

Chương 6

ĐẠI SỐ BOOLEAN VÀ MẠCH LOGIC

1

Nội dung

- 6.1. Giới thiệu
- 6.2. Đại số Boolean
- 6.3. Hàm Boolean
- 6.4. Các công luận lý
- 6.5. Mạch Logic
- 6.6. Thiết kế của mạch kết hợp
- 6.7. Câu hỏi và bài tập

GIỚI THIỆU

- Đại số Boole được phát minh bởi nhà toán học Anh George Boole vào năm 1854.
- Đại số Boole nghiên cứu các phép toán thực hiện trên các biến chỉ có 2 giá trị 0 và 1, tương ứng với hai trạng thái luận lý "sai" và "đúng" (hay "không" và "có") của đời thường.

GIỚI THIỆU

- Tương tự các hệ đại số khác được xây dựng thông qua những vấn đề cơ bản sau:
 - **Miền** (domain) là tập hợp (set) các phần tử (element)
 - **Các phép toán** (operation) thực hiện được trên miền
 - **Các định đề** (postulate), **hay tiên đề** (axiom) được công nhận không qua chứng minh
 - **Tập các hệ quả** (set of consequences) được suy ra từ định đề, định lý (theorem), định luật (law) hay luật(rule)

NHỮNG NGUYÊN TẮC CƠ BẢN

- Sử dụng hệ cơ số nhị phân.
- Các phép toán:
 - Phép cộng luận lí (logical addition) : (+) hay (OR)
 - Phép nhân luận lí (logical multiplication): (.) hay (AND)
 - Phép bù (NOT)
- Độ ưu tiên của các phép toán
- **Tính đóng** (closure): tồn tại miền B với ít nhất 2 phần tử phân biệt và 2 phép toán (+) và (•) sao cho: Nếu x và y là các phần tử thuộc B thì $(x + y)$, $(x \cdot y)$ cũng là 1 phần tử thuộc B

PHÉP CỘNG LUẬN LÍ

Phép toán: Dấu '+' hay OR

Biểu thức : $A + B = C$
Hay $A \text{ OR } B = C$

Nguyên tắc:

- Kết quả trả về 0 (FALSE) khi và chỉ khi tất cả giá trị đầu vào là 0 (FALSE).
- Kết quả là 1 (TRUE) khi có bất kì một giá trị nhập vào có giá trị là 1 (TRUE).

Ví dụ:

| Input | | | Output |
|-------|---|---|--------|
| A | + | B | = C |
| 0 | | 0 | 0 |
| 0 | | 1 | 1 |
| 1 | | 0 | 1 |
| 1 | | 1 | 1 |

| | |
|------------------|-----------------|
| A | 1 0 0 1 1 0 1 0 |
| B | 1 1 0 0 1 0 0 1 |
| A + B hay A OR B | 1 1 0 1 1 0 1 1 |

PHÉP NHÂN LUẬN LÍ

Phép toán: Dấu '.' hay AND

Biểu thức : $A . B = C$

Hay $A \text{ AND } B = C$

Nguyên tắc:

- Kết quả trả về 1 (TRUE) khi và chỉ khi tất cả giá trị đầu vào là 1 (TRUE).
- Kết quả là 0 (FALSE) khi có bất kì một giá trị nhập vào có giá trị là 0 (FALSE).

| Input | | Output |
|-------|---|-------------|
| A | B | $A . B = C$ |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Ví dụ:

| | |
|----------------------|-----------------|
| A | 1 0 0 1 1 0 1 0 |
| B | 1 1 0 0 1 0 0 1 |
| A . B hay A AND B | 1 0 0 0 1 0 0 0 |

PHÉP BÙ

Phép toán: Dấu '-' hay NOT (phép toán một ngôi)

Biểu thức : \bar{A}
Hay NOT A

Nguyên tắc:

- Kết quả trả về 1 (TRUE) nếu giá trị đầu vào là 0 (FALSE).
- Ngược lại, kết quả là 0 (FALSE) nếu giá trị nhập vào là 1 (TRUE).

| Input | Output |
|-------|-----------|
| A | \bar{A} |
| 0 | 1 |
| 1 | 0 |

Ví dụ:

| | |
|---------------------|-----------------|
| | 1 0 0 1 1 0 1 0 |
| A | |
| \bar{A} hay NOT A | 0 1 1 0 0 1 0 1 |

ĐỘ ƯU TIÊN CỦA CÁC PHÉP TOÁN

- Biểu thức được tính từ trái sang phải.
- Biểu thức trong ngoặc đơn được đánh giá trước.
- Các phép toán bù (NOT) được ưu tiên tiếp theo.
- Tiếp theo là các phép toán '.' (AND).
- Cuối cùng là các phép toán '+' (OR).

BẢNG CHÂN TRỊ TỔNG QUÁT

| A | B | NOT A | A AND B | A OR B | A XOR B |
|---|---|-------|---------|--------|---------|
| 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 |

Ví dụ: $C = A \text{ Or } B \text{ Not } A$

| | |
|---|-----------------|
| A | 1 0 0 1 1 0 1 0 |
| B | 1 1 0 0 1 0 0 1 |
| C | ?????????? |

Độ ưu tiên của các phép toán giảm dần từ trái sang phải

CÁC ĐỊNH ĐỀ *Huntington* CỦA ĐẠI SỐ BOOLEAN

○ Định đề 1:

- $A = 0$ khi và chỉ khi A không bằng 1
- $A = 1$ khi và chỉ khi A không bằng 0

○ Định đề 2: Phần tử đồng nhất

- $x + 0 = x$
- $x \cdot 1 = x$

○ Định đề 3: Tính giao hoán

- $x + y = y + x$
- $x \cdot y = y \cdot x$

○ Định đề 4: Tính kết hợp

- $x + (y + z) = (x + y) + z$
- $x \cdot (y \cdot z) = (x \cdot y) \cdot z$

○ Định đề 5: Tính phân phối

- $x \cdot (y + z) = x \cdot y + x \cdot z$
- $x + y \cdot z = (x + y) \cdot (x + z)$

○ Định đề 6: Tính bù

- $x + \overline{x} = 1$
- $x \cdot \overline{x} = 0$

NGUYÊN LÝ ĐỐI NGẪU

- Đại số Boolean mang tính đối ngẫu
 - Đổi phép toán (+) thành (•)
 - Đổi phần tử đồng nhất 0 thành 1

| | Cột 1 | Cột 2 | Column 3 |
|-------|-----------------|-----------------------------|-----------------|
| Row 1 | $1 + 1 = 1$ | $1 + 0 = 0 + 1 = 1$ | $0 + 0 = 0$ |
| Row 2 | $0 \cdot 0 = 0$ | $0 \cdot 1 = 1 \cdot 0 = 0$ | $1 \cdot 1 = 1$ |

CÁC ĐỊNH LÝ CỦA ĐẠI SỐ BOOLEAN

○ Định lý 1 (Luật lũy đẳng)

- $x + x = x$
- $x \cdot x = x$

○ Định lý 2 (Định luật nuốt)

- $x + 1 = 1$
- $x \cdot 0 = 0$

○ Định lý 3 (Định luật hấp thu)

- $x + x \cdot y = x$
- $x \cdot (x + y) = x$

○ Định lý 4 (Định luật bù kép)

$$\overline{\overline{x}} = x$$

○ Định lý 5

$$(a) x \cdot (\bar{x} + y) = x \cdot y$$

$$(b) x + \bar{x} \cdot y = x + y$$

○ Định lý 6 (Định luật De Morgan)

$$(a) \overline{x + y} = \bar{x} \cdot \bar{y}$$

$$(b) \overline{x \cdot y} = \bar{x} + \bar{y}$$

HÀM BOOLEAN

- Một hàm Boolean là một biểu thức được tạo từ:
 - Các biến nhị phân,
 - Các phép toán hai ngôi OR và AND, phép toán một ngôi NOT,
 - Các cặp dấu ngoặc đơn và dấu bằng.
- Với giá trị cho trước của các biến, giá trị của hàm chỉ có thể là 0 hoặc 1.
- Phương trình

$$W = X + \bar{Y} . Z \quad \text{Hay} \quad W = f(X, Y, Z)$$

Với: X, Y và Z được gọi là các biến của hàm.

HÀM BOOLEAN

- Một hàm Boole cũng có thể được biểu diễn bởi dạng bảng chân trị. Số hàng của bảng là 2^n , n là số các biến nhị phân được sử dụng trong hàm.

| X | Y | Z | W |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

Bảng chân trị của hàm Boole
 $W = x + \bar{y} \cdot z$

SỰ DƯ THỪA (redundant)

- Khái niệm:

- Literal: là 1 biến hay phủ định của biến đó (A hay \bar{A})
- Term của n literal là sự kết hợp của các literal mà mỗi biến chỉ xuất hiện một lần duy nhất.

Ví dụ: term của 3 biến A, B, C là $A.B.C$

- Một biểu thức gọi là dư thừa nếu nó có chứa

- Literal lặp: xx hay $x+x$
- Biến và bù của biến: xx' hay $x+x'$
- Hằng: 0 hay 1

- Các thành phần dư thừa có thể loại bỏ khỏi biểu thức

- Các thành phần thừa trong biểu thức không cần hiện thực trong phần cứng

TỐI THIỂU HÀM BOOLEAN

- Tối thiểu hàm Boolean là việc tối ưu hóa số lượng phần tử và số hạng để tạo ra một mạch với số lượng phần tử ít hơn.
- **Phương pháp:** sử dụng phương pháp đại số, áp dụng các định lý, định đề, các luật,...cắt-và-thử nhiều lần để tối thiểu hàm Boolean tới mức thấp nhất.
- **Ví dụ:**

$$(a) x + \bar{x}.y$$

$$= (x + \bar{x}).(x + y) \quad \text{theo định đề 5(b)}$$

$$= 1.(x + y) \quad \text{theo định đề 6(a)}$$

$$= (x + y).1 \quad \text{theo định đề 3(b)}$$

$$= x + y \quad \text{theo định đề 2(b)}$$

$$(b) x.(\bar{x} + y)$$

$$= (x.\bar{x}) + (x.y) \quad \text{theo định đề 5(a)}$$

$$= 0 + (x.y) \quad \text{theo định đề 6(b)}$$

$$= (x.y) + 0 \quad \text{theo định đề 3(a)}$$

$$= x.y \quad \text{theo định đề 2(a)}$$

TỐI THIỂU HÀM BOOLEAN

- Hãy rút gọn biểu thức sau: $x.y + \bar{x}.z + y.z$
$$= x.y + \bar{x}.z + y.z.1$$
$$= x.y + \bar{x}.z + y.z.(x + \bar{x})$$
$$= x.y + \bar{x}.z + y.z.x + y.z.\bar{x}$$
$$= x.y + \bar{x}.z + x.y.z + \bar{x}.y.z$$
$$= x.y.1 + \bar{x}.z + x.y.z + \bar{x}.y.z$$
$$= x.y.1 + x.y.z + \bar{x}.z + \bar{x}.y.z$$
$$= x.y.(1 + z) + \bar{x}.z.(1 + y)$$
$$= x.y.(z + 1) + \bar{x}.z.(y + 1) = x.y.1 + \bar{x}.z.1 = x.y + \bar{x}.z$$

PHẦN BÙ CỦA MỘT HÀM

- Phần bù của một hàm Boolean F là \bar{F} có được bằng cách thay 0 thành 1 và 1 thành 0 trong bảng chân trị của hàm đó

| x | y | z | F | \bar{F} |
|---|---|---|---|-----------|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 |

$$F = x.\bar{y} + \bar{x}z$$

PHẦN BÙ CỦA MỘT HÀM

○ Ví dụ: Áp dụng định lí De Morgan

Tìm đối ngẫu của các hàm cho ở dưới:

$$(a) F_1 = \bar{x} \cdot y \cdot \bar{z} + \bar{x} \cdot \bar{y} \cdot z$$

$$(b) F_2 = x \cdot (\bar{y} \cdot \bar{z} + y \cdot z)$$

$$\begin{aligned}(a) \quad \overline{F_1} &= \overline{\bar{x} \cdot y \cdot \bar{z} + \bar{x} \cdot \bar{y} \cdot z} \\ &= (\overline{\bar{x} \cdot y \cdot \bar{z}}) \cdot (\overline{\bar{x} \cdot \bar{y} \cdot z}) \\ &= (\overline{\bar{x}} + \overline{y} + \overline{\bar{z}}) \cdot (\overline{\bar{x}} + \overline{\bar{y}} + \overline{z}) \\ &= (x + \bar{y} + z) \cdot (x + y + \bar{z})\end{aligned}$$

$$\begin{aligned}(b) \quad \overline{F_2} &= \overline{x \cdot (\bar{y} \cdot \bar{z} + y \cdot z)} \\ &= \bar{x} + \overline{(\bar{y} \cdot \bar{z} + y \cdot z)} \\ &= \bar{x} + (\overline{\bar{y} \cdot \bar{z}}) \cdot (\overline{y \cdot z}) \\ &= \bar{x} + (y + z) \cdot (\bar{y} + \bar{z}) \\ &= \bar{x} + (y + z) \cdot (\bar{y} + \bar{z})\end{aligned}$$

PHẦN BÙ CỦA MỘT HÀM

- Ví dụ: Tìm phần bù của các hàm F_1 và F_2 bằng cách tìm đối ngẫu

$$(a) F_1 = \bar{x}.y.\bar{z} + \bar{x}.\bar{y}.z$$

$$(b) F_2 = x.(\bar{y}.\bar{z} + y.z)$$

Giải

$$(a) F_1 = \bar{x}.y.\bar{z} + \bar{x}.\bar{y}.z$$

Đối ngẫu của F_1 là $(\bar{x} + y + \bar{z}) . (\bar{x} + \bar{y} + z)$

Phần bù của mỗi phần tử ta có

$$\bar{F}_1 = (x + \bar{y} + z) . (x + y + \bar{z})$$

$$(b) F_2 = x.(\bar{y}.\bar{z} + y.z)$$

Đối ngẫu của F_2 là: $x + (\bar{y} + \bar{z}).(y + z)$

Phần bù của mỗi phần tử ta có

$$\bar{F}_2 = \bar{x} + (y + z).(\bar{y} + \bar{z})$$

DẠNG CHÍNH TẮC CỦA HÀM BOOLEAN

(*canonic form*)

Một biểu thức n biến luôn có thể được biểu diễn dưới 2 dạng:

- **Dạng tổng các tích (sum-of-product hay s-o-p):** biểu thức được biểu diễn dưới dạng tổng (sum) các toán hạng (term), mỗi toán hạng là tích (product) của các literal

$$E = x y + x y' z + x' y z'$$

- **Dạng tích các tổng (product-of-sum hay p-o-s):** biểu thức được biểu diễn dưới dạng tích các toán hạng, mỗi toán hạng là tổng của các literal

$$E = (x + y) (x + y' + z) (x' + y + z')$$

- **Dạng chính tắc:** biểu thức n biến dạng s-o-p hay p-o-s có đặc điểm mỗi toán hạng của nó có đủ mặt n literal và không chứa các literal thừa

DẠNG CHÍNH TẮC CỦA HÀM BOOLEAN

(*canonic form*)

- Luôn có thể biến đổi một s-o-p (hay p-o-s) không chính tắc (noncanonic) về dạng chính tắc
- Vd: $E = xy' + x'y + xz + yz$
$$= xy'(z + z') + x'y(z + z') + xz(y + y') + yz(x + x')$$
$$= xy'z + xy'z' + x'yz + x'yz' + xyz + xy'z + xyz + x'yz$$
$$= xy'z + xy'z' + x'yz + x'yz' + xyz$$

DẠNG CHÍNH TẮC CỦA HÀM BOOLEAN

(*canonic form*)

- **Minterm:** một tích không dư thừa các literal của dạng chính tắc (Thực hiện phép toán AND giữa các literal tạo thành một Term)
- **Maxterm:** một tổng không dư thừa các literal của dạng chính tắc (Thực hiện phép toán OR giữa các literal tạo thành một Term)

| Variables | | | Minterms | | Maxterms | |
|-----------|---|---|---------------------------------------|-------------|-------------------------------|-------------|
| x | y | z | Term | Designation | Term | Designation |
| 0 | 0 | 0 | $\bar{x} \cdot \bar{y} \cdot \bar{z}$ | m_0 | $x + y + z$ | m_0 |
| 0 | 0 | 1 | $\bar{x} \cdot \bar{y} \cdot z$ | m_1 | $x + y + \bar{z}$ | m_1 |
| 0 | 1 | 0 | $\bar{x} \cdot y \cdot \bar{z}$ | m_2 | $x + \bar{y} + z$ | m_2 |
| 0 | 1 | 1 | $\bar{x} \cdot y \cdot z$ | m_3 | $x + \bar{y} + \bar{z}$ | m_3 |
| 1 | 0 | 0 | $x \cdot \bar{y} \cdot \bar{z}$ | m_4 | $\bar{x} + y + z$ | m_4 |
| 1 | 0 | 1 | $x \cdot \bar{y} \cdot z$ | m_5 | $\bar{x} + y + \bar{z}$ | m_5 |
| 1 | 1 | 0 | $x \cdot y \cdot \bar{z}$ | m_6 | $\bar{x} + \bar{y} + z$ | m_6 |
| 1 | 1 | 1 | $x \cdot y \cdot z$ | m_7 | $\bar{x} + \bar{y} + \bar{z}$ | m_7 |

Minterms và Maxterms ứng với ba biến
 Maxterms là phần bù của minterms và ngược lại

F2????

TỔNG CÁC TÍCH

- Các bước để biểu diễn hàm Bool theo dạng tổng của các tích:
 1. Xây dựng một bảng chân trị cho hàm Boolean.
 2. Hình thành một minterm cho mỗi sự kết hợp của các biến tạo ra hàm có giá trị là 1
 3. Biểu thức cuối cùng là cộng tất cả các minterm thu được từ bước 2.

Ví dụ: Hàm F1 có giá trị 1 là sự kết hợp của 3 biến 001, 100, 111

Các minterm tương ứng là

$$\bar{x} \cdot \bar{y} \cdot z, \quad x \cdot \bar{y} \cdot \bar{z} \quad \text{và} \quad x \cdot y \cdot z$$

Sau đó, lấy tổng (OR) của tất cả các minterm này, được biểu thức hàm F1 dưới dạng tổng của các tích như sau:

$$F_1 = \bar{x} \cdot \bar{y} \cdot z + x \cdot \bar{y} \cdot \bar{z} + x \cdot y \cdot z \quad \text{hay}$$

$$F_1 = m_1 + m_4 + m_7$$

| x | y | z | F ₁ | F ₂ |
|---|---|---|----------------|----------------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 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 |

Bảng chân trị của hàm F1

Tổng của các tích của biểu thức được kí hiệu:
 $F(A, B, C) = \sum(1, 4, 5, 6, 7)$

TỔNG CÁC TÍCH

- Ví dụ: Tính biểu thức hàm Bool $F = A + B \cdot C$ dưới dạng tổng của Hàm có 3 biến A, B và C. Toán hạng đầu tiên A có 2 biến vắng mặt, cho nên

$$\begin{aligned} A &= A \cdot (B + \bar{B}) \\ &= A \cdot B + A \cdot \bar{B} \end{aligned}$$

Vẫn còn một biến vắng mặt, nên

$$\begin{aligned} A &= A \cdot B \cdot (C + \bar{C}) + A \cdot \bar{B} \cdot (C + \bar{C}) \\ &= A \cdot B \cdot C + A \cdot B \cdot \bar{C} + A \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C} \end{aligned}$$

Toán hạng thứ hai $\bar{B} \cdot C$ có một biến vắng mặt, cho nên

$$\begin{aligned} \bar{B} \cdot C &= \bar{B} \cdot C \cdot (A + \bar{A}) \\ &= A \cdot \bar{B} \cdot C + \bar{A} \cdot \bar{B} \cdot C \end{aligned}$$

Sau đó kết hợp tất cả các toán hạng lại ta được

$$F = A \cdot B \cdot C + A \cdot B \cdot \bar{C} + A \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C} + A \cdot \bar{B} \cdot C + \bar{A} \cdot \bar{B} \cdot C$$

Nhưng biểu thức trên, toán hạng $A \cdot \bar{B} \cdot C$ xuất hiện 2 lần và theo định lí 1(a) ta có $x+x=x$, vì vậy chúng ta có thể bỏ một trong số chúng đi. Sắp xếp lại các minterm theo thứ tự tăng dần, cuối cùng ta thu được:

$$\begin{aligned} F &= \bar{A} \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C} + A \cdot \bar{B} \cdot C + A \cdot B \cdot \bar{C} + A \cdot B \cdot C \\ &= m_1 + m_4 + m_5 + m_6 + m_7 \end{aligned}$$

TÍCH CÁC TỔNG

- Các bước để biểu diễn hàm Bool theo dạng tích của các tổng
 1. Xây dựng một bảng chân trị cho hàm Boolean.
 2. Hình thành một maxterm cho mỗi sự kết hợp của các biến với các biến này thì hàm này có giá trị là 0
 3. Biểu thức cuối cùng là nhân tất cả các maxterm thu được từ bước 2.

Ví dụ: Hàm F1 có giá trị 0 là sự kết hợp của 5 biến 000,010,011, 01, và 110

Các maxterm tương ứng là

$(x + y + z)$, $(x + \bar{y} + z)$, $(x + \bar{y} + \bar{z})$, $(\bar{x} + y + \bar{z})$, và $(\bar{x} + \bar{y} + z)$

Sau đó, lấy tích(AND) của tất cả các maxterm này, được biểu thức hàm F1 dưới dạng tích của các tổng như sau:

$$F_1 = (x + y + z) \cdot (x + \bar{y} + \bar{z}) \cdot (x + \bar{y} + z) \cdot (\bar{x} + y + \bar{z}) \cdot (\bar{x} + \bar{y} + z)$$

$$F_1 = M_0 \cdot M_2 \cdot M_3 \cdot M_5 \cdot M_6$$

Bảng chân trị của hàm F1

| x | y | z | F ₁ | F ₂ |
|---|---|---|----------------|----------------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 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 |

TÍCH CÁC TỔNG

Ví dụ: Tính biểu thức hàm Bool $F = x \cdot y + \bar{x} \cdot z$ dưới dạng tích của các tổng

Đầu tiên chúng ta biến đổi hàm thành toán hạng OR sử dụng luật phân phối

$$\begin{aligned} F &= x \cdot y + \bar{x} \cdot z \\ &= (x \cdot y + \bar{x}) \cdot (x \cdot y + z) \\ &= (x + \bar{x}) \cdot (y + \bar{x}) \cdot (x + z) \cdot (y + z) \\ &= (\bar{x} + y) \cdot (x + z) \cdot (y + z) \end{aligned}$$

Hàm này có 3 biến x, y và z . Mỗi toán hạng OR có một biến vắng mặt, cho nên

$$\begin{aligned} \bar{x} + y &= \bar{x} + y + z \cdot \bar{z} = (\bar{x} + y + z) \cdot (\bar{x} + y + \bar{z}) \\ x + z &= x + z + y \cdot \bar{y} = (x + z + y) \cdot (x + z + \bar{y}) = (x + y + z) \cdot (x + \bar{y} + z) \\ y + z &= x \cdot \bar{x} + y + z = (x + y + z) \cdot (\bar{x} + y + z) \end{aligned}$$

Kết hợp tất cả các toán hạng và bỏ đi những cái xuất hiện nhiều hơn một lần, cuối cùng chúng ta thu được:

$$\begin{aligned} F &= (x + y + z) \cdot (x + \bar{y} + z) \cdot (\bar{x} + y + z) \cdot (\bar{x} + y + \bar{z}) \\ &= M_0 \cdot M_2 \cdot M_4 \cdot M_5 \end{aligned}$$

Một cách để biểu diễn hàm này như sau:

$$F(x, y, z) = \pi(0, 2, 4, 5)$$

Π có nghĩa là phép AND của các toán hạng

SỰ CHUYỂN ĐỔI GIỮA CÁC DẠNG CHÍNH TẮC

- Để chuyển đổi từ một dạng chính tắc này sang một dạng chính tắc khác, đổi các kí hiệu và liệt kê danh sách các tham số không có mặt từ hàm ban đầu.

Ví dụ:

$$F(A, B, C) = \sum(1, 4, 5, 6, 7) \\ = m_1 + m_4 + m_5 + m_6 + m_7$$

Phần bù đó có thể được biểu diễn như sau:

$$F(A, B, C) = \pi(0, 2, 3) = m_0 + m_2 + m_3$$

Áp dụng định lý De Morgan's chúng ta thu được F dưới một dạng khác :

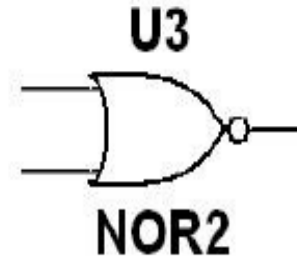
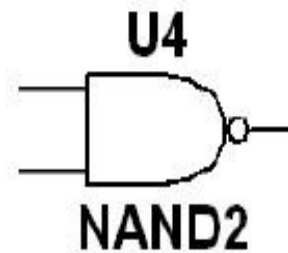
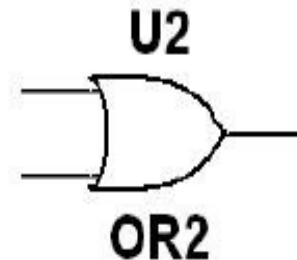
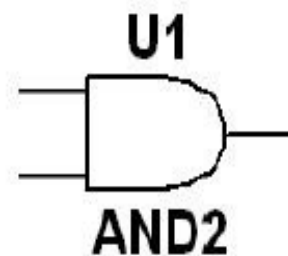
$$F = \overline{m_0 + m_2 + m_3} \\ = \overline{m_0} \cdot \overline{m_2} \cdot \overline{m_3} \\ = M_0 \cdot M_2 \cdot M_3 \\ = \pi(0, 2, 3)$$

Bảng chân trị của hàm F1

| x | y | z | F ₁ | F ₂ |
|---|---|---|----------------|----------------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 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 |

CÁC CỘNG LUẬN LÝ - Cổng AND

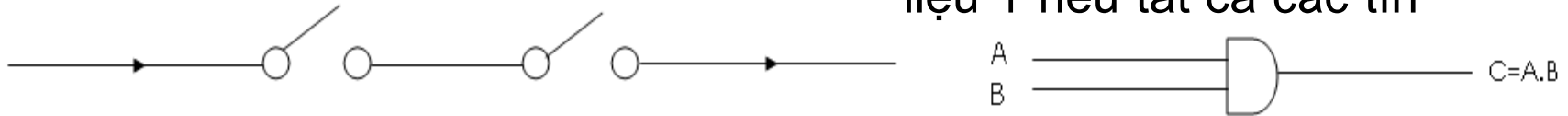
- Cổng AND
- Cổng OR
- Cổng NOT
- Cổng NAND
- Cổng NOR



CÁC CỒNG LUẬN LÍ - Cổng AND

- Cổng AND là sự thực hiện vật lí của phép toán nhân luận lí (AND).

○ Là một mạch điện tử có đầu ra là tín hiệu 1 nếu tất cả các tín



Hình 6.18. Hai hay nhiều công tắc mắc nối tiếp như cổng AND.

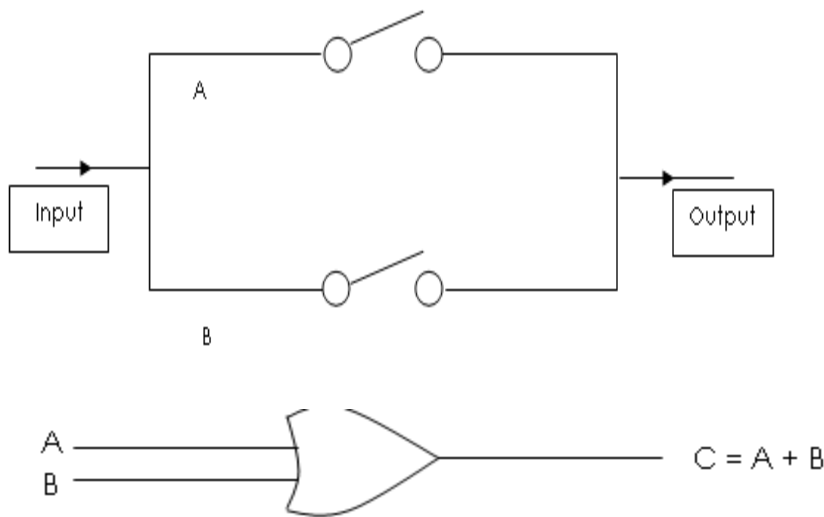
- Hoạt động: các trạng thái của tín hiệu đầu ra phụ thuộc vào sự kết hợp khác nhau của các tín hiệu đầu vào, được mô tả bằng bảng chân trị.

Bảng chân trị của cổng AND

| Inputs | | Output |
|--------|---|-----------------|
| A | B | $C = A \cdot B$ |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

CÁC CỘNG LUẬN LÍ - Cổng OR

- Cổng OR là sự thực hiện vật lí của phép toán cộng luận lí (OR).
- Là một mạch điện tử có tín hiệu đầu ra là 0 nếu tất cả các tín hiệu đầu vào là 0.

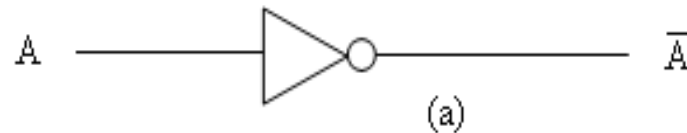


Bảng chân trị của cổng OR

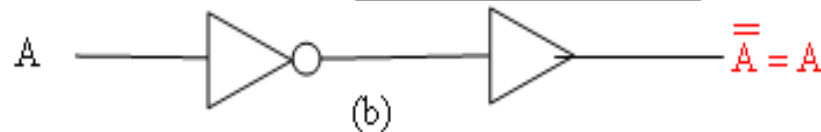
| Input | | Output |
|-------|---|-------------|
| A | B | $C = A + B$ |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

CÁC CÔNG LUẬN LÝ - Cổng NOT

- Cổng NOT là sự thực hiện vật lí của phép bù.
- Là một mạch điện tử có tín hiệu đầu ra là phần đảo của tín hiệu đầu vào.



| Input | Output |
|-------|-----------|
| A | \bar{A} |
| 0 | 1 |
| 1 | 0 |



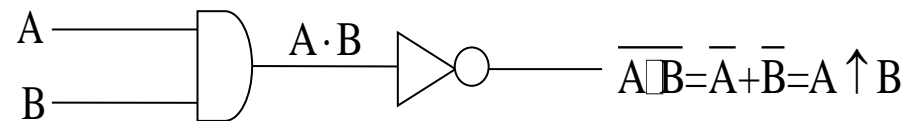
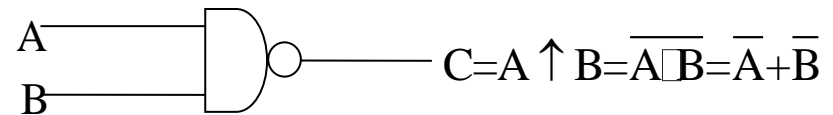
Hình 6.23. Biểu tượng của sơ đồ khối và bảng chân trị cho
(a) Một cổng Not và (b) Hai cổng Not

CÁC CÔNG LUẬN LÝ - Cổng NAND

- Cổng NAND là một phần bù của cổng AND.
- Cổng ra của NAND sẽ là 0 khi tất cả cổng vào là 1.
- Ký hiệu: $AB = A \cdot B = A + B = A \cdot B$

| Inputs | | Output |
|--------|---|---------------------------------------|
| A | B | $C = \overline{A} \cdot \overline{B}$ |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Bảng chân trị của cổng NAND



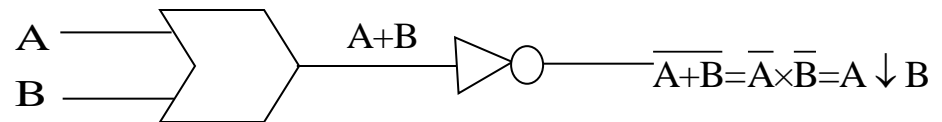
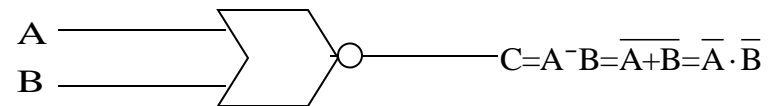
Cổng NAND được tạo từ cổng AND và cổng NOT

CÁC CÔNG LUẬN LÝ - Cổng NOR

- Cổng NOR là một phần bù của cổng OR.
- Cổng ra của cổng NOR sẽ là 1 khi và chỉ khi tất cả các cổng vào là 0.
- Ký hiệu: $A \downarrow B = \overline{A+B} = \overline{A} + \overline{B}$

| Inputs | | Output |
|--------|---|----------------------|
| A | B | $C = \overline{A+B}$ |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

Bảng chân trị của cổng NAND

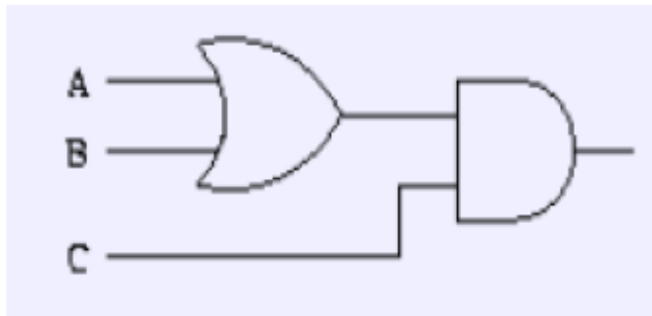


Cổng NAND được tạo từ cổng OR và cổng NOT

MẠCH LOGIC

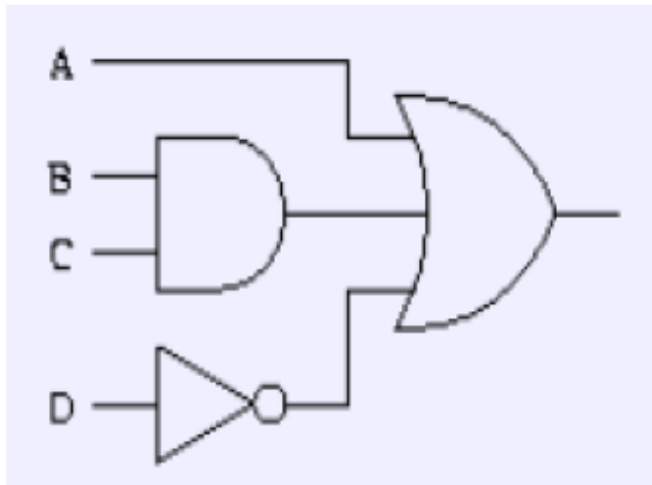
- Mạch Logic là sự kết hợp của các mạch And, Or, Nand, Nor,...

○ V
đi



$$(A + B).C$$

hoặc

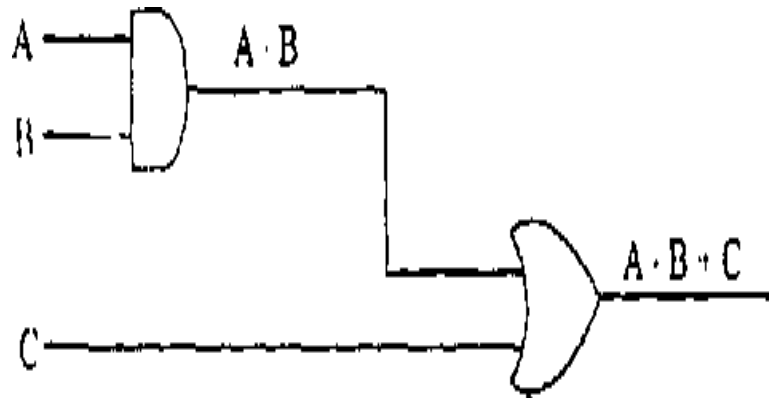


$$A + BC + \bar{D}$$

CHUYỂN ĐỔI BIỂU THỨC THÀNH MẠCH LOGIC

- **Ví dụ:** Xây dựng một mạch logic cho biểu thức luận lý.
 $A \cdot B + C$

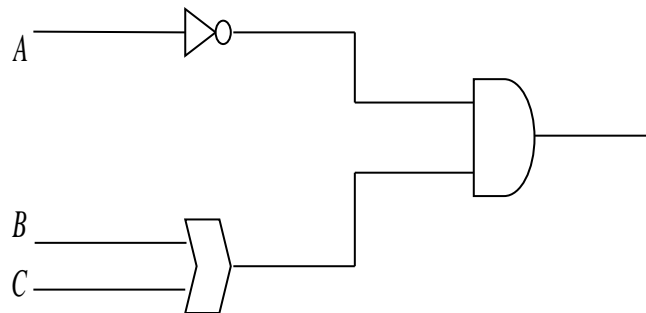
- *Giải*



MẠCH LOGIC

Cách giải

- Đầu vào A được chuyển vào cổng NOT và đầu ra của nó sẽ là \bar{A}
- Đầu vào B và C được chuyển vào cổng OR và đầu ra của nó sẽ là $B + C$
- Hai cổng ra (\bar{A} và $B + C$) được chuyển vào cổng AND. Đầu ra của cổng AND sẽ là $\bar{A} \cdot (B + C)$
- $D = \bar{A} \cdot (B + C)$, là biểu thức luận lý đầu ra của mạch logic.

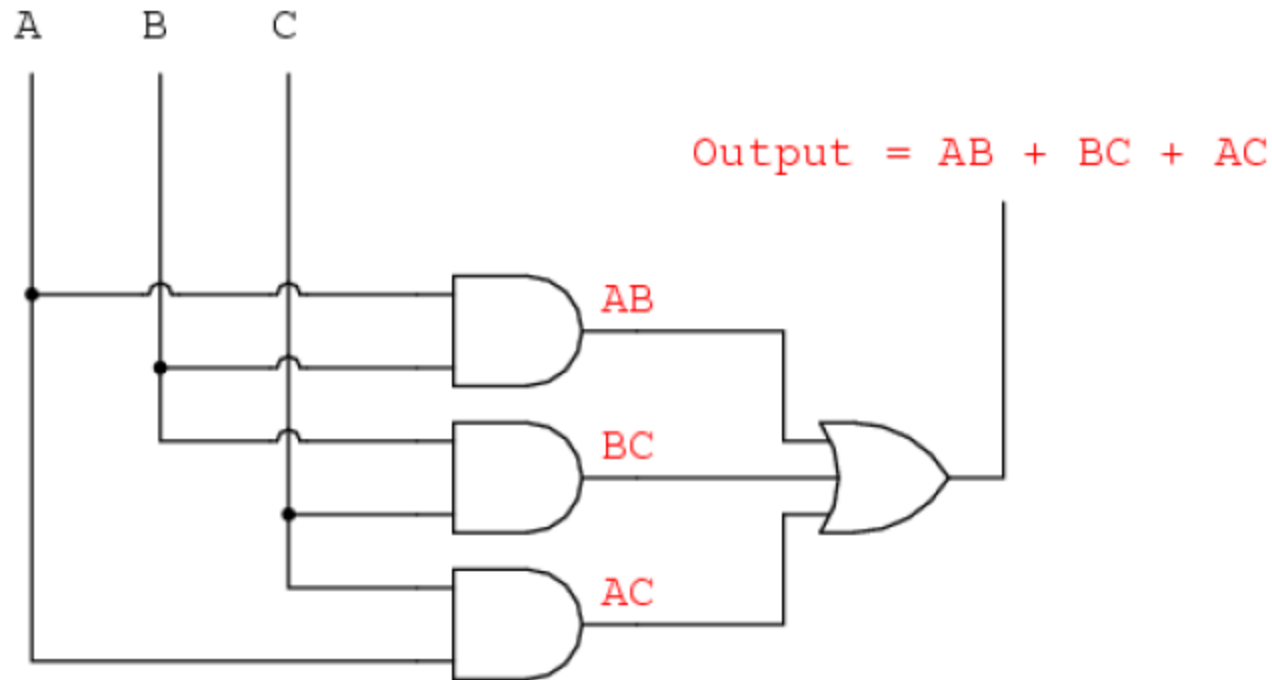


CHUYỂN ĐỔI BIỂU THỨC THÀNH MẠCH LOGIC

- Ví dụ: Xây dựng một mạch logic cho biểu thức luận lý.

$$AB + BC + AC$$

- Giải

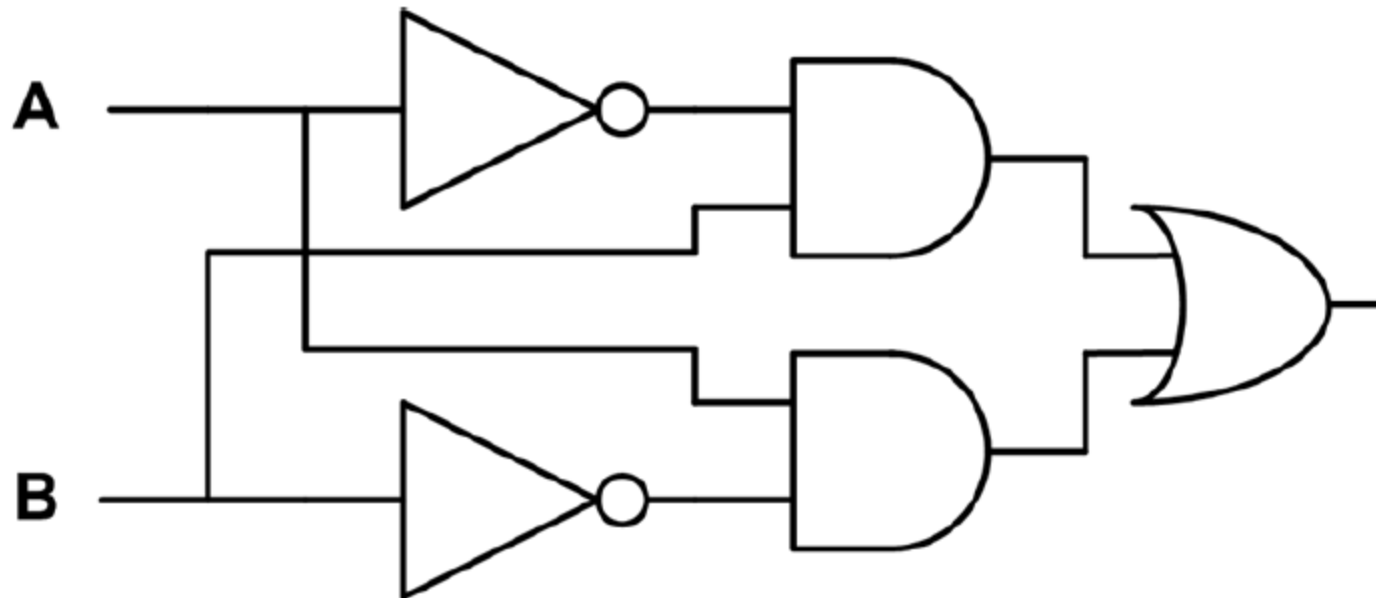


CHUYỂN ĐỔI BIỂU THỨC THÀNH MẠCH LOGIC

- Ví dụ: Xây dựng một mạch logic cho biểu thức luận lý.

$$A.B + A.B$$

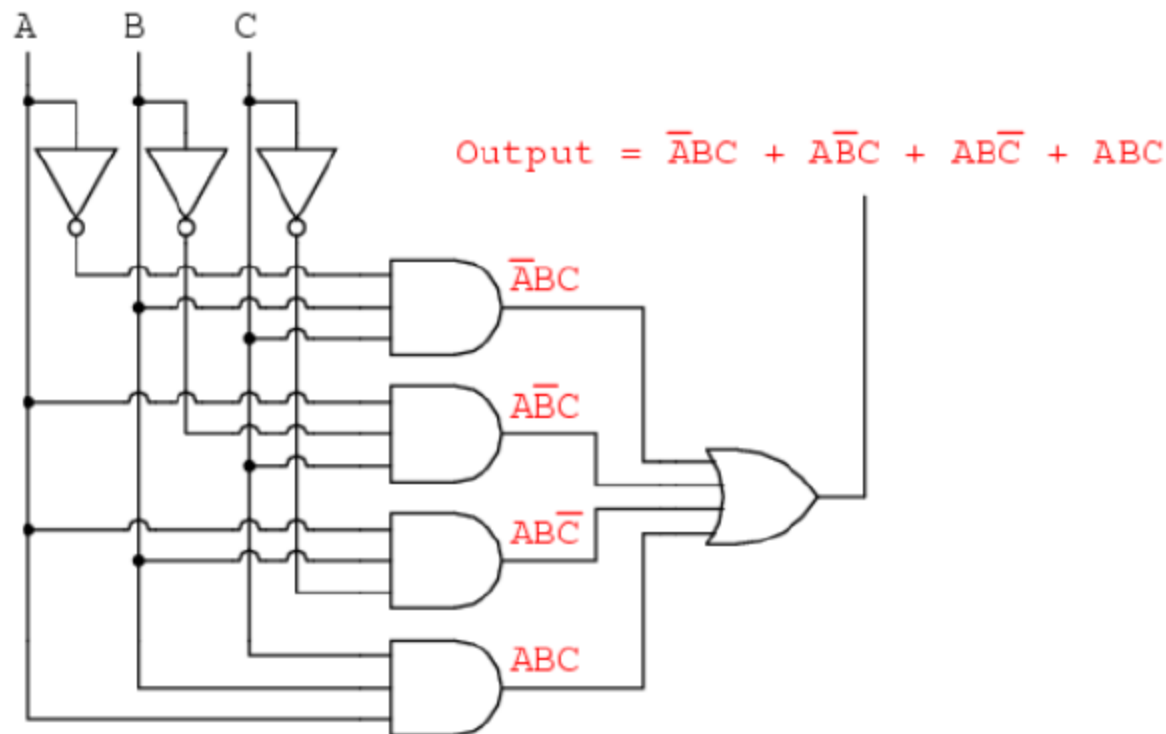
- $C' \cdot D$



CHUYỂN ĐỔI BIỂU THỨC THÀNH MẠCH LOGIC

- **Ví dụ:** Xây dựng một mạch logic cho biểu thức luận lý.
$$\overline{A}.B.C + A.\overline{B}.C + A.B.\overline{C} + ABC$$

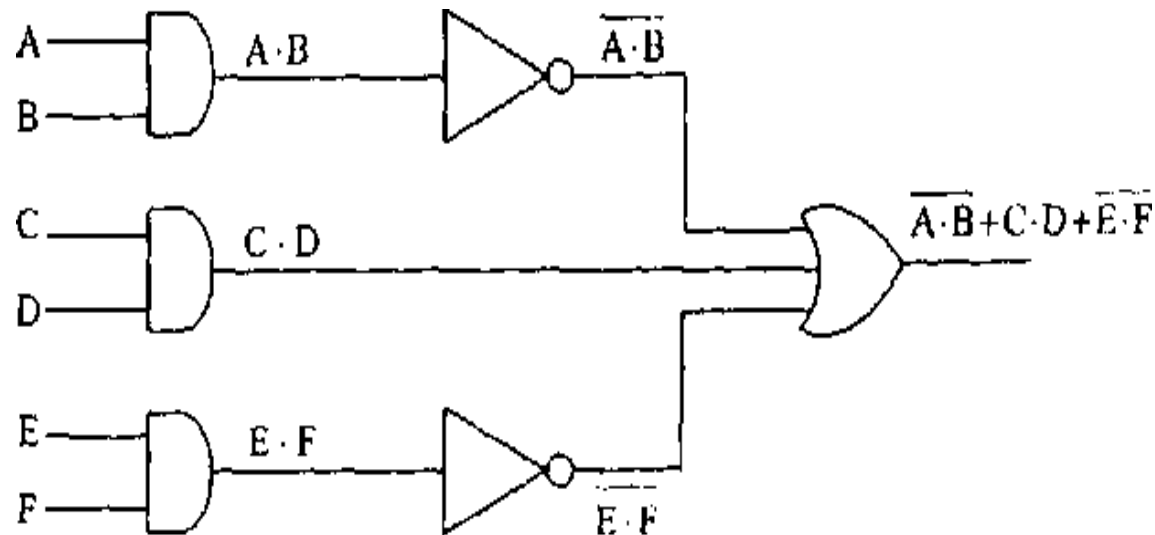
- **Giải**



CHUYỂN ĐỔI BIỂU THỨC THÀNH MẠCH LOGIC

- Ví dụ: Xây dựng một mạch logic cho biểu thức luận lý.
$$\overline{A \cdot B} + C \cdot D + \overline{E \cdot F}$$

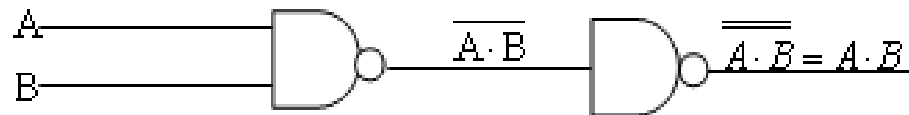
- Giải



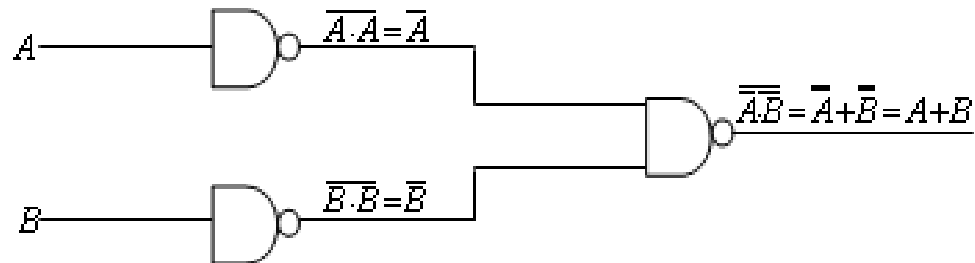
CỔNG NAND CHUNG

- Là sự hoạt động hợp lý của AND, OR và NOT có thể thực hiện với NAND

(a) Sự thực hiện đầy đủ của cổng NOT



(b) Sự thực hiện đầy đủ của cổng AND



(c) Sự thực hiện đầy đủ của cổng OR



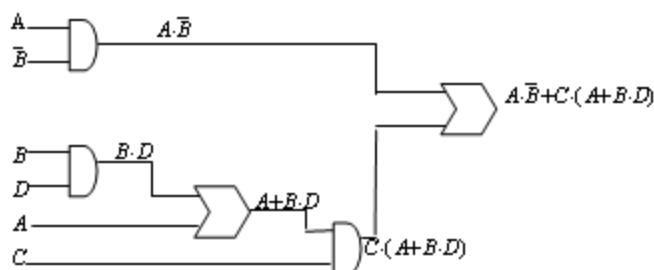
CỔNG NAND CHUNG

Phương pháp xây dựng cổng NAND chung

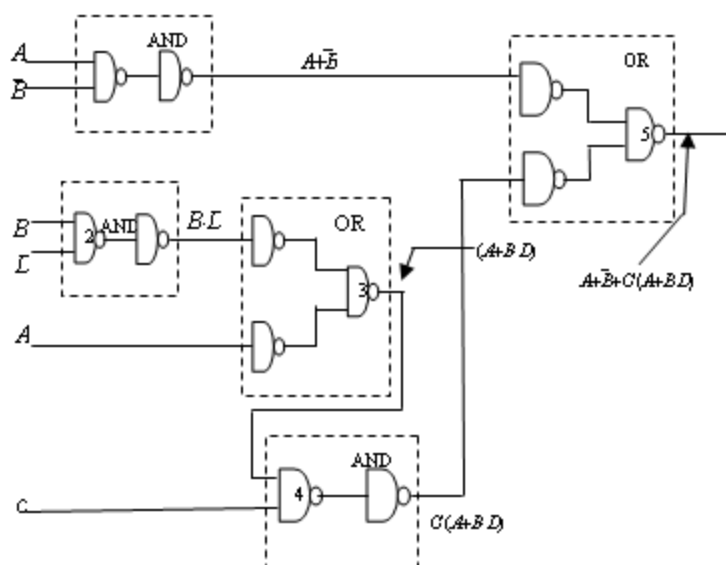
- **Bước 1:** Xuất phát từ biểu thức đại số đã cho, vẽ sơ đồ logic với các cổng AND, OR và NOT. Giả sử cả đường vào của (A) và phần bù của (A) là có sẵn.
- **Bước 2:** Vẽ một sơ đồ logic thứ hai với cổng logic NAND thay thế tương ứng cho mỗi cổng AND, OR, và NOT.
- **Bước 3:** Xóa hai đường đảo chiều từ sơ đồ (là các đường có 1 ngõ vào). Xóa cả đường đảo chiều nối đến đường vào bên ngoài và thêm biến số đường vào tương ứng.

CỔNG NAND CHUNG

Ví dụ: Xây dựng một mạch logic cho biểu thức luận lý sau chỉ dùng cổng NAND.

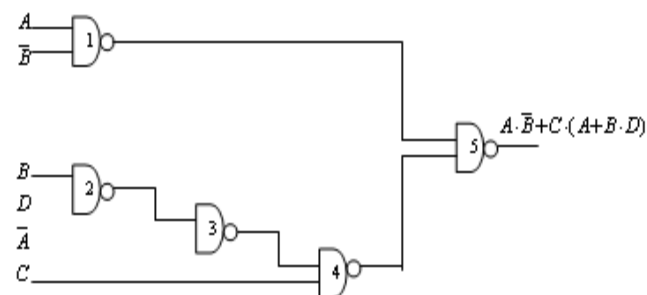


(a) Sự thực hiện đầy đủ của AND/OR



(b) Hàm NAND thay thế tương đương

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



(c) Việc thực hiện của cổng NAND

CỔNG NOR CHUNG

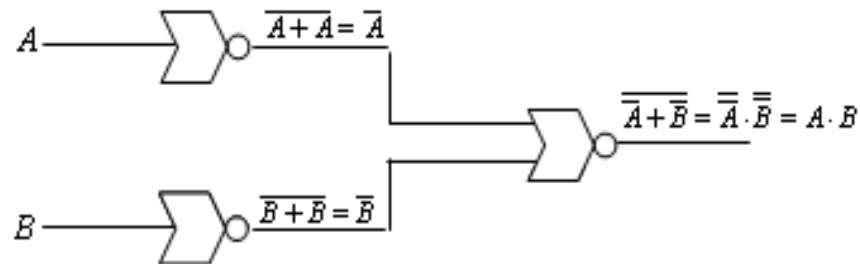
- Là sự hoạt động hợp lý của AND, OR và NOT có thể thực hiện với NAND



(a) Sự thực hiện của cổng NOT



(b) Sự thực hiện của cổng OR



(c) Sự thực hiện của AND

CỔNG NOR CHUNG

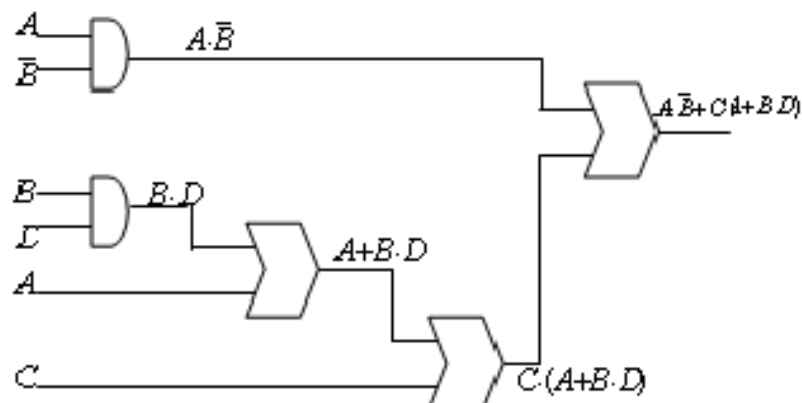
Phương pháp xây dựng cổng NOR chung

- **Bước 1:** Với biểu thức đại số đã cho, vẽ sơ đồ logic với cổng AND, OR và NOT. Biết rằng cả đầu vào biểu thức (A) và phần bù (A) đều có sẵn
- **Bước 2:** Vẽ một sơ đồ logic thứ hai tương đương với cổng NOR thay thế cho mỗi cổng AND, OR và NOT.
- **Bước 3:** Xóa 2 đường đảo chiều. Xóa cả những đường đảo chiều nối đến đầu vào bên ngoài cổng đơn và thêm biến số đầu vào thích hợp.

CỔNG NOR CHUNG

Ví dụ: Xây dựng một mạch logic cho biểu thức luận lý sau chỉ dùng cổng NOR.

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

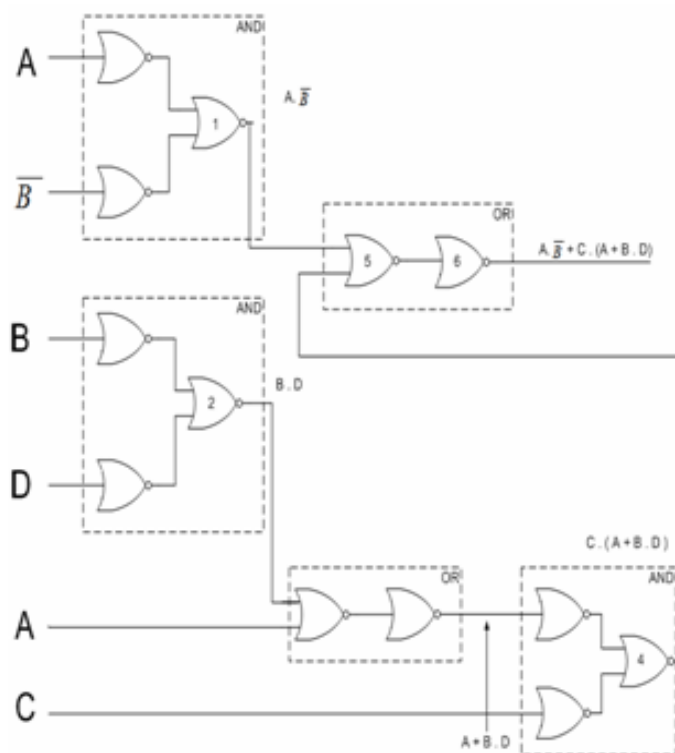


(a) Việc thực hiện của AND/OR

CỔNG NOR CHUNG

Ví dụ: Xây dựng một mạch logic cho biểu thức luận lý sau chỉ dùng cổng NOR.

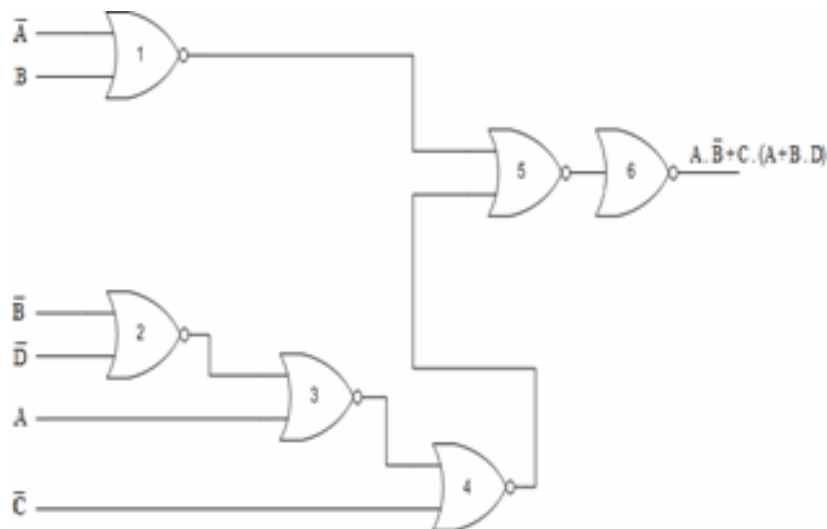
$$A \cdot \bar{B} + C \cdot (A + B \cdot D)$$



(b) Thay thế tương đương chức năng NOR

CỔNG NOR CHUNG

Ví dụ: Xây dựng một mạch logic cho biểu thức luận lý sau chỉ dùng cổng



(c) Sự thi hành NOR

Cổng NAND và NOR cao cấp hơn các cổng AND và OR từ phần cứng, vì chúng cung cấp đầu ra duy trì giá trị tín hiệu mà không làm mất độ lớn.

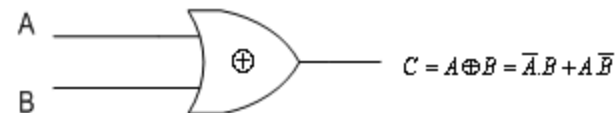
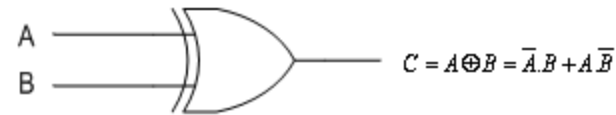
Cổng OR và AND thỉnh thoảng cần phục hồi độ lớn sau khi tín hiệu đi qua vài cấp độ.

PHÉP TOÁN LOẠI TRỪ VÀ HÀM TƯƠNG ĐƯƠNG

- Phép toán loại trừ OR (Exclusive-OR) : Ký hiệu \oplus
- Phép tương đương (Equivalence): Ký hiệu \odot
- Là các phép toán nhị phân thực hiện theo những hàm Boolean sau:

$$A \oplus B = A \cdot \bar{B} + \bar{A} \cdot B$$

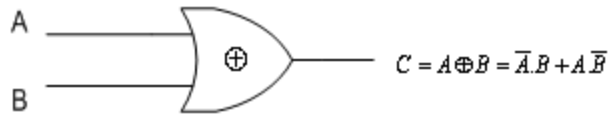
$$A \odot B = A \cdot B + \bar{A} \cdot \bar{B}$$



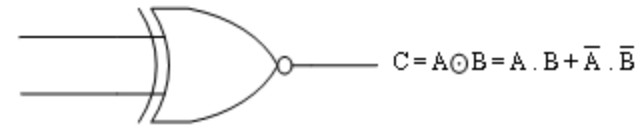
(a) Ký hiệu sơ đồ khối cho phép toán loại trừ OR

| Inputs | | Outputs |
|--------|---|------------------|
| A | B | $C = A \oplus B$ |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

PHÉP TOÁN LOẠI TRỪ VÀ HÀM TƯƠNG ĐƯƠNG



| Inputs | | Outputs |
|--------|---|------------------|
| A | B | $C = A \oplus B$ |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |



| Inputs | | Outputs |
|--------|---|-----------------|
| A | B | $C = A \odot B$ |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

- Phép toán loại trừ OR và phép toán tương đương là những ứng cử viên cho những cổng logic nhưng quá mắc để xây dựng các thành phần vật lý trong máy tính.
- Chúng có sẵn như những cổng logic chuẩn tại gói IC nhưng thường được xây dựng bên trong với những cổng tiêu chuẩn khác.

THIẾT KẾ CỦA MẠCH KẾT HỢP

Các bước thiết kế mạch kết hợp:

- Phát biểu bài toán đã cho hoàn toàn chính xác.
- Giải thích vấn đề và xác định những biến số vào là có sẵn và những biến đầu ra được yêu cầu.
- Gán một ký hiệu bằng chữ tới mỗi biến đầu vào và mỗi biến đầu ra.
- Thiết kế bảng chân trị định nghĩa những quan hệ được yêu cầu giữa đầu vào và đầu ra.
- Hàm Boolean được đơn giản hóa cho mỗi đầu ra.
- Vẽ sơ đồ mạch logic để thực hiện hàm Boolean

THIẾT KẾ MẠCH CỘNG BÁN PHẦN

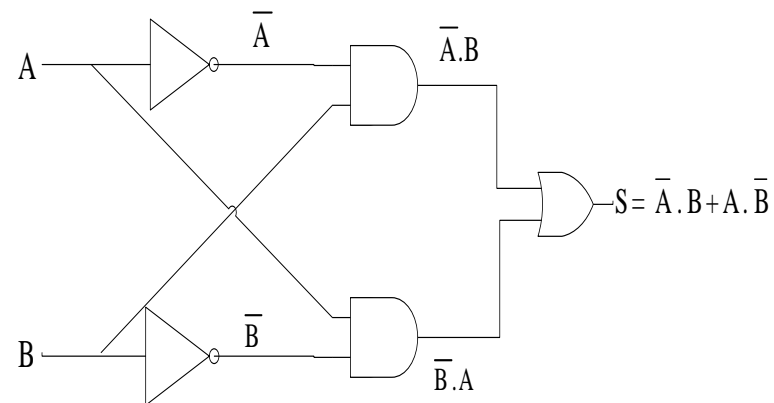
- Giả sử A và B là hai biến đầu vào, S (tổng) và C (số nhớ).
- Ta có bảng chân trị của một mạch cộng bán phần.

| Inputs | | Outputs | |
|--------|---|---------|---|
| A | B | C | S |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Từ bảng chân trị có

$$S = \bar{A} \cdot B + A \cdot \bar{B}$$

$$C = A \cdot B$$



Sơ đồ mạch logic của mạch cộng bán phần.

THIẾT KẾ MẠCH CỘNG TOÀN PHẦN

- Giả sử A và B là hai biến đầu vào, S (tổng) và C (số nhớ), biến số vào thứ ba (D) đại diện cho số nhớ
- Ta có bảng chân trị của một mạch cộng toàn phần.

$$S = \bar{A} \cdot \bar{B} \cdot D + \bar{A} \cdot B \cdot \bar{D} + A \cdot \bar{B} \cdot \bar{D} + A \cdot B \cdot D$$

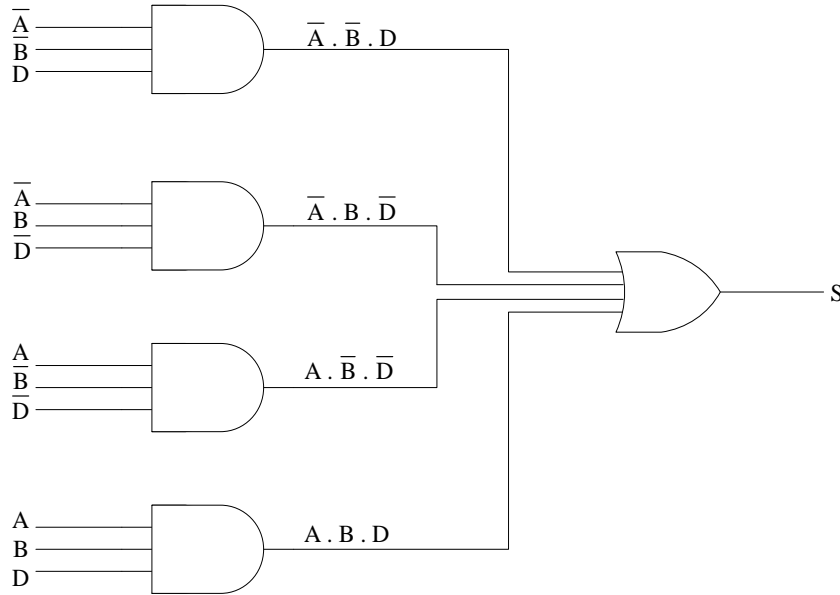
$$C = A \cdot B + A \cdot D + B \cdot D$$

| Inputs | | | Outputs | |
|--------|---|---|---------|---|
| A | B | D | C | S |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

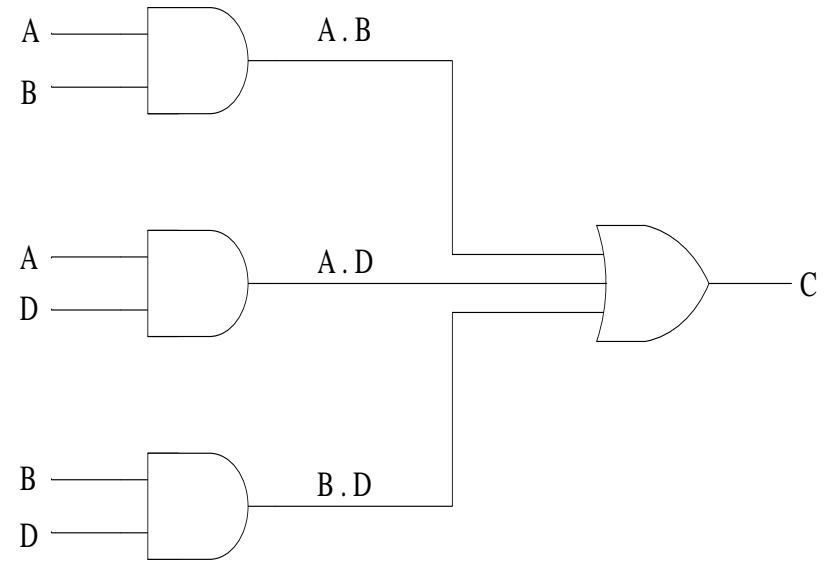
$$S = \bar{A} \cdot \bar{B} \cdot D + \bar{A} \cdot B \cdot \bar{D} + A \cdot \bar{B} \cdot \bar{D} + A \cdot B \cdot D$$

$$C = A \cdot B + A \cdot D + B \cdot D$$

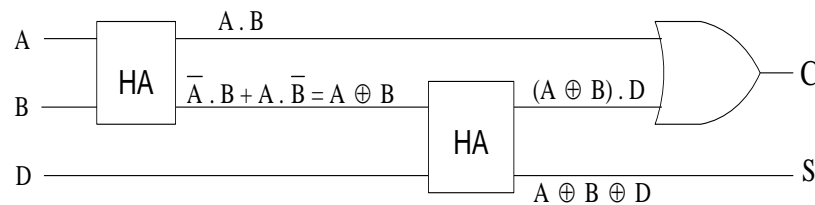
THIẾT KẾ MẠCH CỘNG TOÀN PHẦN



Sơ đồ mạch Logic cho Sum



Sơ đồ mạch logic cho số nhớ



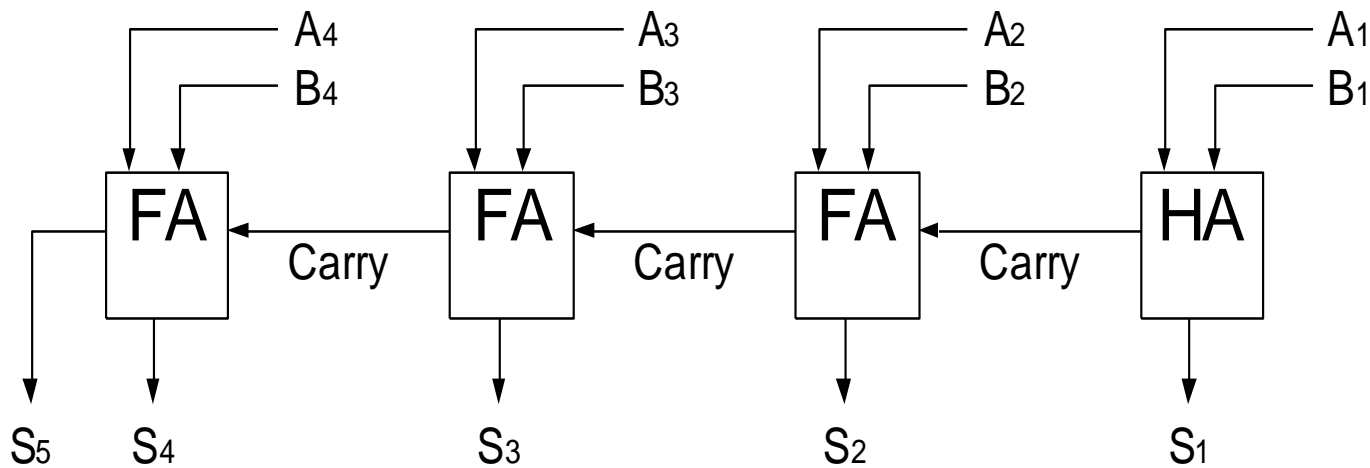
**Sự thi hành của mạch cộng toàn phần
với hai mạch cộng bán phần và một cổng OR**

THIẾT KẾ MẠCH CỘNG NHỊ PHÂN SONG SONG

- Mạch cộng nhị phân song song được dùng để thêm hai số nhị phân.
- Nếu chúng ta muốn thêm hai số bốn bit, chúng ta cần xây dựng một mạch cộng nhị phân bốn bit song song.
- Một mạch cộng như vậy yêu cầu mạch cộng bán phần (được biểu thị bởi HA) và ba mạch cộng toàn phần (được biểu thị bởi FA). Những số nhị phân được bổ sung là $A_4 A_3 A_2 A_1$ và $B_4 B_3 B_2 B_1$, và kết quả là:

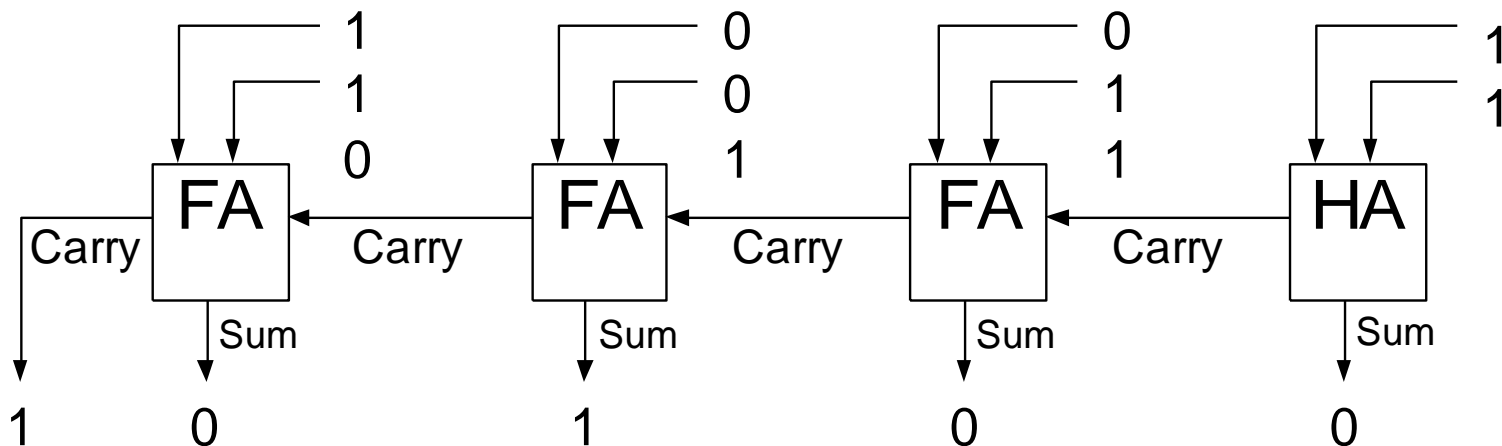
THIẾT KẾ MẠCH CỘNG NHỊ PHÂN SONG SONG

$$\begin{array}{r} A_4 A_3 A_2 A_1 \\ + B_4 B_3 B_2 B_1 \\ \hline S_5 S_4 S_3 S_2 S_1 \end{array}$$



THIẾT KẾ MẠCH CỘNG NHỊ PHÂN SONG SONG

Thêm hai số 9 và 11 thêm vào, số nhị phân tương đương của số thập phân 9 là 1001, và số thập phân 11 là 1011



Những số nhị phân lớn nhất có thể được thêm vào việc sử dụng mạch cộng toàn phần là 1111 và 1111

CÂU HỎI VÀ BÀI TẬP

1. Giải thích nguyên lý đối ngẫu trong đại số Boolean. Nó hữu ích như thế nào?
2. Các cổng AND, OR và NOT là những hoàn thành luận lý, hãy thảo luận về vấn đề đó.
3. Tại sao cổng NAND và NOR gọi là cổng chung?
4. Trình bày sự thực hiện của các phép toán logic AND, OR và NOT chỉ với cổng NAND và chỉ với cổng NOR.
5. Xây dựng biểu đồ mạch logic cho “half- adder” sử dụng duy nhất cổng NAND
6. Xây dựng biểu đồ mạch logic cho “half- adder” sử dụng duy nhất cổng NOR
7. Tại sao các mạch tổ hợp hay được xây dựng thường xuyên với cổng NAND và NOR hơn là cổng AND, Or, NOT?
8. Mạch logic có 3 đầu vào là A,B,C. Nó tạo 1 đầu ra duy nhất khi $A=0, B=1, C=0$, Xây dựng mạch tổ hợp cho hệ thống này. `