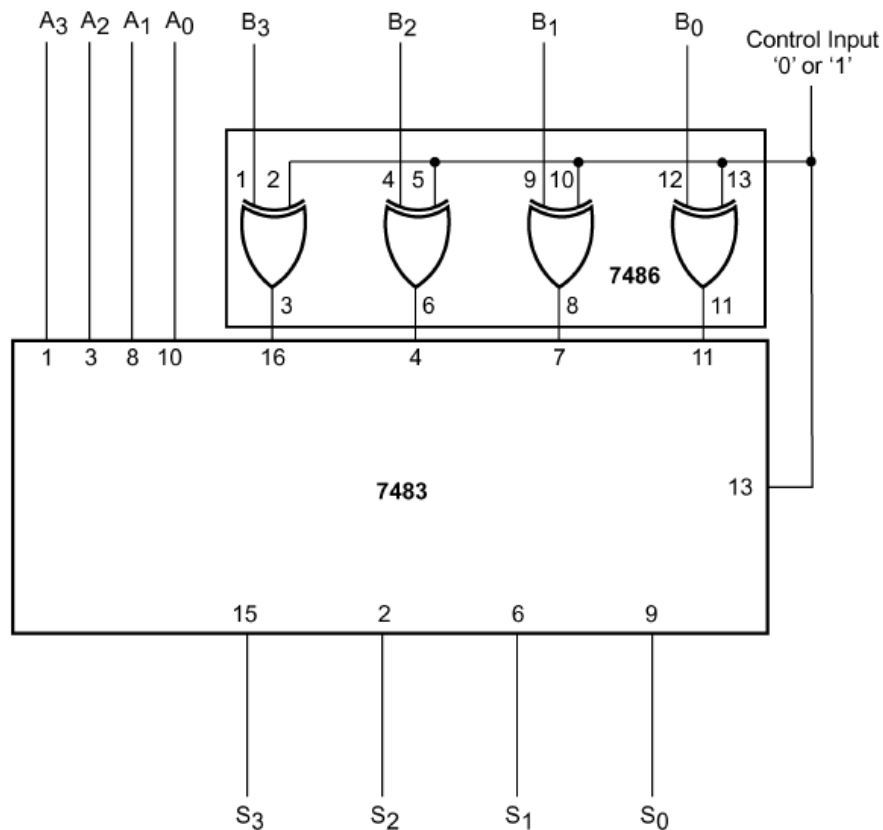
- $C_{in}=1$

- $A-B$



b. Bộ trừ \rightarrow Bộ trừ cộng

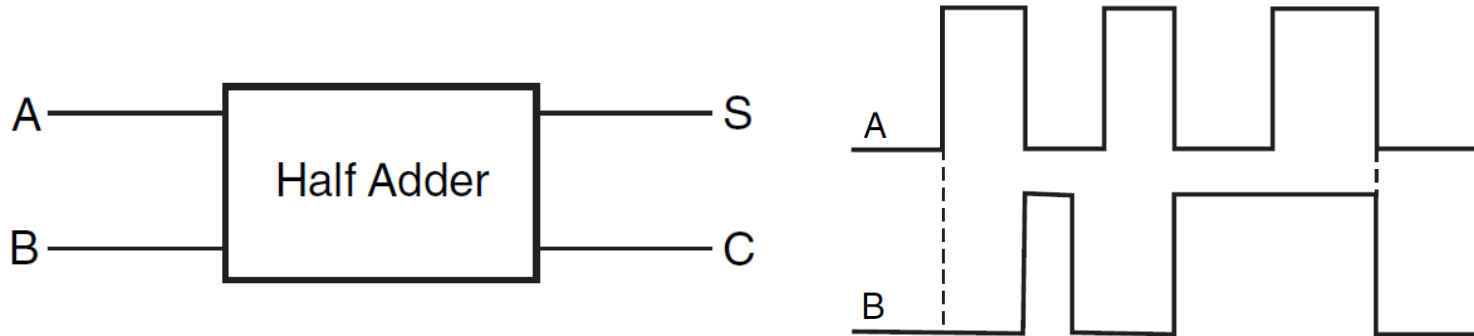
- Bộ trừ cộng 4 bit sử dụng:



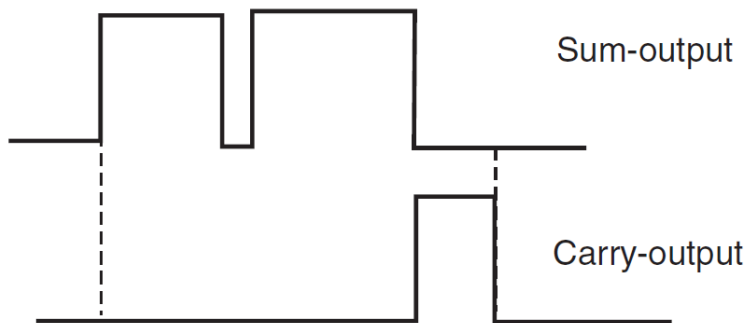
- Bộ cộng 4 bit đầy đủ 7483
 - Bộ 4 cổng XOR
- 2 đầu vào
7486:

Ví dụ 1

- Cho mạch bán cộng như hình sau với đầu vào A, B

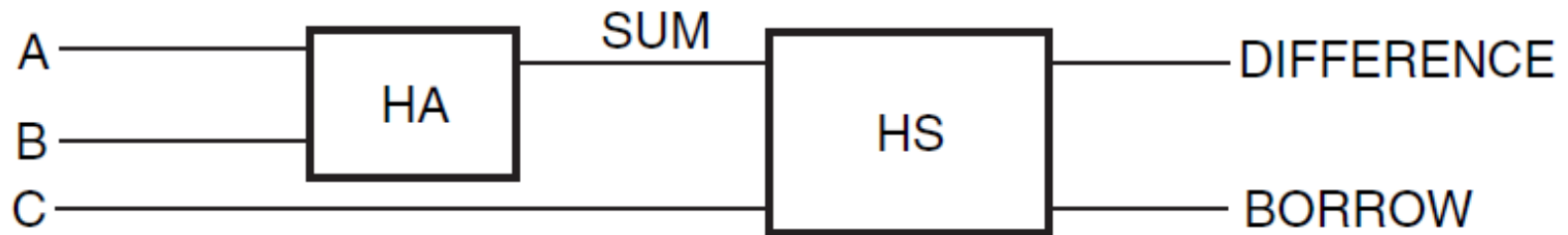


- Vẽ tín hiệu đầu ra của S và C:



Ví dụ 2

- Cho mạch sau:



- Viết hàm Boolean tối thiểu hóa cho đầu ra D và B_o

Ví dụ 2

- Viết hàm Boolean tối thiểu hóa cho đầu ra D và B_o

$$\text{DIFFERENCE output} = X \oplus Y = \overline{X}.Y + X.\overline{Y}$$

$$\text{BORROW output} = \overline{X}.Y$$

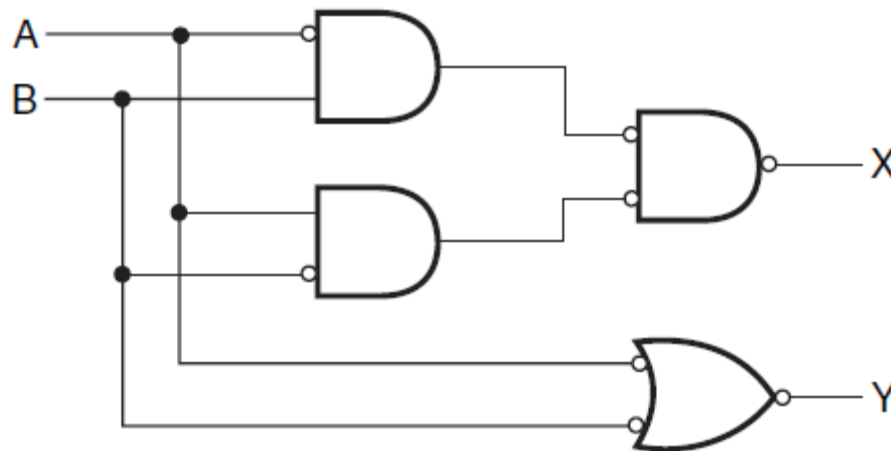
$$X = \overline{A}.B + A.\overline{B} \text{ and } Y = C$$

$$\begin{aligned}\text{DIFFERENCE output} &= \overline{(\overline{A}.B + A.\overline{B})}.C + (\overline{A}.B + A.\overline{B}).\overline{C} \\ &= (A.B + \overline{A}.\overline{B}).C + (\overline{A}.B + A.\overline{B}).\overline{C} \\ &= A.B.C + \overline{A}.\overline{B}.C + \overline{A}.B.\overline{C} + A.\overline{B}.\overline{C}\end{aligned}$$

$$\begin{aligned}\text{BORROW output} &= \overline{X}.Y = \overline{(\overline{A}.B + A.\overline{B})}.C \\ &= (A.B + \overline{A}.\overline{B}).C = A.B.C + \overline{A}.\overline{B}.C\end{aligned}$$

Ví dụ 3

- Cho mạch sau:



- Đây là mạch gì?

$$X = (\overline{\overline{A.B}}) . (\overline{\overline{A.B}}) = (\overline{\overline{A.B} + \overline{\overline{A.B}}}) = \overline{A.B} + A.\overline{B}$$

$$Y = (\overline{\overline{A} + \overline{B}}) = A.B$$

Các mạch số học cơ bản

- a. Bộ cộng
- b. Bộ trừ
- c. Bộ so sánh

c. Bộ so sánh

- Dùng để so sánh 2 số nhị phân
- Có 2 kiểu so sánh:
 - So sánh đơn giản: Kết quả so sánh: bằng nhau, khác nhau
 - So sánh đầy đủ: Kết quả so sánh: **lớn hơn, nhỏ hơn, bằng nhau**
- Có 2 loại bộ so sánh:
 - Bộ so sánh đơn giản
 - Bộ so sánh đầy đủ

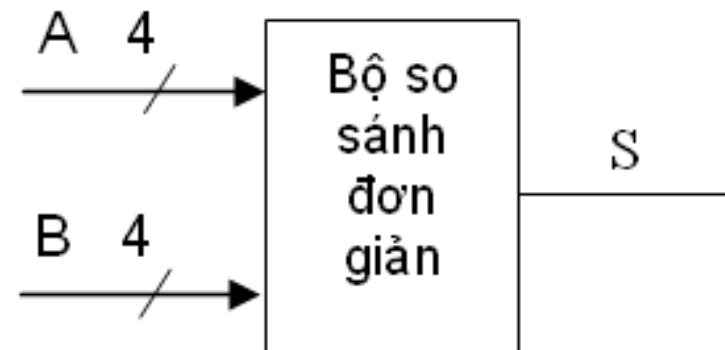
c. Bộ so sánh \rightarrow Bộ so sánh đơn giản

- Giả sử cần xây dựng bộ so sánh đơn giản 2 số A và B:

$$\begin{array}{rcccc} A & a_3 & a_2 & a_1 & a_0 \\ B & b_3 & b_2 & b_1 & b_0 \end{array}$$

- Đầu ra S

- $S = 1 \Leftrightarrow A = B$
- $S = 0 \Leftrightarrow A \neq B$



c. Bộ so sánh \rightarrow Bộ so sánh đơn giản (tiếp)

- Ta có:

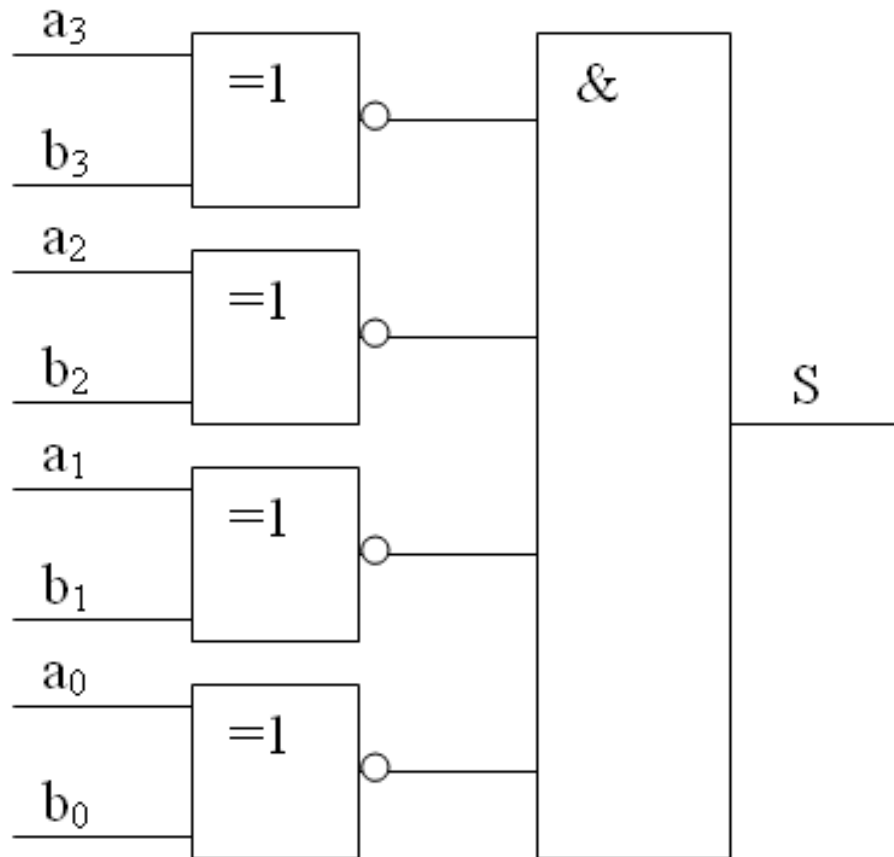
$$A = B \Leftrightarrow \begin{cases} a_3 = b_3 \\ a_2 = b_2 \\ a_1 = b_1 \\ a_0 = b_0 \end{cases} \Leftrightarrow \begin{cases} a_3 \oplus b_3 = 0 \\ a_2 \oplus b_2 = 0 \\ a_1 \oplus b_1 = 0 \\ a_0 \oplus b_0 = 0 \end{cases} \Leftrightarrow \begin{cases} \overline{a_3 \oplus b_3} = 1 \\ \overline{a_2 \oplus b_2} = 1 \\ \overline{a_1 \oplus b_1} = 1 \\ \overline{a_0 \oplus b_0} = 1 \end{cases}$$

- Biểu thức Boolean:

$$S = \overline{(a_3 \oplus b_3)} \overline{(a_2 \oplus b_2)} \overline{(a_1 \oplus b_1)} \overline{(a_0 \oplus b_0)}$$

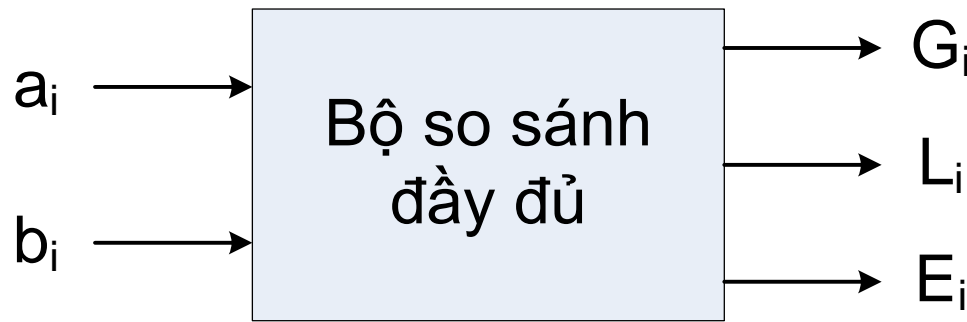
c. Bộ so sánh \rightarrow Bộ so sánh đơn giản (tiếp)

- Sơ đồ mạch:



c. Bộ so sánh \rightarrow Bộ so sánh đầy đủ

- Bộ so sánh 2 bit đầy đủ:
 - Đầu vào: 2 bit cần so sánh a_i và b_i
 - Đầu ra: 3 tín hiệu để báo kết quả lớn hơn, nhỏ hơn, bằng nhau của 2 bit
 - $a_i > b_i \Leftrightarrow G_i = 1$ còn $E_i, L_i = 0$
 - $a_i < b_i \Leftrightarrow L_i = 1$ còn $E_i, G_i = 0$
 - $a_i = b_i \Leftrightarrow E_i = 1$ còn $G_i, L_i = 0$
- Sơ đồ khối:



c. Bộ so sánh → Bộ so sánh 2 bit đầy đủ

- Bảng thật:

a_i	b_i	G_i	L_i	E_i
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

- Biểu diễn đầu ra theo đầu vào:

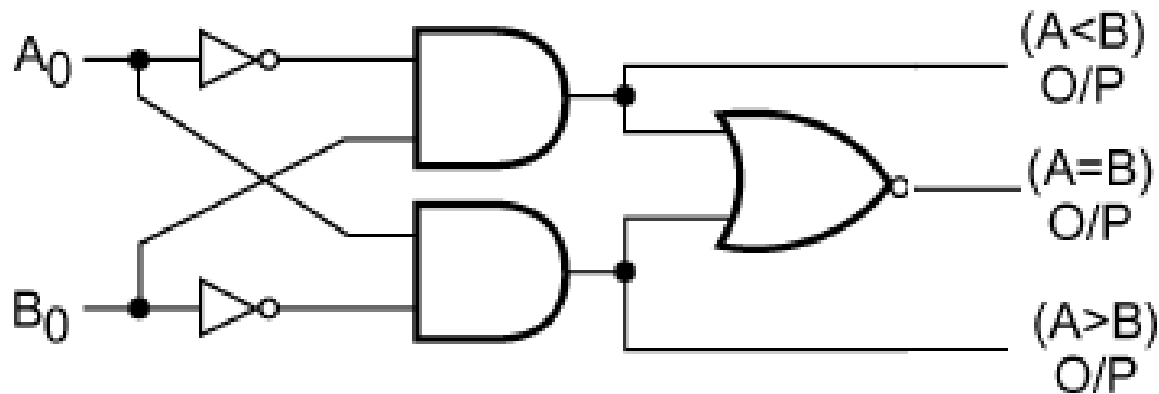
$$G_i = a_i \cdot \overline{b_i}$$

$$L_i = \overline{a_i} \cdot b_i$$

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

c. Bộ so sánh \rightarrow Bộ so sánh 2 bit đầy đủ

- Sơ đồ mạch:



c. Bộ so sánh \rightarrow Bộ so sánh đầy đủ 2 số nhị phân 2 bit

- Đầu vào: $A (A_1 A_0)$
 $B (B_1 B_0)$

- Đầu ra: $X, Y, Z \rightarrow A = B, A > B, A < B$

$X = 1, Y = 0, Z = 0 \quad A = B$

$X = 0, Y = 1, Z = 0 \quad A > B$

$X = 0, Y = 0, Z = 1 \quad A < B$

c. Bộ so sánh \rightarrow Bộ so sánh đầy đủ 2 số nhị phân 2 bit

- Biểu thức Boolean:

$$X = x_1 . x_0$$

$$x_0 = A_0 . B_0 + \overline{A_0} . \overline{B_0}$$

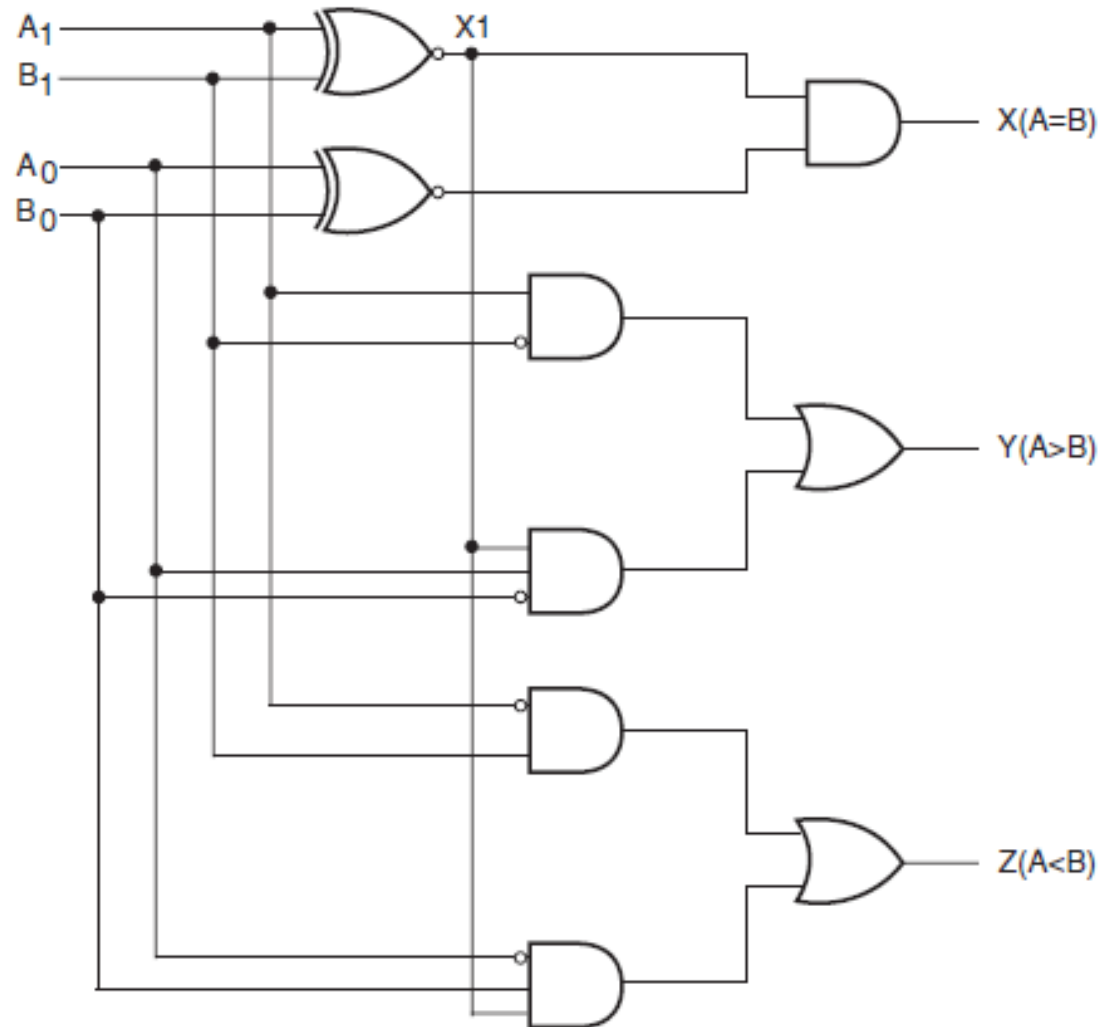
$$x_1 = A_1 . B_1 + \overline{A_1} . \overline{B_1}$$

$$Y = A_1 . \overline{B_1} + x_1 . A_0 . \overline{B_0}$$

$$Z = \overline{A_1} . B_1 + x_1 . \overline{A_0} . B_0$$

c. Bộ so sánh \rightarrow Bộ so sánh đầy đủ 2 số nhị phân 2 bit

- Sơ đồ mạch:



c. Bộ so sánh \rightarrow Bộ so sánh đầy đủ 2 số nhị phân

- Cấu tạo: gồm các bộ so sánh 2 bit
- Có tín hiệu CS (Chip Select)
 - CS = 0, tất cả các đầu ra = 0 (không so sánh)
 - CS = 1, hoạt động bình thường
- Biểu diễn các đầu ra của bộ so sánh 2 bit theo đầu vào:

$$G_i = CS \cdot a_i \cdot \overline{b_i}$$

$$L_i = CS \cdot \overline{a_i} \cdot b_i$$

$$E_i = CS \cdot \overline{(a_i \oplus b_i)}$$

Một số hệ tổ hợp cơ bản

- Các mạch số học cơ bản
- Bộ chọn kênh
- Bộ mã hóa
- Bộ phân kênh
- Bộ giải mã

Bộ chọn kênh

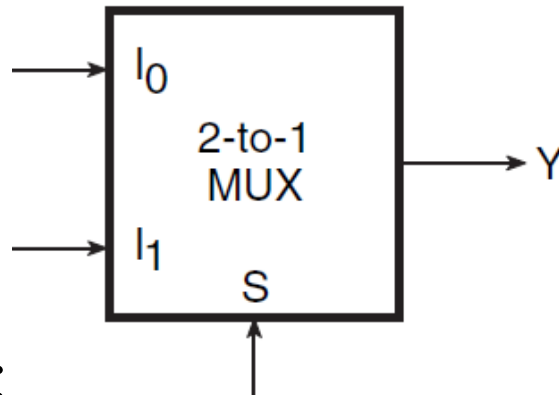
- MultiPlexor – MUX/data selector
- Có nhiều đầu vào tín hiệu và 1 đầu ra
- Chức năng: chọn 1 tín hiệu trong nhiều tín hiệu đầu vào để đưa ra đầu ra
- Bao gồm:
 - MUX 2-1
 - MUX 4-1
 - MUX 8-1
 - MUX 16-1
 - MUX $2^n - 1$ với n đầu vào điều khiển

Bộ chọn kênh

- Ứng dụng MUX:
 - Sử dụng MUX $2^n - 1$ thiết kế các hàm logic n biến
 - Sử dụng MUX $2^n - 1$ thiết kế các hàm logic $n+1$ biến
 - Bộ chuyển đổi song song – nối tiếp
 - MUX nhiều tầng

Bộ chọn kênh 2-1

- Sơ đồ khối:



- Tín hiệu chọn:

S	Y
0	I_0
1	I_1

- Tín hiệu ra:

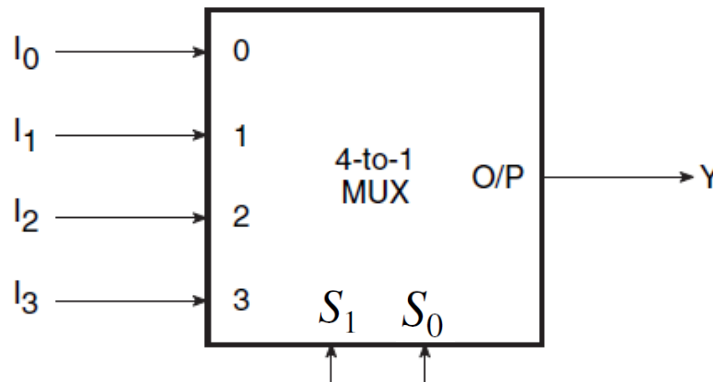
$$S = 0 \quad Y = I_0$$

$$S = 1 \quad Y = I_1$$

$$Y = \bar{S}I_0 + SI_1$$

Bộ chọn kênh 4-1

- Sơ đồ khối:



- Tín hiệu chọn

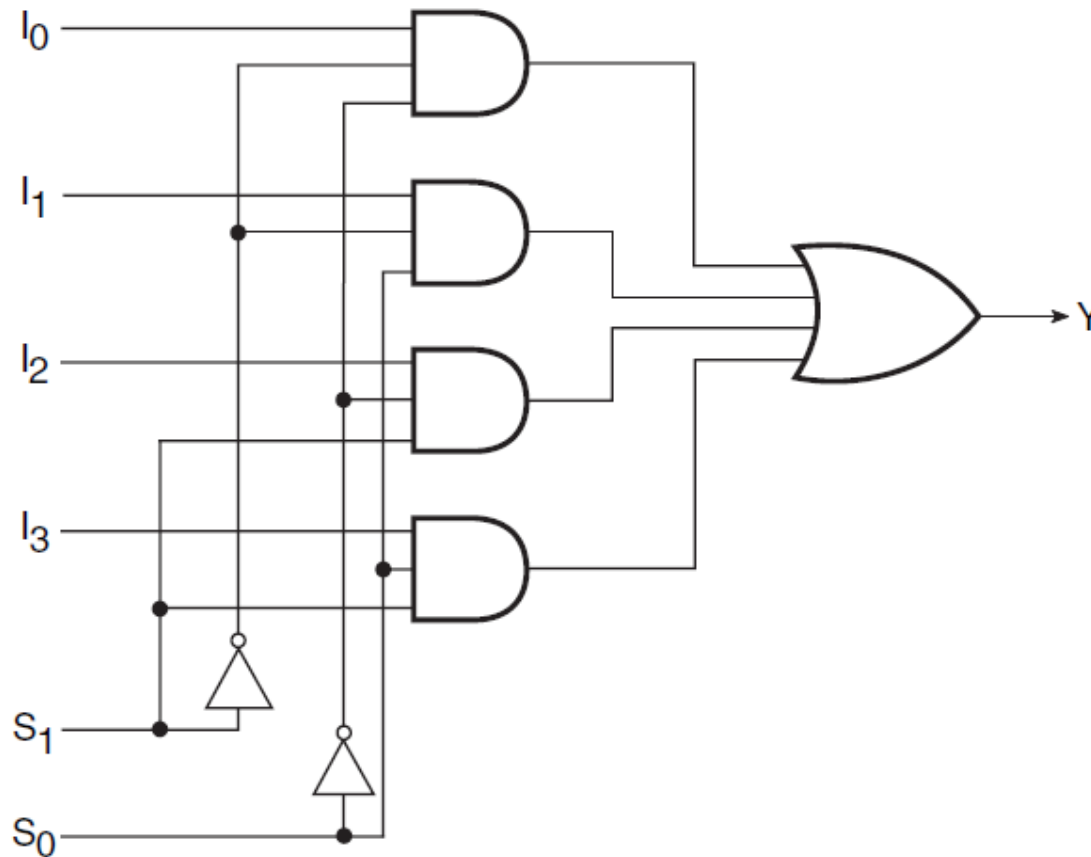
S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

- Tín hiệu ra:

$$Y = I_0 \cdot \overline{S_1} \cdot \overline{S_0} + I_1 \cdot \overline{S_1} \cdot S_0 + I_2 \cdot S_1 \cdot \overline{S_0} + I_3 \cdot S_1 \cdot S_0$$

Bộ chọn kênh 4-1

- Sơ đồ mạch:



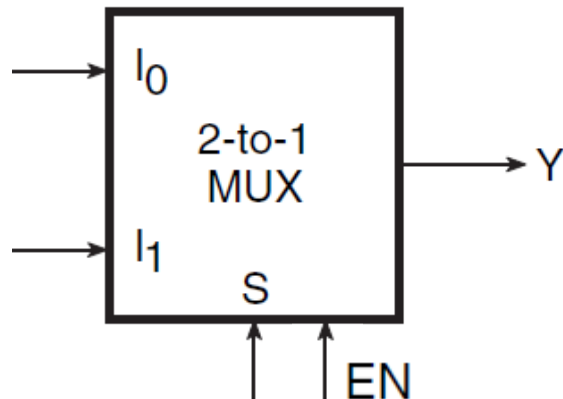
Bộ chọn kênh 8-1

- Tín hiệu ra:

$$\begin{aligned} Y = & I_0.\overline{S_2}.\overline{S_1}.\overline{S_0} + I_1.\overline{S_2}.\overline{S_1}.S_0 + I_2.\overline{S_2}.S_1.\overline{S_0} \\ & + I_3.\overline{S_2}.S_1.S_0 + I_4.S_2.\overline{S_1}.\overline{S_0} + I_5.S_2.\overline{S_1}.S_0 \\ & + I_6.S_2.S_1.\overline{S_0} + I_7.S_2.S_1.S_0 \end{aligned}$$

Bộ chọn kênh 2-1 với đầu vào điều khiển

- Sơ đồ khối:



- Tín hiệu chọn:

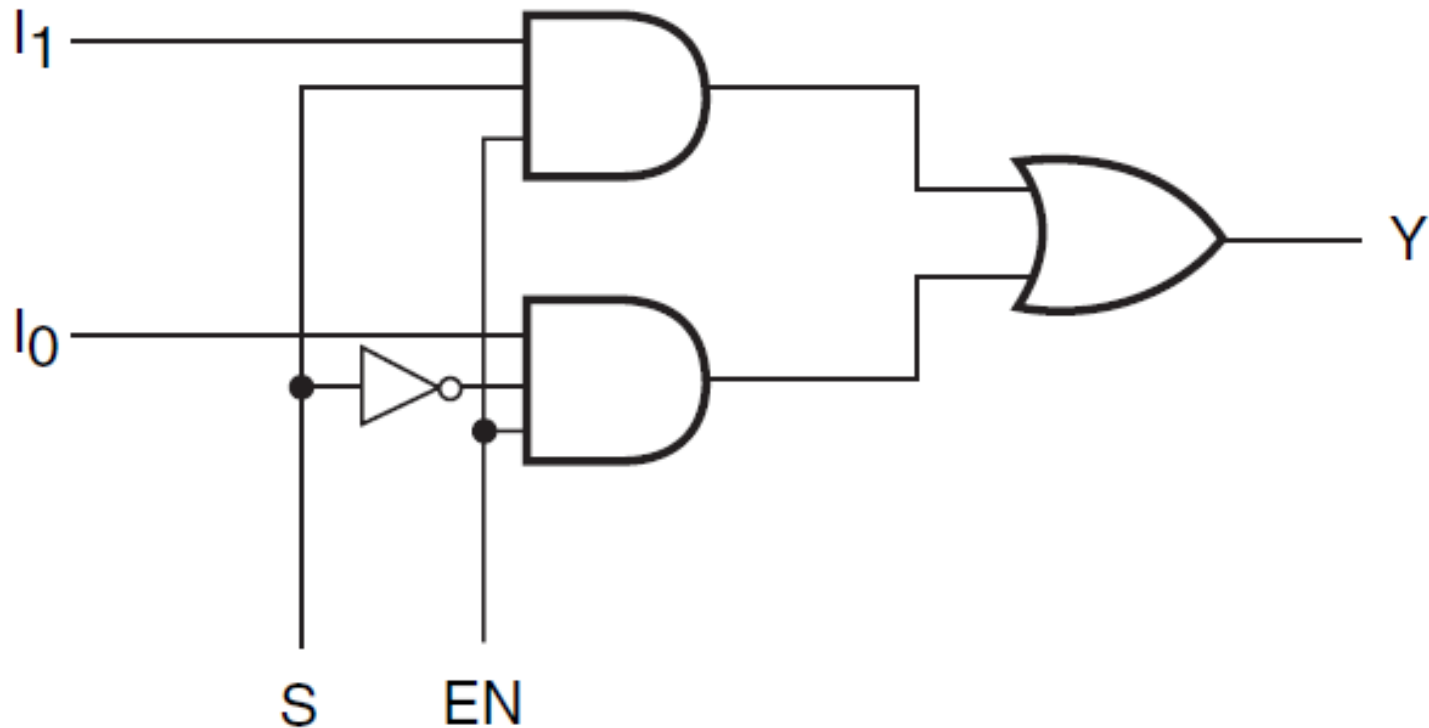
S	EN	Y
X	0	0
0	1	I_0
1	1	I_1

- Tín hiệu ra:

$$\begin{aligned} \text{EN} = 1 \quad S = 0 \quad Y = I_0 \quad Y &= EN(\overline{S}I_0 + SI_1) \\ S = 1 \quad Y &= I_1 \end{aligned}$$

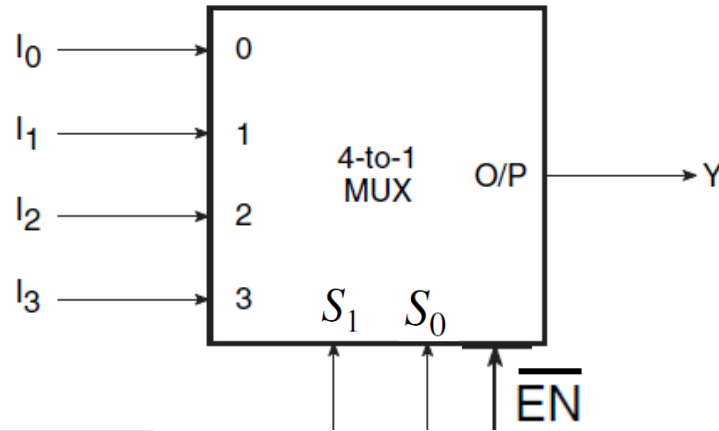
Bộ chọn kênh 2-1 với đầu vào điều khiển

- Sơ đồ mạch:



Bộ chọn kênh 4-1 với đầu vào điều khiển

- Sơ đồ khối:



- Tín hiệu chọn:

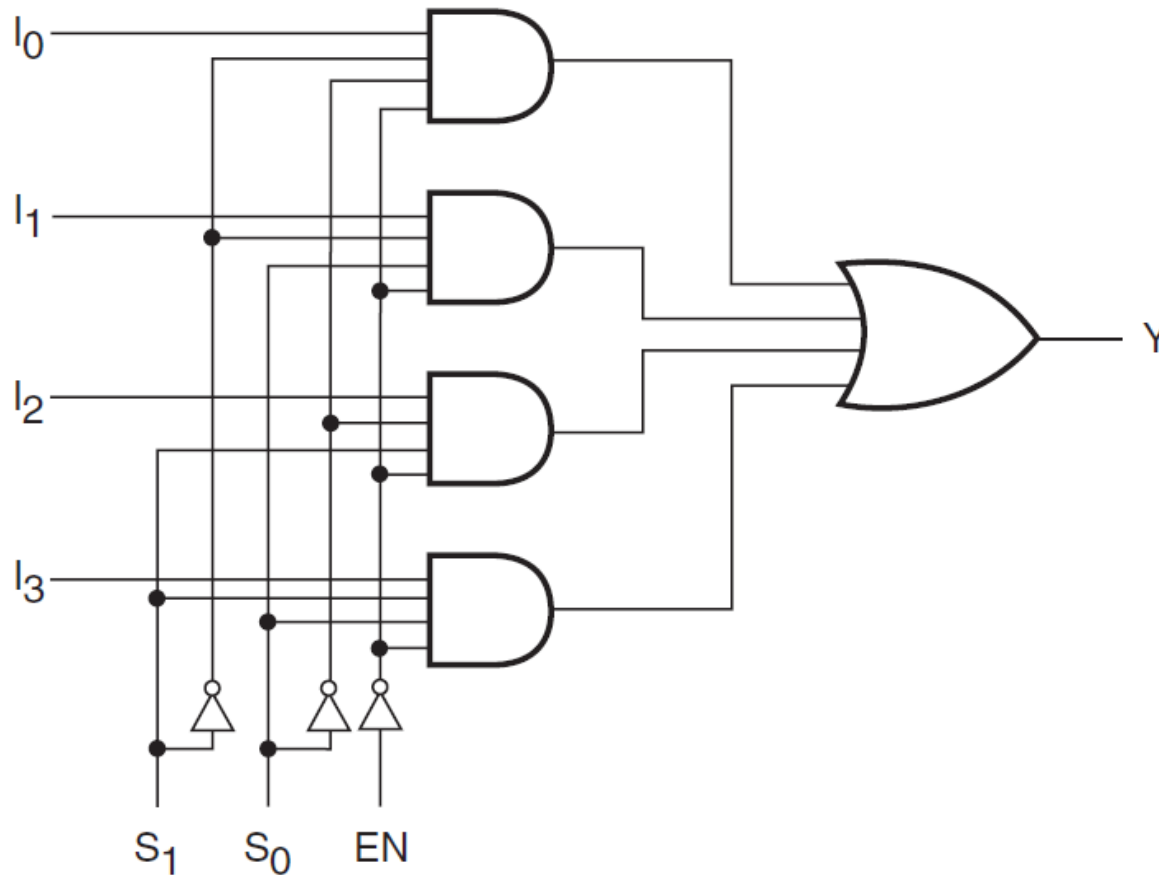
S_1	S_0	EN	Y
X	X	1	0
0	0	0	I_0
0	1	0	I_1
1	0	0	I_2
1	1	0	I_3

- Tín hiệu ra:

$$Y = (I_0 \cdot \overline{S_1} \cdot \overline{S_0} + I_1 \cdot \overline{S_1} \cdot S_0 + I_2 \cdot S_1 \cdot \overline{S_0} + I_3 \cdot S_1 \cdot S_0) \overline{EN}$$

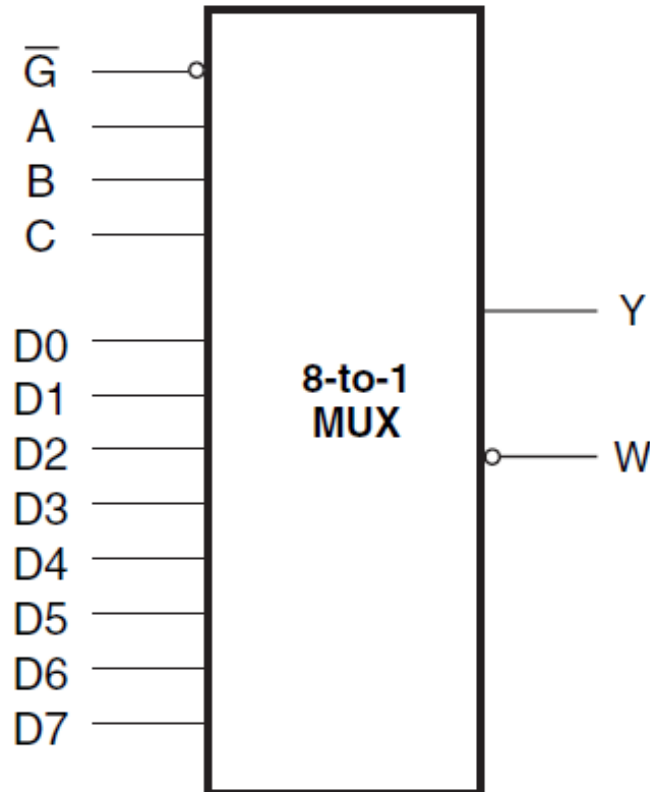
Bộ chọn kênh 4-1 với đầu vào điều khiển

- Sơ đồ mạch:



Bộ chọn kênh 8-1 với đầu vào điều khiển

- Sơ đồ khối:
- Bảng thật

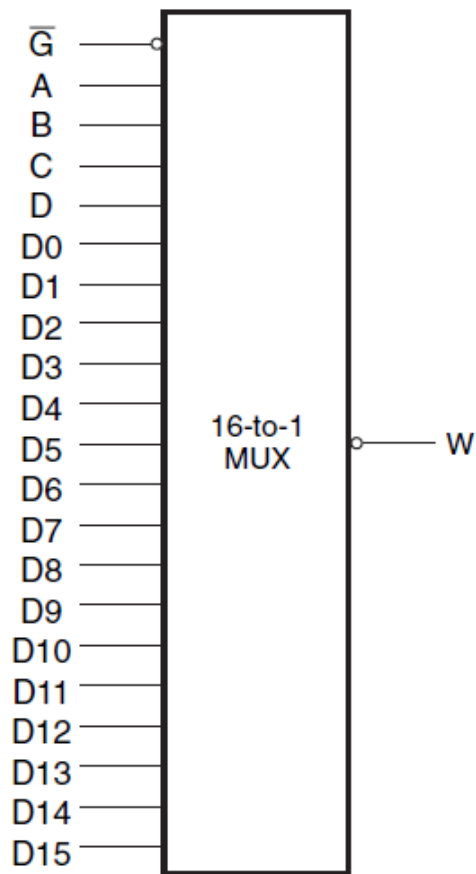


Inputs				Output	
Select			Enable \overline{G}	Y W	
C	B	A			
X	X	X	H	L	\overline{H}
L	L	L	L	D0	$\overline{D0}$
L	L	H	L	D1	$\overline{D1}$
L	H	L	L	D2	$\overline{D2}$
L	H	H	L	D3	$\overline{D3}$
H	L	L	L	D4	$\overline{D4}$
H	L	H	L	D5	$\overline{D5}$
H	H	L	L	D6	$\overline{D6}$
H	H	H	L	D7	$\overline{D7}$

\overline{G} : ENABLE input
A, B, C : Select inputs
D0-D7 : Data inputs
Y,W : outputs

Bộ chọn kênh 16-1 với đầu vào điều khiển

- Sơ đồ khối:



- Bảng thật

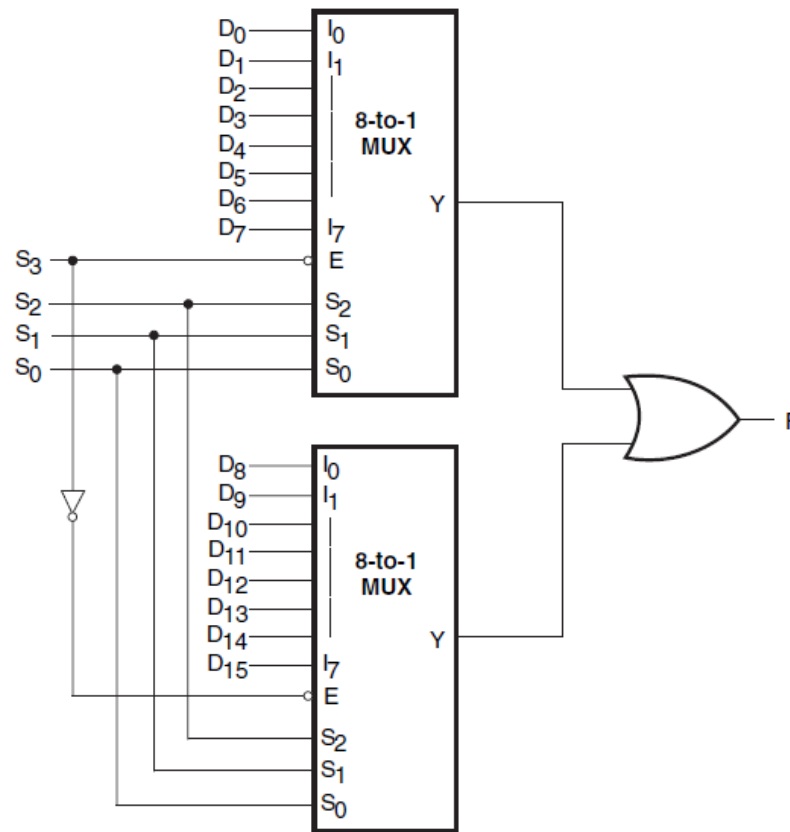
Inputs					Output W
Select				Enable \overline{G}	
D	C	B	A		
X	X	X	X	H	H
L	L	L	L	L	$\overline{D0}$
L	L	L	H	L	$\overline{D1}$
L	L	H	L	L	$\overline{D2}$
L	L	H	H	L	$\overline{D3}$
L	H	L	L	L	$\overline{D4}$
L	H	L	H	L	$\overline{D5}$
L	H	H	L	L	$\overline{D6}$
L	H	H	H	L	$\overline{D7}$
H	L	L	L	L	$\overline{D8}$
H	L	L	H	L	$\overline{D9}$
H	L	H	L	L	$\overline{D10}$
H	L	H	H	L	$\overline{D11}$
H	H	L	L	L	$\overline{D12}$
H	H	L	H	L	$\overline{D13}$
H	H	H	L	L	$\overline{D14}$
H	H	H	H	L	$\overline{D15}$

Bộ MUX nhiều tầng

- Nếu số lượng đầu vào của 1 IC MUX không đủ nhiều, có thể ghép một số bộ MUX để tạo ra một bộ MUX có số lượng đầu vào lớn hơn.
- Nếu số lượng đầu vào cho phép của 1 IC MUX là 2^n , trong khi số lượng đầu vào mong muốn là 2^N ($N > n$), thì số lượng MUX cần sử dụng là 2^{N-n}
- Kết nối các bit nhỏ nhất của các đầu vào lựa chọn với nhau.
- Các bit còn lại của đầu vào lựa chọn được sử dụng để quyết định bật/tắt các IC.
- Đầu ra của các bộ MUX được OR với nhau để cho ra kết quả cuối cùng.

Ví dụ 4

- Sử dụng 2 bộ MUX 8-to-1 để thiết kế bộ MUX 16-to-1 với đầu vào tích cực ở mức thấp.



Thực hiện các hàm Boolean bằng bộ chọn kênh

- Một trong những ứng dụng của bộ chọn kênh là sử dụng để thiết kế các hàm Boolean.
- Kỹ thuật đơn giản nhất là sử dụng bộ chọn kênh 2^n -to-1 để thiết kế các hàm Boolean n biến.
 - Đầu vào tương ứng với các tích số của đầy đủ các biến ở dạng bù hay không bù (minterm) trong hàm Boolean tương ứng với trạng thái logic '1'.
 - Các minterm còn lại vắng mặt trong hàm Boolean bị vô hiệu hóa bằng cách cho các đầu vào tương ứng ở trạng thái logic '0'.

Ví dụ 5

- Thực hiện hàm sau:

$$f(A, B, C) = \sum 2, 4, 7$$

- Triển khai hàm f :

$$f(A, B, C) = \overline{A}.B.\overline{C} + A.\overline{B}.\overline{C} + A.B.C$$

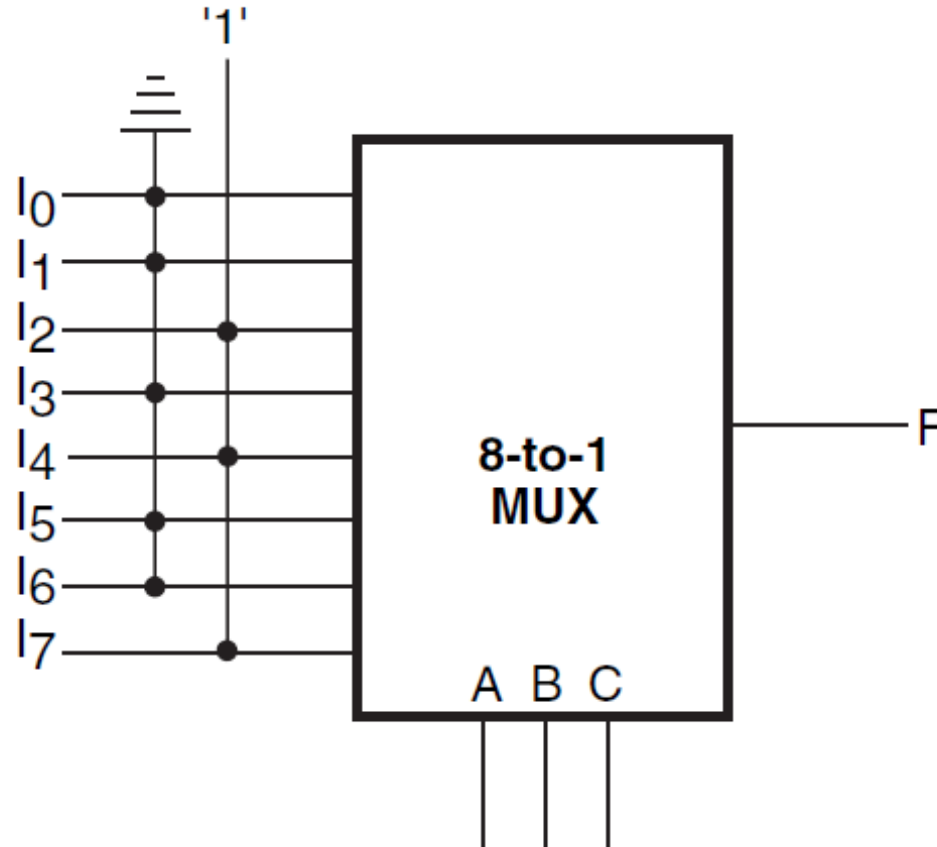
Ví dụ 5

- Bảng thật:

Minterm	A	B	C	$f(A,B,C)$
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

Ví dụ 5

- Sơ đồ mạch:



Thực hiện các hàm Boolean bằng bộ chọn kênh

- Ngoài ra, có thể sử dụng bộ chọn kênh 2^n -to-1 để thiết kế các hàm Boolean $n+1$ biến.
 - Trong $n+1$ biến, n biến được kết nối với các đầu vào chọn lọc của MUX, biến còn lại được nối với đầu vào.
 - Các đầu vào của bộ MUX sẽ được thiết lập 1 trong 4 trạng thái: trạng thái logic '0', '1', biến còn lại và bù của biến còn lại.

Ví dụ 6

- Thực hiện hàm sau sử dụng bộ MUX 4-1:

$$f(A, B, C) = \sum 2, 4, 7$$

- Triển khai hàm f :

$$f(A, B, C) = \overline{A}.B.\overline{C} + A.\overline{B}.\overline{C} + A.B.C$$

Ví dụ 6

- Bảng thật:

Minterm	A	B	C	$f(A,B,C)$
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

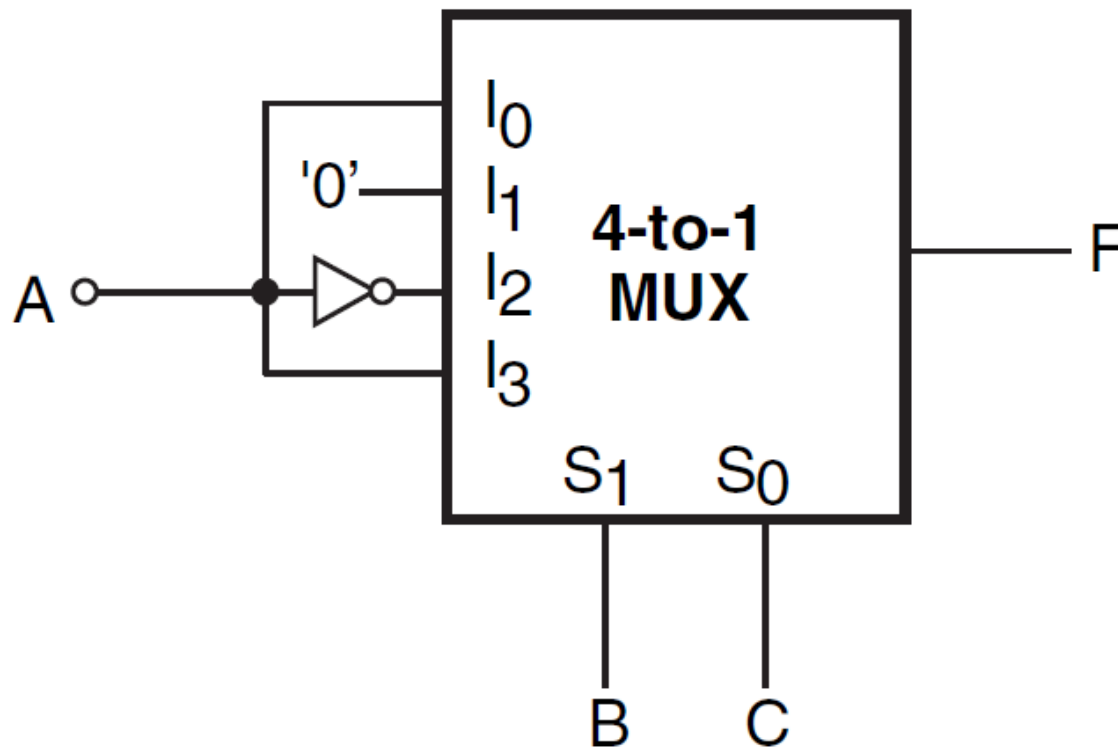
Ví dụ 6

- Chọn A kết nối với đầu vào, B và C kết nối với đầu chọn lọc.
- Xây dựng bảng loại:
- Nếu cả hai minterm không được bôi đậm, viết '0' ở dưới.
- Nếu cả 2 được bôi đậm, viết '1' ở dưới.
- Nếu 1 trong 2 được bôi đậm thì biến tương ứng được viết (ở dạng bù hoặc không bù).

	I_0	I_1	I_2	I_3
\bar{A}	0	1	2	3
A	4	5	6	7
	A	0	\bar{A}	A

Ví dụ 6

- Sơ đồ mạch:

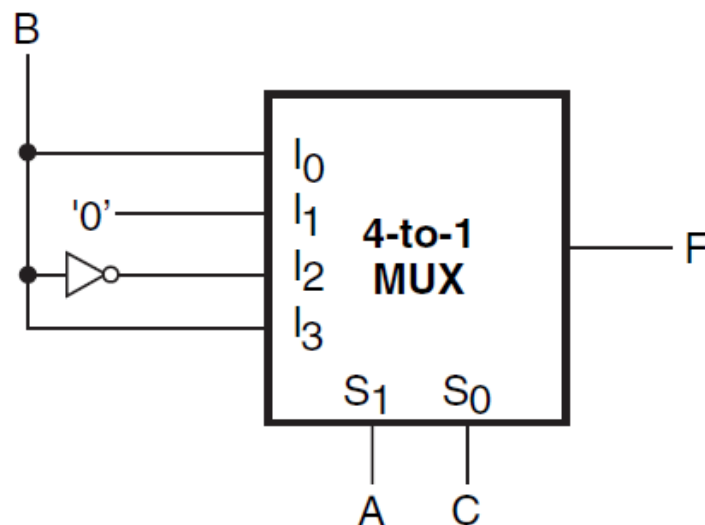


Ví dụ 6

- Nếu chọn B kết nối với đầu vào:

	I_0	I_1	I_2	I_3
\overline{B}	0	1	4	5
B	2	3	6	7
	B	0	\overline{B}	B

- Sơ đồ mạch:

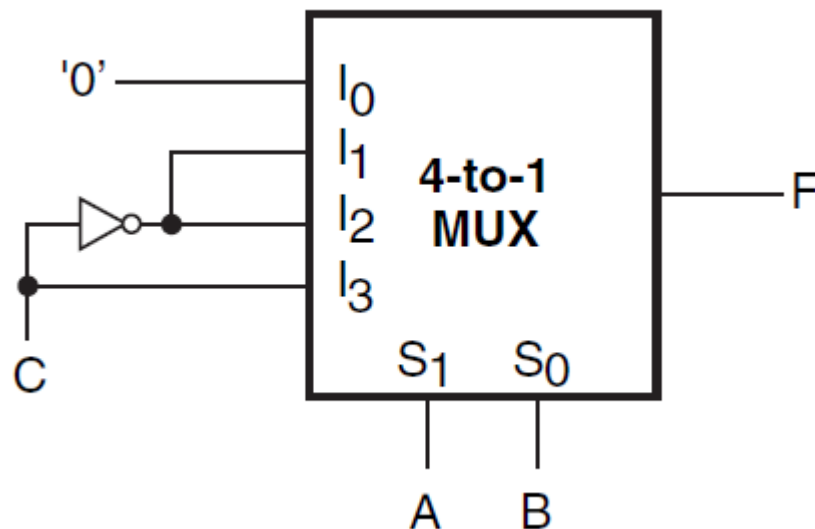


Ví dụ 6

- Nếu chọn C kết nối với đầu vào:

	I_0	I_1	I_2	I_3
\overline{C}	0	2	4	6
C	1	3	5	7
	0	\overline{C}	\overline{C}	C

- Sơ đồ mạch:



Một số hệ tổ hợp cơ bản

- Các mạch số học cơ bản
- Bộ chọn kênh
- Bộ mã hóa
- Bộ phân kênh
- Bộ giải mã

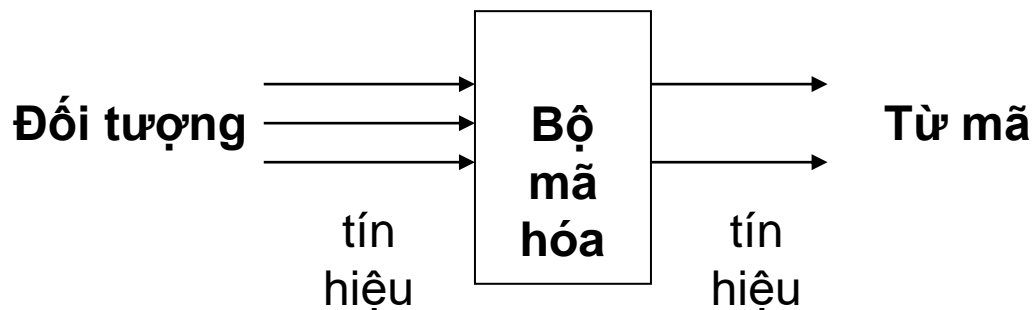
Bộ mã hóa

- Mã hóa là việc sử dụng ký hiệu để biểu diễn đặc trưng cho một đối tượng nào đó.
- Ký hiệu tương ứng với một đối tượng được gọi là từ mã.
- Ví dụ:

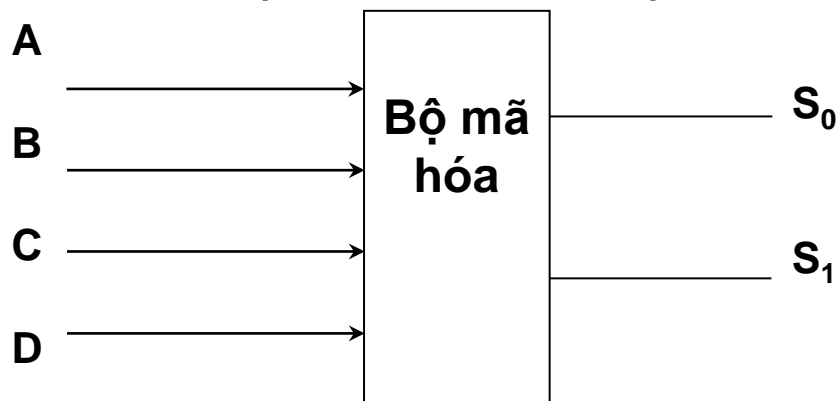
Đối tượng	Từ mã thập phân	Từ mã nhị phân
A	0	00
B	1	01
C	2	10
D	3	11

Bộ mã hóa

- Bộ mã hóa cũng là một bộ chọn kênh với 2^n (hoặc ít hơn) đầu vào và n đầu ra.
- Thực hiện việc mã hóa các tín hiệu tương ứng với các đối tượng thành các từ mã nhị phân.



- Ví dụ:



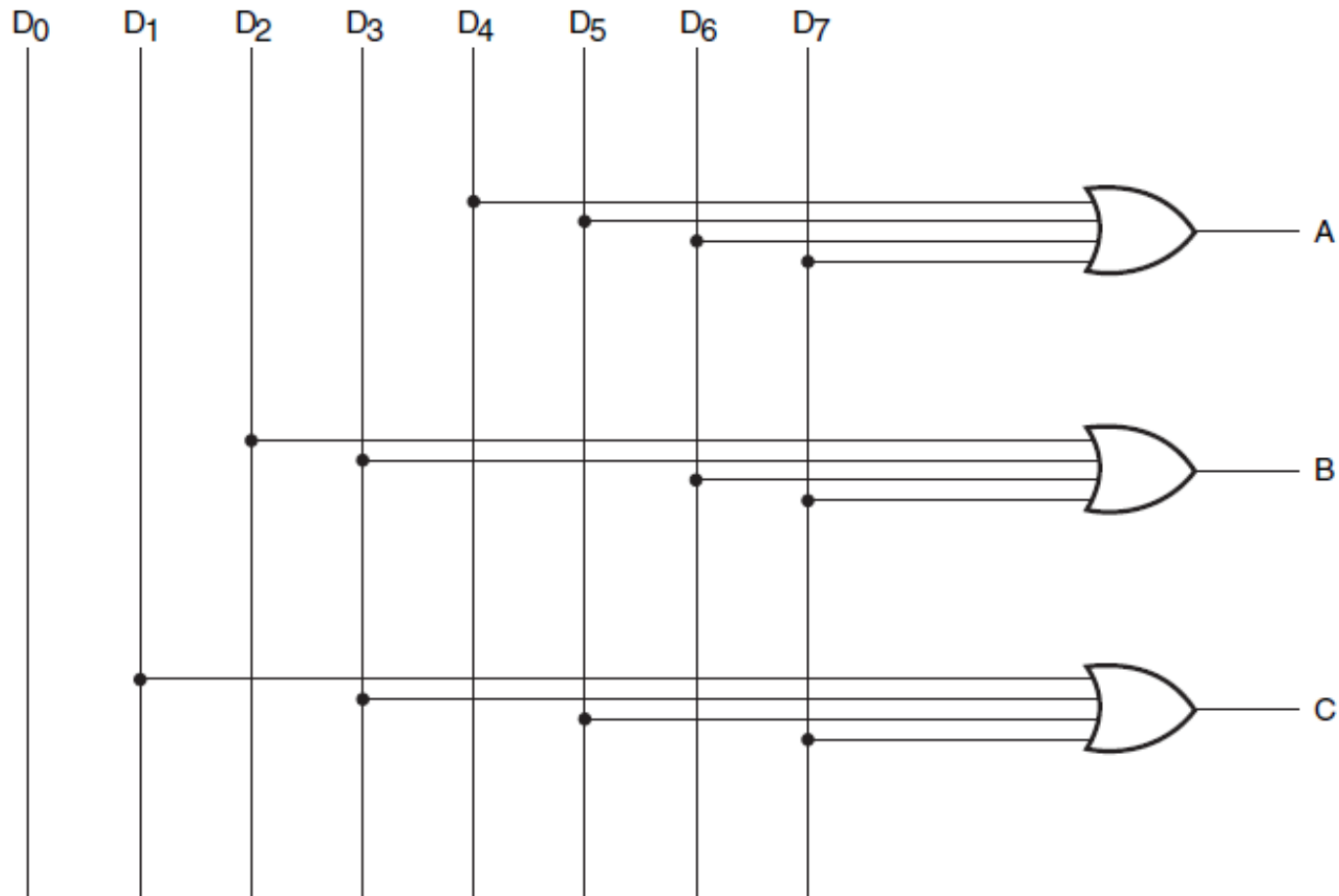
Ví dụ 8

- Thiết kế bộ mã hóa 8 – 3
- Bảng thật:

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	A	B	C
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

Ví dụ 8

- Sơ đồ mạch:



Bộ mã hóa ưu tiên

- Là một dạng thực tế của bộ mã hóa.
- Mỗi đầu vào được gán cho một độ ưu tiên.
- Khi có nhiều đầu vào thì đầu vào có độ ưu tiên cao nhất sẽ được mã hóa.

Ví dụ 9

- Thiết kế bộ mã hóa ưu tiên 4 – 2
 - Đầu vào và ra tích cực ở mức cao
 - Độ ưu tiên cao hơn dành cho các bit ở vị trí cao.
- Bảng thật:

D_0	D_1	D_2	D_3	X	Y
1	0	0	0	0	0
X	1	0	0	0	1
X	X	1	0	1	0
X	X	X	1	1	1

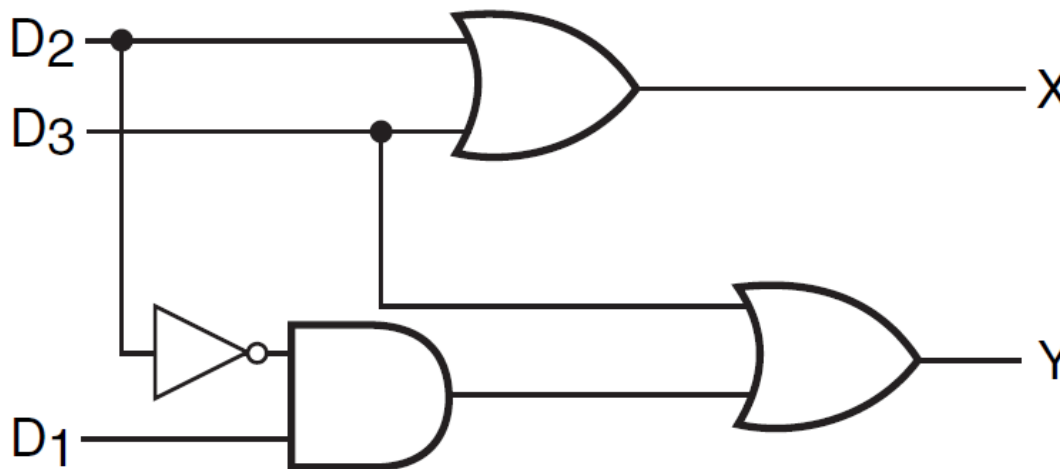
Ví dụ 9

- Hàm Boolean:

$$X = D_2 \cdot \overline{D_3} + D_3 = D_2 + D_3$$

$$Y = D_1 \cdot \overline{D_2} \cdot \overline{D_3} + D_3 = D_1 \cdot \overline{D_2} + D_3$$

- Sơ đồ mạch:



Một số hệ tổ hợp cơ bản

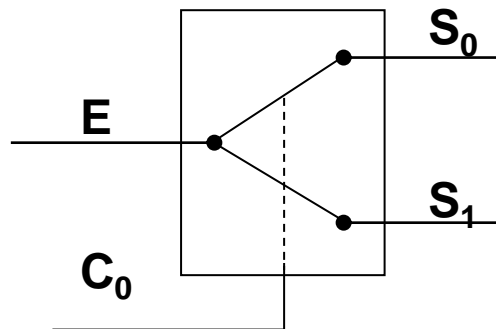
- Các mạch số học cơ bản
- Bộ chọn kênh
- Bộ mã hóa
- Bộ phân kênh
- Bộ giải mã

Bộ phân kênh

- DeMultiPlexor – DeMUX
- Có 1 đầu vào tín hiệu và 2^n đầu ra
- Chức năng: đưa tín hiệu từ đầu vào tới một trong những đầu ra

Bộ phân kênh 1-2

- Sơ đồ khối:

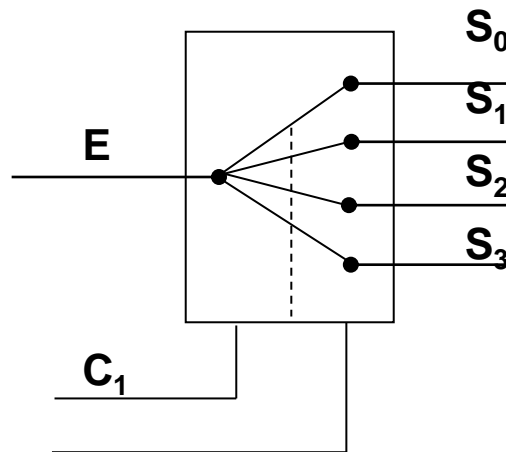


- Tín hiệu chọn:

C_0	S_0	S_1
0	E	0
1	0	E

Bộ phân kênh 1-4

- Sơ đồ khối:



- Tín hiệu chọn: C_0

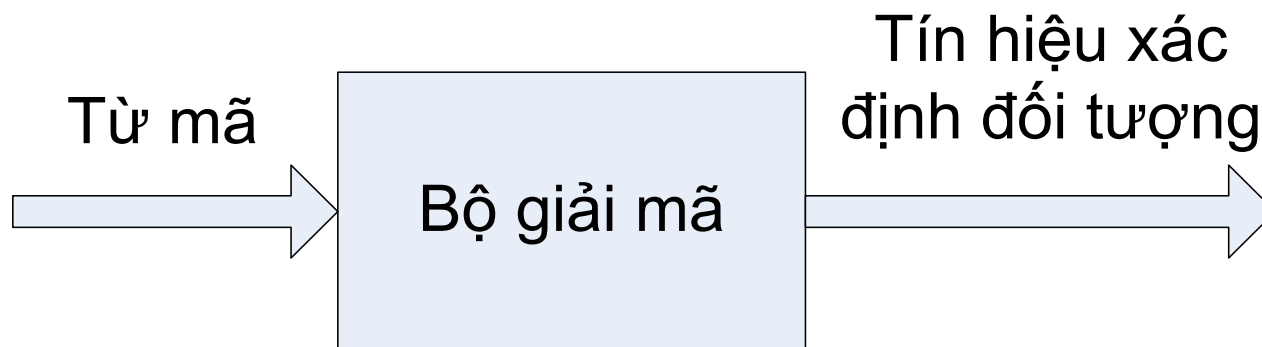
C_1	C_0	S_0	S_1	S_2	S_3
0	0	E	0	0	0
0	1	0	E	0	0
1	0	0	0	E	0
1	1	0	0	0	E

Một số hệ tổ hợp cơ bản

- Các mạch số học cơ bản
- Bộ chọn kênh
- Bộ mã hóa
- Bộ phân kênh
- Bộ giải mã

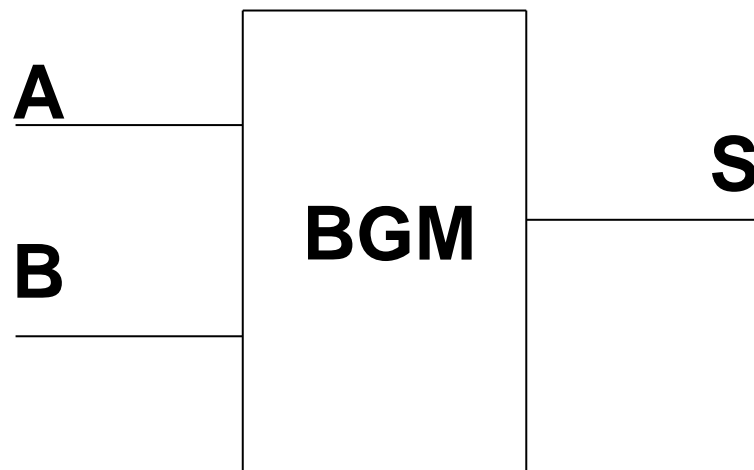
Bộ giải mã

- Bộ giải mã thực hiện chức năng ngược với bộ mã hóa.
- Cung cấp tín hiệu để xác định đối tượng đã mã hóa ở đầu ra khi tổ hợp các biến nhị phân ở đầu vào ứng với 1 hay nhiều từ mã đã được chọn.
- Từ từ mã xác định được tín hiệu tương ứng với đối tượng đã mã hóa.



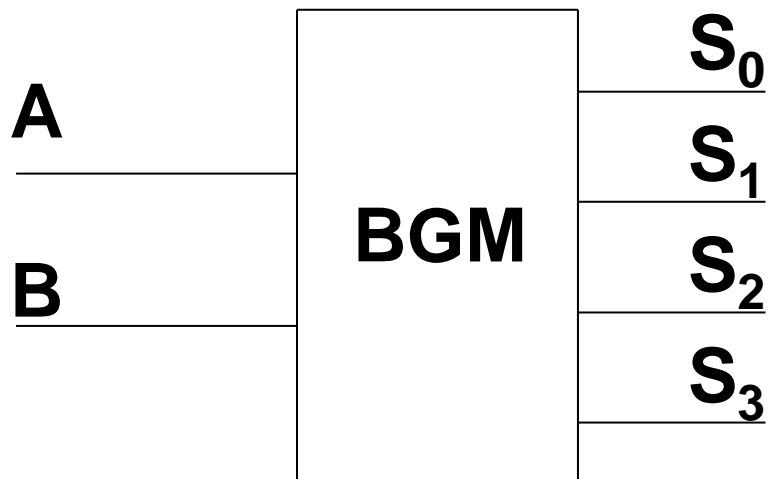
Bộ giải mã

- Giải mã cho 1 từ mã:
 - Nguyên lý: ứng với một tổ hợp cần giải mã ở đầu vào thì đầu ra bằng 1, các tổ hợp đầu vào còn lại, đầu ra bằng 0.
 - VD: $S = 1$ nếu $(AB) = (10)$, $S = 0$ nếu $(AB) \neq (10)$



Bộ giải mã

- Giải mã cho 1 từ mã:
 - Nguyên lý: ứng với một tổ hợp cần giải mã ở đầu vào thì đầu ra bằng 1, các tổ hợp đầu vào còn lại, đầu ra bằng 0.
 - VD: $S = 1$ nếu $(AB) = (10)$, $S = 0$ nếu $(AB) \neq (10)$
- Giải mã cho toàn bộ mã:
 - Nguyên lý: ứng với một tổ hợp nào đó ở đầu vào thì 1 trong các đầu ra bằng 1, các đầu ra còn lại bằng 0.



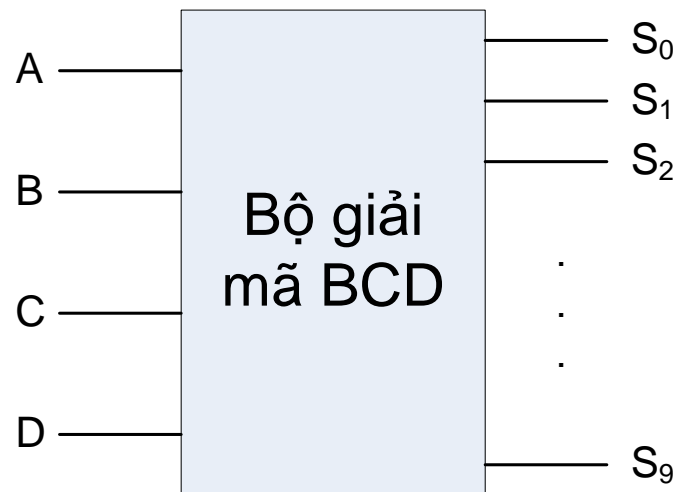
Ví dụ 10

- Thiết kế bộ giải mã BCD - Mã hóa số nguyên thập phân bằng nhị phân

Chữ số thập phân	Từ mã nhị phân
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Ví dụ 10

- Xác định đầu vào và đầu ra:
 - Vào: từ mã nhị phân 4 bit (\Rightarrow có 16 tổ hợp)
 - Ra: các tín hiệu tương ứng với các số nhị phân mà từ mã mã hóa
- Chỉ sử dụng 10 tổ hợp, 6 tổ hợp không sử dụng đến được coi là không xác định.



Ví dụ 10

- Bảng thật:

ABCD	S ₀	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉
0000	1	0	0	0	0	0	0	0	0	0
0001	0	1	0	0	0	0	0	0	0	0
0010	0	0	1	0	0	0	0	0	0	0
0011	0	0	0	1	0	0	0	0	0	0
0100	0	0	0	0	1	0	0	0	0	0
0101	0	0	0	0	0	1	0	0	0	0
0110	0	0	0	0	0	0	1	0	0	0
0111	0	0	0	0	0	0	0	1	0	0
1000	0	0	0	0	0	0	0	0	1	0
1001	0	0	0	0	0	0	0	0	0	1
1010	-	-	-	-	-	-	-	-	-	-
1011	-	-	-	-	-	-	-	-	-	-
1100	-	-	-	-	-	-	-	-	-	-
1101	-	-	-	-	-	-	-	-	-	-
1110	-	-	-	-	-	-	-	-	-	-
1111	-	-	-	-	-	-	-	-	-	-

□

Ví dụ 10

- Tìm biểu thức của từng đầu ra: S_0 và S_1

$$S_0(A,B,C,D) = \overline{A} \overline{B} \overline{C} \overline{D}$$

$$S_1(A,B,C,D) = \overline{A} \overline{B} \overline{C} D$$

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

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

Ví dụ 10

- Tìm biểu thức của từng đầu ra: S_2 và S_3

$$S_2(A,B,C,D) = \overline{B} C \overline{D}$$

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

$$S_3(A,B,C,D) = \overline{B} C D$$

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

Ví dụ 10

- Tìm biểu thức của từng đầu ra: S_4 và S_5

$$S_4(A,B,C,D) = B \bar{C} \bar{D}$$

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

$$S_5(A,B,C,D) = B \bar{C} D$$

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

Ví dụ 10

- Tìm biểu thức của từng đầu ra: S_6 và S_7

$$S_6(A,B,C,D) = BC \bar{D}$$

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

$$S_7(A,B,C,D) = BCD$$

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

Ví dụ 10

- Tìm biểu thức của từng đầu ra: S_8 và S_9

$$S_8(A,B,C,D) = A \bar{D}$$

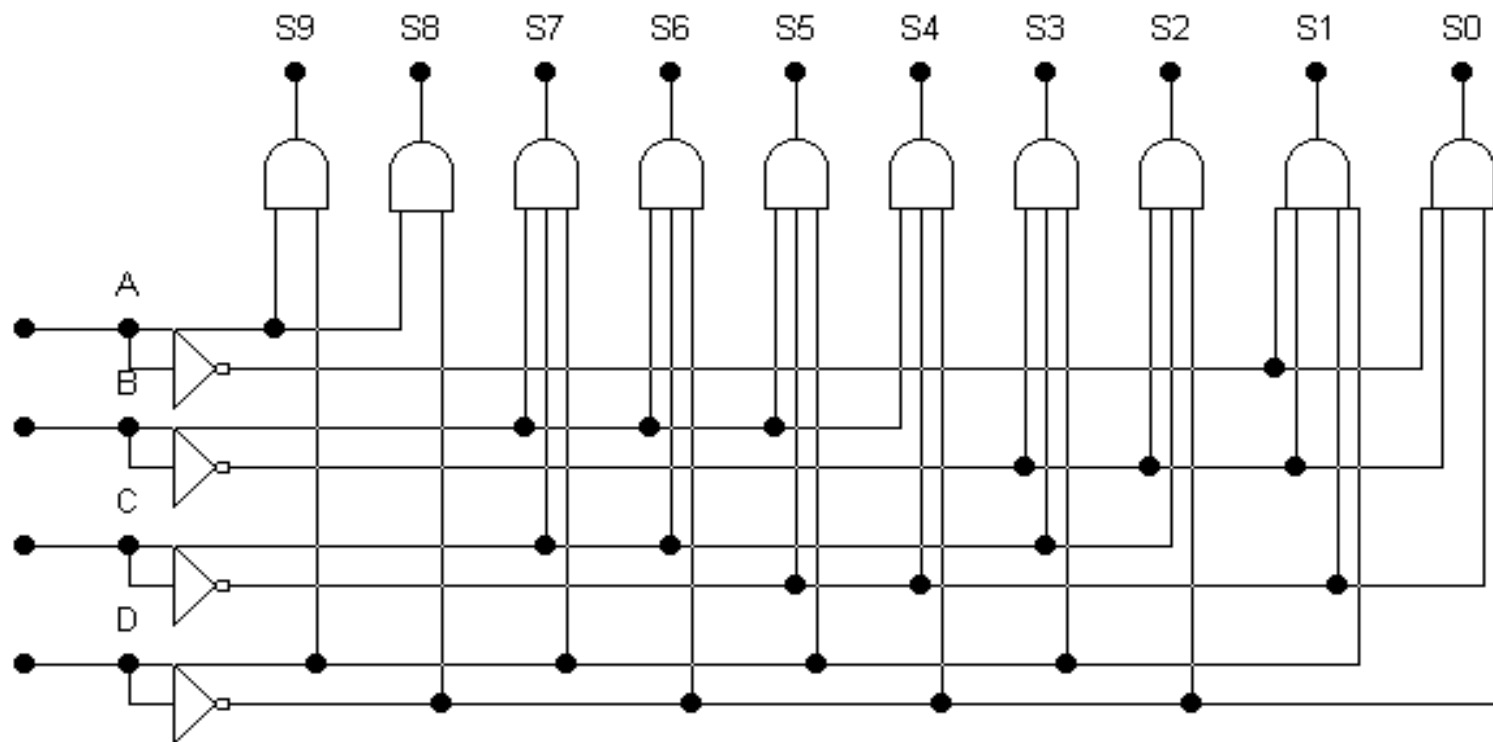
CD \ AB	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	-	-	-	-
10	1	0	-	-

$$S_9(A,B,C,D) = AD$$

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

Ví dụ 10

- Sơ đồ mạch:



Thực hiện các hàm Boolean bằng bộ giải mã

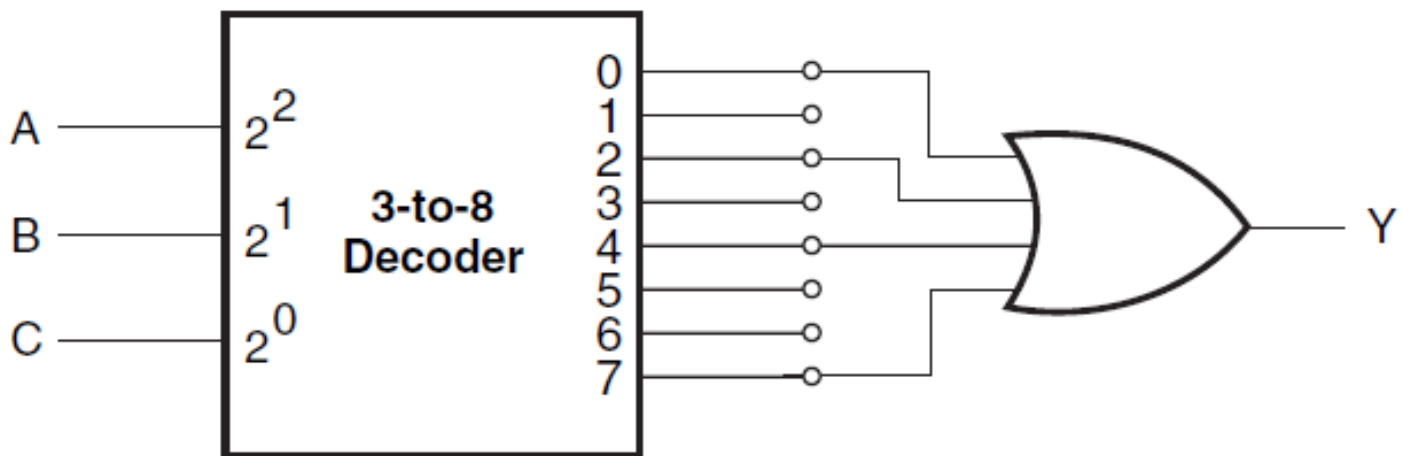
- Bộ giải mã có thể được sử dụng để thực hiện một hàm Boolean.
- Bộ giải mã tạo ra số lượng hội cần thiết và sau đó sử dụng một cổng OR để tạo ra tổng các hội.

Ví dụ 11

- Thực hiện hàm sau sử dụng bộ giải mã:

$$Y = A.\overline{B}.\overline{C} + \overline{A}.B.\overline{C} + A.B.C + \overline{A}.\overline{B}.C$$

- Sơ đồ mạch:

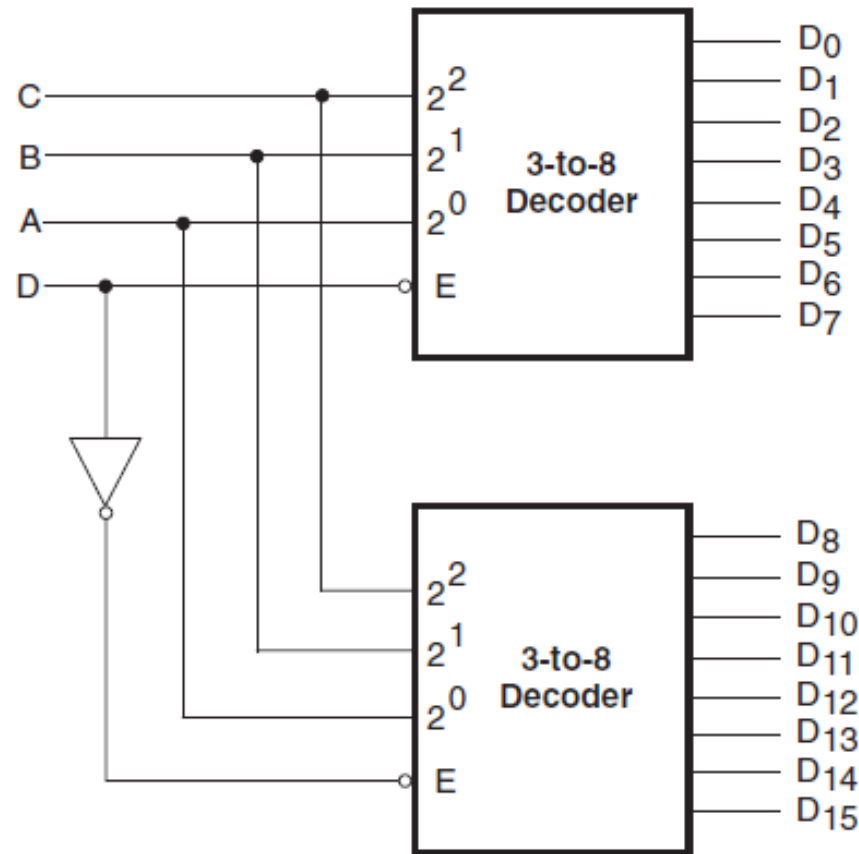


Bộ giải mã nhiều tầng

- Nếu số lượng đầu vào và đầu ra của bộ giải mã không đủ nhiều, có thể ghép một số bộ giải mã để tạo ra một bộ giải mã có số lượng đầu vào và đầu ra lớn hơn.
- Nếu số lượng đầu vào cho phép 1 IC là 2^n , trong khi số lượng đầu vào mong muốn là 2^N ($N > n$), thì số lượng bộ giải mã cần sử dụng là 2^{N-n} .
- Kết nối các bit nhỏ nhất của các đầu vào với nhau.
- Các bit còn lại của đầu vào lựa chọn được sử dụng để quyết định bật/tắt các IC.
- Đầu ra của các bộ giải mã được ghép lại với nhau thành đầu ra cuối cùng.

Ví dụ 12

- Xây dựng bộ giải mã 4-16 sử dụng 2 bộ giải mã 3-8 có đầu vào tích cực ở mức thấp.



Ví dụ 13

- Thực hiện bộ cộng đầy đủ sử dụng bộ giải mã 3-8.
- Bảng thật:

<i>A</i>	<i>B</i>	<i>C</i>	<i>S</i>	<i>C_o</i>
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

Ví dụ 13

- Hàm Boolean:

$$\text{Sum output } S = \Sigma 1, 2, 4, 7$$

$$\text{Carry output } C_o = \Sigma 3, 5, 6, 7$$

- Sơ đồ mạch:

