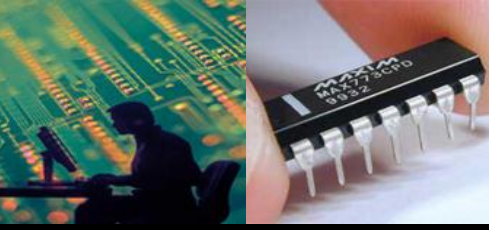


ĐIỆN TỬ SỐ - CHƯƠNG 2

HÀM LOGIC VÀ CỔNG LOGIC

Bộ môn Kỹ thuật vi xử lý
Khoa Vô tuyến điện tử
Học viện Kỹ thuật quân sự



NỘI DUNG CHƯƠNG 2

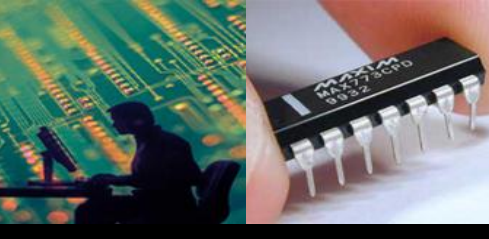
1. Cơ bản về đại số logic

2. Giới thiệu về cổng logic

3. Biểu diễn hàm logic

4. Tối thiểu hàm logic

5. Hệ hàm logic



1.1. Các hàm logic cơ bản

□ Biến logic

- Xét tập B chỉ chứa hai phần tử 0 và 1: $B = \{0, 1\}$
- x_i được gọi là biến logic nếu x_i là biến của một hàm và $x_i \in B$

□ Hàm logic

- Hàm f được gọi là hàm logic nếu nó là hàm của các biến logic
- Hàm f chỉ nhận một trong hai giá trị 0 hoặc $1 \in B$

□ Ví dụ: hàm 3 biến $y = f(x_1, x_2, x_3)$

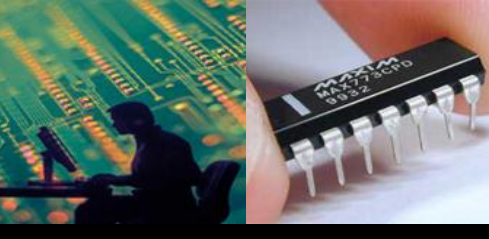
- Các tổ hợp biến: 000, 001, 010, 011, 100, 101, 110, 111

- Cho theo dạng cơ số 2 $y = f^1(010, 011, 101, 110)$

$$y = \sum(010, 011, 101, 110)$$

- Cho theo dạng cơ số 10 $y = f^1(2, 3, 5, 6)$

$$y = \sum(2, 3, 5, 6)$$



1.1. Các hàm logic cơ bản

□ Hàm VÀ (AND) hay còn gọi phép hội (nhân)

- $Y = x_1 \cdot x_2 \dots \cdot x_n = x_1 \wedge x_2 \wedge \dots \wedge x_n$
- Chỉ nhận giá trị logic 1 khi tất cả các biến đều bằng 1

□ Hàm HOẶC (OR) hay còn gọi phép tuyển (tổng)

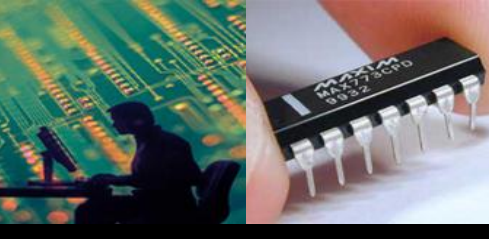
- $Y = x_1 + x_2 \dots + x_n = x_1 \vee x_2 \vee \dots \vee x_n$
- Chỉ nhận giá trị logic 0 khi tất cả các biến đều bằng 0

AND

x_1	x_2	Y
0	0	0
0	1	0
1	0	0
1	1	1

OR

x_1	x_2	Y
0	0	0
0	1	1
1	0	1
1	1	1



1.1. Các hàm logic cơ bản

❑ **Hàm ĐẢO (NOT):** có giá trị bằng nghịch đảo giá trị của biến $y = \overline{x}$

❑ **Hàm VÀ ĐẢO (NOT AND - NAND)**

➤ $Y = \overline{x_1 \cdot x_2 \dots \cdot x_n} = \overline{x_1 \wedge x_2 \wedge \dots \wedge x_n}$

➤ Giá trị bằng giá trị đảo của hàm VÀ (AND)

❑ **Hàm HOẶC ĐẢO (NOT OR - NOR)**

➤ $Y = \overline{x_1 + x_2 \dots + x_n} = \overline{x_1 \vee x_2 \vee \dots \vee x_n}$

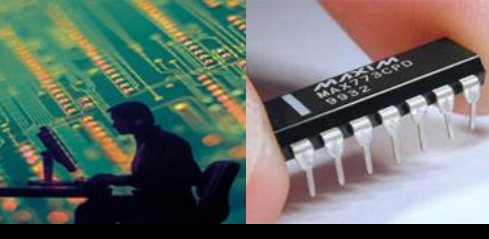
➤ Giá trị bằng giá trị đảo của hàm HOẶC (OR)

NAND

x_1	x_2	Y
0	0	1
0	1	1
1	0	1
1	1	0

NOR

x_1	x_2	Y
0	0	1
0	1	0
1	0	0
1	1	0



1.1. Các hàm logic cơ bản

□ Hàm HOẶC LOẠI TRỪ (EXCLUSIVE OR - XOR)

➤ $Y = x_1 \oplus x_2 \oplus \dots \oplus x_n$

➤ Giá trị bằng 1 khi có số lẻ các biến nhận giá trị 1

□ Hàm CỘNG TƯƠNG ĐƯƠNG (NOT XOR)

➤ $Y = \overline{x_1 \oplus x_2 \oplus \dots \oplus x_n} = x_1 \sim x_2 \sim \dots \sim x_n$

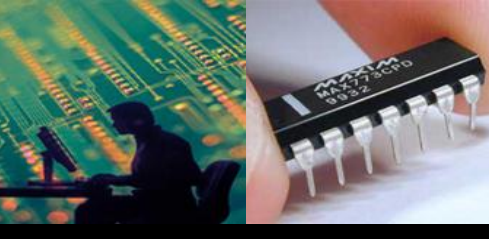
➤ Giá trị bằng 1 khi có số chẵn các biến nhận giá trị 1

XOR

x_1	x_2	Y
0	0	0
0	1	1
1	0	1
1	1	0

**NOT
XOR**

x_1	x_2	Y
0	0	1
0	1	0
1	0	0
1	1	1



1.2. Các tính chất của đại số logic

□ Tính chất giao hoán

$$A \cdot B = B \cdot A; \quad A + B = B + A; \quad A \oplus B = B \oplus A$$

□ Tính chất kết hợp

$$A \cdot B \cdot C = (A \cdot B) \cdot C = A \cdot (B \cdot C)$$

$$A + B + C = (A + B) + C = A + (B + C)$$

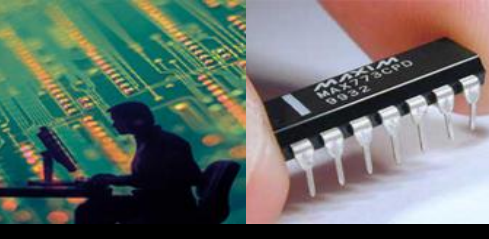
$$A \oplus B \oplus C = (A \oplus B) \oplus C = A \oplus (B \oplus C)$$

□ Tính chất phân phối

$$(A + B) \cdot C = A \cdot C + B \cdot C$$

$$(A \cdot B) + C = (A + C) \cdot (B + C)$$

$$A \cdot (B \oplus C) = A \cdot B \oplus A \cdot C$$



1.2. Các tính chất của đại số logic

□ Tính không số mũ, không thừa số

$$A \cdot A \cdot \dots \cdot A = A; \quad A + A + \dots + A = A$$

□ Tính chất phép đảo

$$\overline{\overline{A}} = A; \quad A \cdot \overline{A} = 0; \quad A + \overline{A} = 1$$

□ Tính chất phép toán với 0 và 1

$$A \cdot 0 = 0; \quad A \cdot 1 = A; \quad A + 0 = A; \quad A + 1 = 1$$

□ Định luật De-Morgan

➤ Đảo của một tổng bằng tích các đảo thành phần

$$\overline{A_1 + A_2 + \dots + A_n} = \overline{A_1} \cdot \overline{A_2} \cdot \dots \cdot \overline{A_n}$$

➤ Đảo của một tích bằng tổng các đảo thành phần

$$\overline{A_1 \cdot A_2 \cdot \dots \cdot A_n} = \overline{A_1} + \overline{A_2} + \dots + \overline{A_n}$$



1.2. Các tính chất của đại số logic

□ Quy tắc thay thế

Trong một đẳng thức bất kỳ, nếu thay thế một biến nào đó bằng một hàm thì đẳng thức vẫn đúng

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

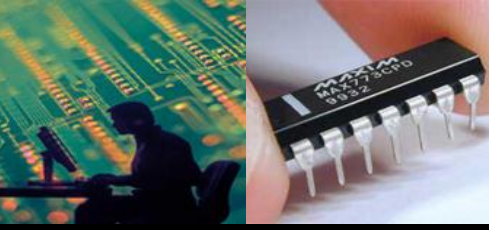
□ Quy tắc đối ngẫu

Một đẳng thức logic sẽ vẫn còn đúng nếu thay thế tương ứng ở cả hai vế phép cộng bằng phép nhân và ngược lại, số 1 bằng số 0 và ngược lại. Hàm thay thế như vậy gọi là hàm đối ngẫu

$$(A + B) \cdot C = A \cdot C + B \cdot C \leftrightarrow (A \cdot B) + C = (A + C) \cdot (B + C)$$

□ Các tính chất của hàm XOR

$$\begin{aligned} A \oplus A &= 0; \quad \underline{A} \oplus \overline{A} = 1; \quad \underline{A} \oplus 0 = A; \quad \underline{A} \oplus 1 = \overline{A} \\ A \oplus B &= A \oplus B = A \oplus B; \quad A \oplus B = \overline{A} \oplus \overline{B} \\ A \oplus B &= C \leftrightarrow A \oplus C = B \leftrightarrow B \oplus C = A \end{aligned}$$



NỘI DUNG CHƯƠNG 2

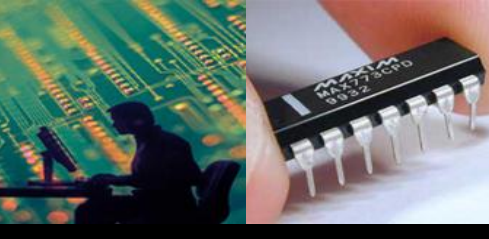
1. Cơ bản về đại số logic

2. Giới thiệu về cổng logic

3. Biểu diễn hàm logic

4. Tối thiểu hàm logic

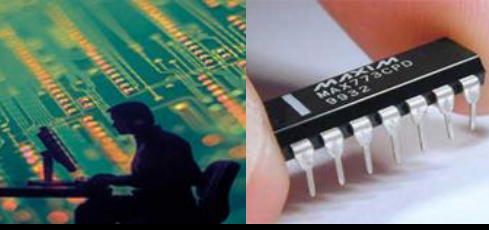
5. Hệ hàm logic



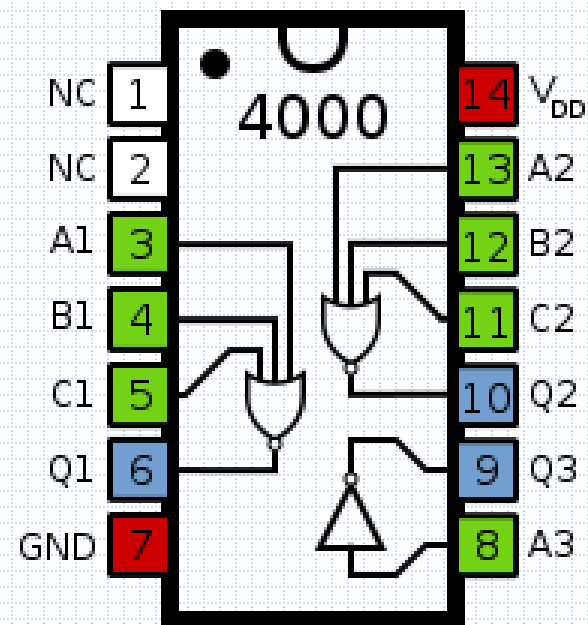
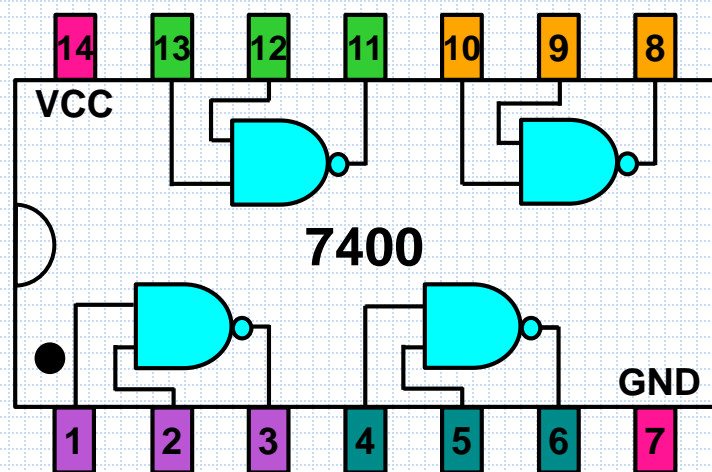
2. Giới thiệu về cổng logic

□ Khái niệm cổng logic

- Cổng logic là mạch điện được thiết kế trên cơ sở các linh kiện bán dẫn như Diode, BJT, CMOS để thực hiện chức năng 1 hàm logic
- Các tín hiệu vào ra của cổng logic chỉ có 2 mức: mức thấp (Low – L) và mức cao (High – H) tương ứng với 2 giá trị logic 0 và 1
- Các cổng logic trên thực tế thường được chế tạo dưới dạng các vi mạch có kích thước rất nhỏ, tích hợp một vài cổng logic cùng lúc, được đóng vỏ nhựa và nổi chân kim loại ra gọi là IC cổng logic.
- Các họ IC cổng logic phổ biến nhất là 7400 sử dụng công nghệ TTL (Transistor Transistor Logic) và 4000 sử dụng công nghệ CMOS (Complementary Metal Oxide Semiconductor). Số lượng tích hợp phổ biến trên mỗi IC thường là 2, 3, 4 hoặc 6 cổng



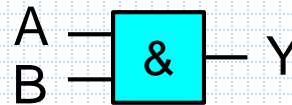
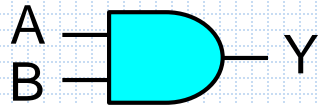
2. Giới thiệu về cổng logic



2. Giới thiệu về cổng logic

❑ Cổng VÀ (AND)

➤ Ký hiệu



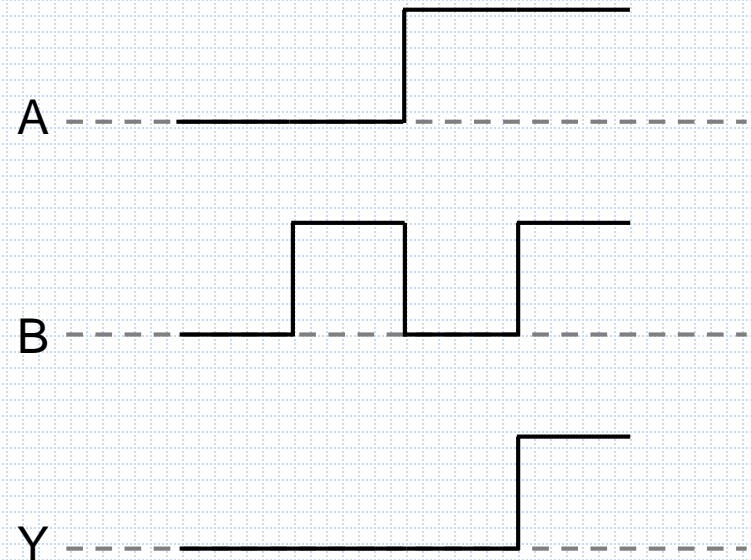
➤ Bảng chân lý

AND

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

➤ Biểu thức $Y = A \cdot B$

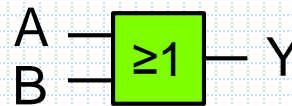
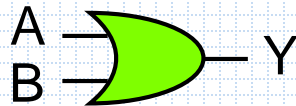
➤ Giải đồ thời gian



2. Giới thiệu về cổng logic

❑ Cổng HOẶC (OR)

➤ Ký hiệu



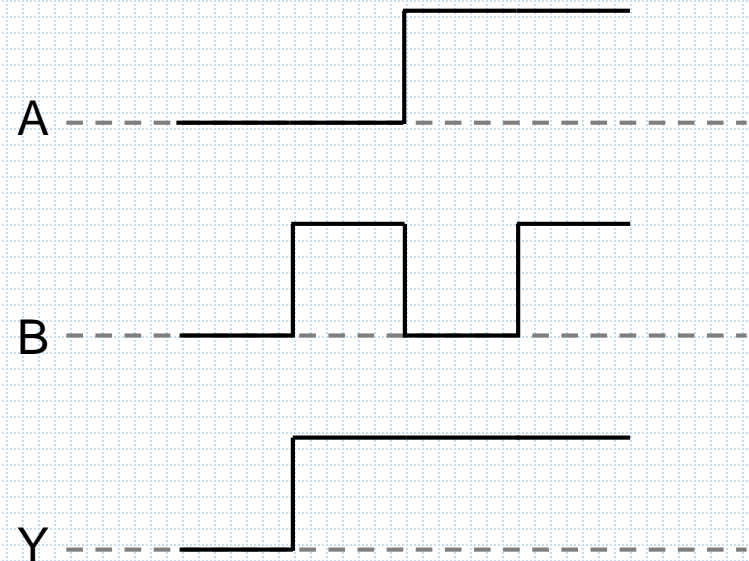
➤ Bảng chân lý

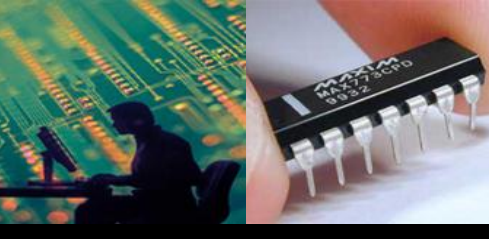
OR

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

➤ Biểu thức $Y = A + B$

➤ Giải đồ thời gian

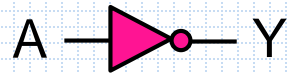




2. Giới thiệu về cổng logic

❑ Cổng ĐẢO (NOT)

➤ Ký hiệu

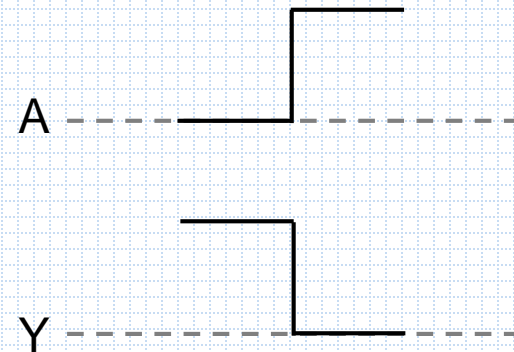


➤ Bảng chân lý

NOT

A	Y
0	1
1	0

➤ Giải đồ thời gian

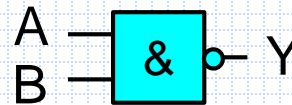
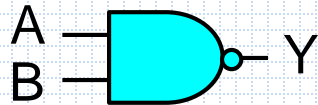


➤ Biểu thức $Y = \overline{A}$

2. Giới thiệu về cổng logic

❑ Cổng VÀ ĐẢO (NOT AND - NAND)

➤ Ký hiệu



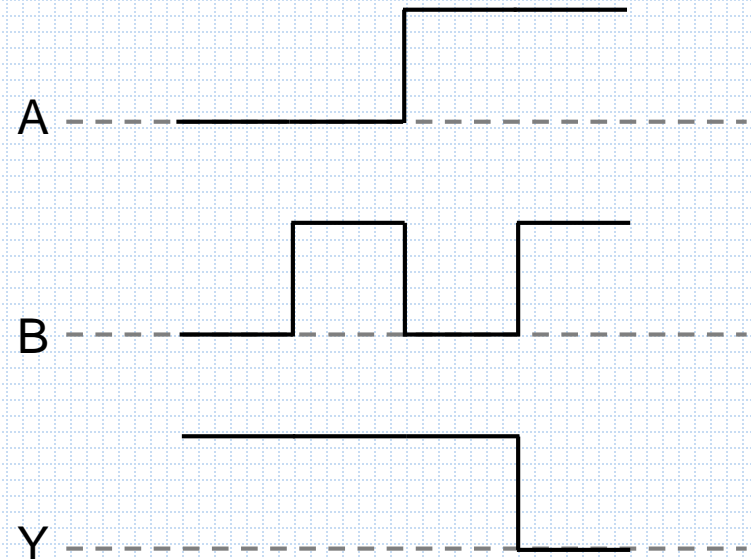
➤ Bảng chân lý

NAND

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

➤ Biểu thức $Y = \overline{A \cdot B}$

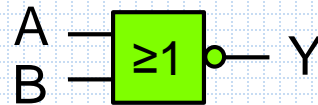
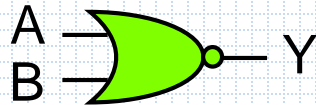
➤ Giải đồ thời gian



2. Giới thiệu về cổng logic

❑ Cổng HOẶC ĐẢO (NOR)

➤ Ký hiệu



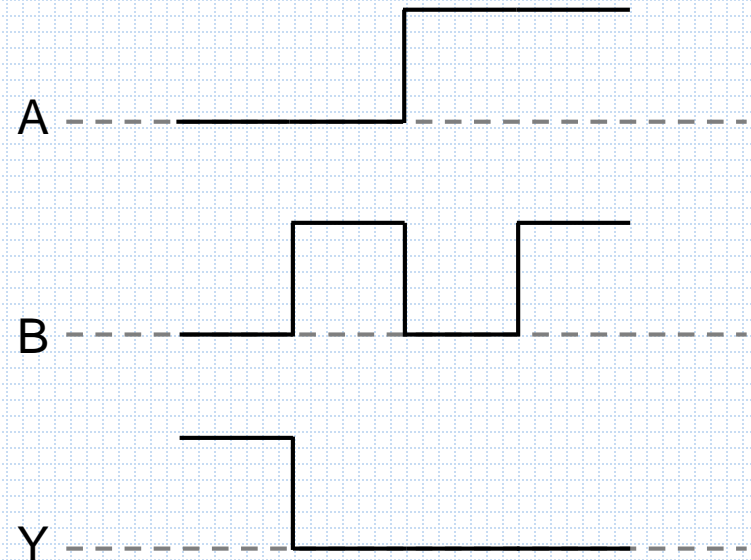
➤ Bảng chân lý

NOR

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

➤ Biểu thức $Y = \overline{A + B}$

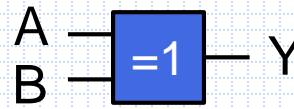
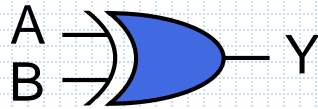
➤ Giải đồ thời gian



2. Giới thiệu về cổng logic

❑ Cổng HOẶC LOẠI TRỪ (EXCLUSIVE OR – XOR)

➤ Ký hiệu

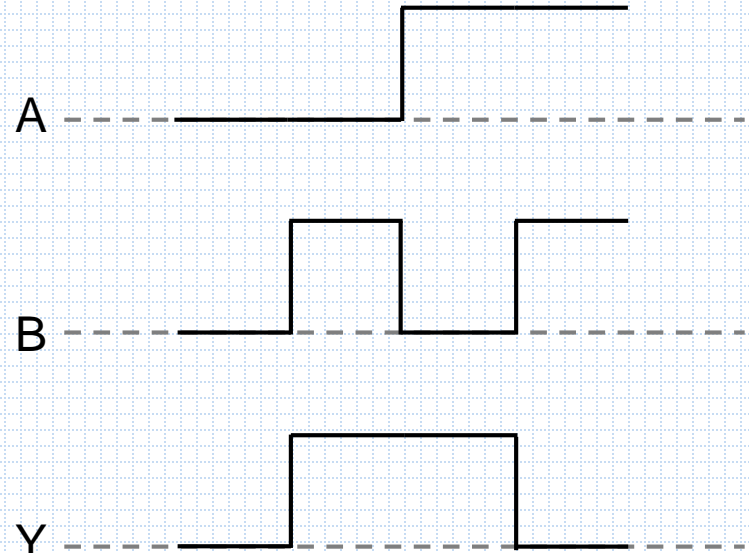


➤ Bảng chân lý

NOR

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

➤ Giải đồ thời gian

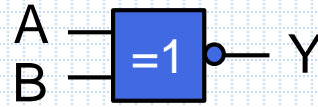
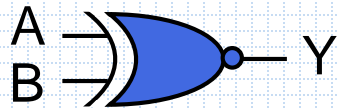


➤ Biểu thức $Y = A \oplus B = \overline{A} \cdot B + A \cdot \overline{B}$

2. Giới thiệu về cổng logic

❑ Cổng TƯƠNG ĐƯƠNG (EXCLUSIVE NOR – XNOR)

➤ Ký hiệu

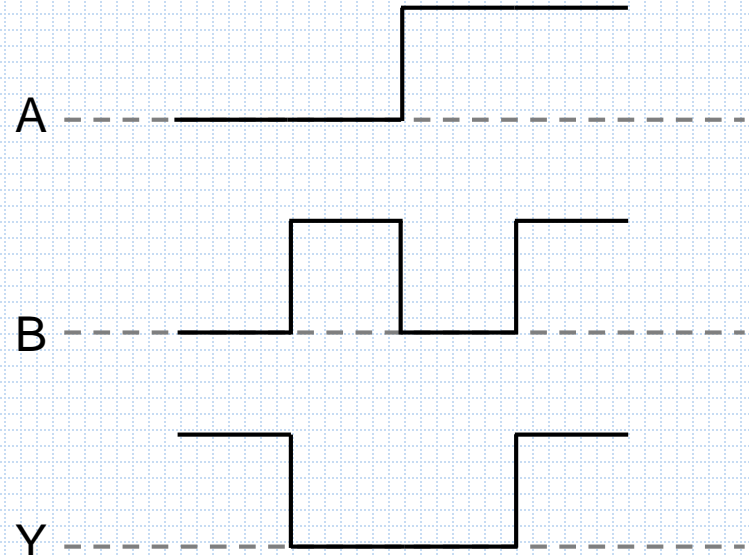


➤ Bảng chân lý

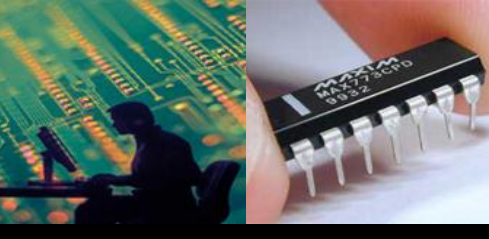
NOR

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

➤ Giải đồ thời gian

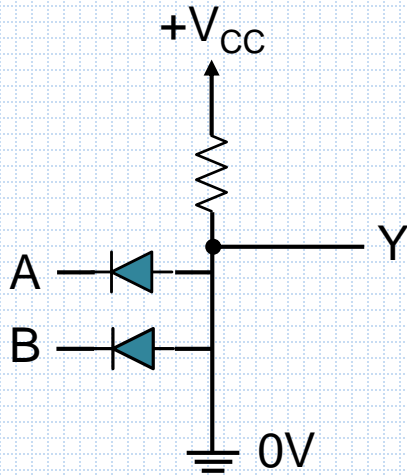


➤ Biểu thức $Y = \overline{A \oplus B} = A \cdot B + \overline{A} \cdot \overline{B}$

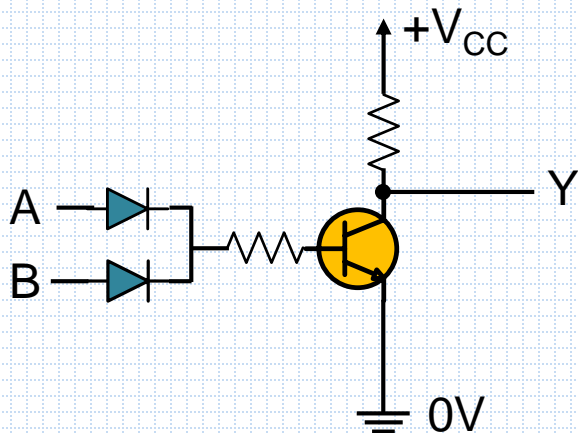


2. Giới thiệu về cổng logic

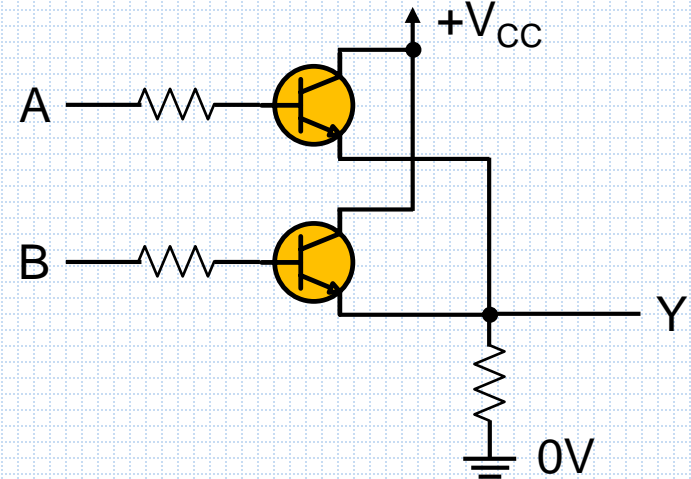
❑ Mạch điện họ DRL cổng AND



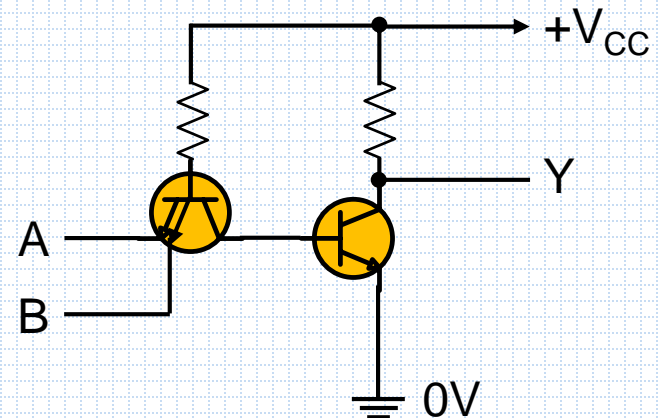
❑ Mạch điện họ DTL cổng NOR

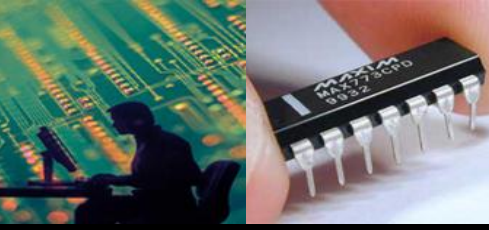


❑ Mạch điện họ RTL cổng OR



❑ Mạch điện họ TTL cổng NAND





NỘI DUNG CHƯƠNG 2

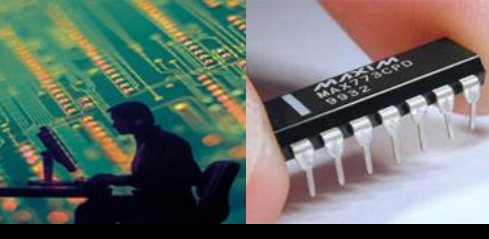
1. Cơ bản về đại số logic

2. Giới thiệu về cổng logic

3. Biểu diễn hàm logic

4. Tối thiểu hàm logic

5. Hệ hàm logic



3.1. Biểu diễn bằng bảng giá trị hàm

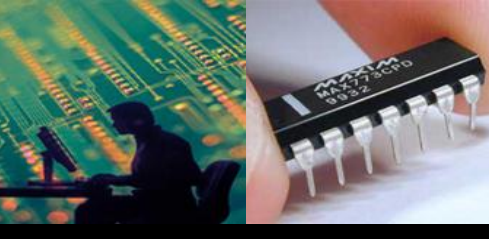
❑ **Cấu trúc bảng:** gồm $(n+1)$ cột và $2^n + 1$ hàng

➤ **Ưu điểm:** trực quan, dễ xác định chức năng logic của mạch

➤ **Nhược điểm:** cồng kềnh, khó đơn giản hàm

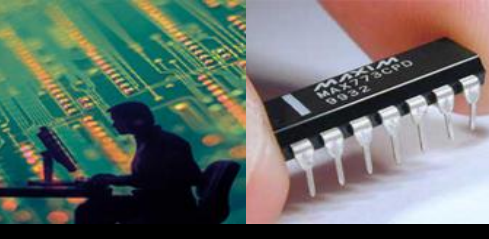
❑ **Ví dụ:** Biểu diễn hàm logic ba biến $Y = f(A, B, C)$ trong đó Y nhận giá trị 1 tại các tổ hợp biến 000, 010, 100, 110

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



3.2. Biểu diễn bằng biểu thức đại số

- ❑ **Tích đầy đủ:** Tích gồm n thừa số của n biến, mỗi biến xuất hiện 1 lần dưới dạng nguyên biến hay đảo biến. Có tất cả 2^n tích đầy đủ
 - Trong tích đầy đủ chỉ có một tổ hợp biến duy nhất làm cho giá trị hàm bằng 1
 - Tích của hai tích đầy đủ bất kỳ luôn bằng 0
 - Tổng của tất cả các tích đầy đủ luôn bằng 1
- ❑ **Tổng đầy đủ:** Tổng gồm n số hạng của n biến, mỗi biến xuất hiện 1 lần dưới dạng nguyên biến hay đảo biến. Có tất cả 2^n tổng đầy đủ
 - Trong tổng đầy đủ chỉ có một tổ hợp biến duy nhất làm cho giá trị hàm bằng 0
 - Tổng của hai tổng đầy đủ bất kỳ luôn bằng 1
 - Tích của tất cả các tổng đầy đủ luôn bằng 0



3.2. Biểu diễn bằng biểu thức đại số

□ Biểu diễn hàm dưới dạng chuẩn tắc tuyển (CTT) đầy đủ

- Dạng CTT đầy đủ là tổng (tuyển) của các tích đầy đủ
- Chỉ xét các tích đầy đủ làm cho hàm nhận giá trị 1.
- Nếu giá trị của biến trong tích đầy đủ bằng 1, dạng biểu diễn CTT của nó trong tích đầy đủ là nguyên biến, ngược lại là đảo biến

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

□ Biểu diễn hàm dưới dạng chuẩn tắc hội (CTH) đầy đủ

- Dạng CTH đầy đủ là tích (hội) của các tổng đầy đủ
- Chỉ xét các tổng đầy đủ làm cho hàm nhận giá trị 0.
- Nếu giá trị của biến trong tổng đầy đủ bằng 0, dạng biểu diễn CTH của nó trong tổng đầy đủ là nguyên biến, ngược lại là đảo biến

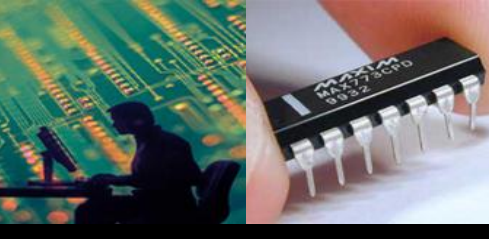
$$Y = (A+B+\overline{C}) (A+\overline{B}+\overline{C}) (\overline{A}+\overline{B}+C) (\overline{A}+\overline{B}+\overline{C})$$



3.3. Biểu diễn bằng bảng Karnaugh

❑ Cấu trúc bảng Karnaugh

- Bảng gồm 2^n ô, mỗi ô tương ứng với một tổ hợp biến
- Mép trái các hàng biểu diễn tổ hợp giá trị của nửa đầu trong toàn bộ n biến; nếu n chẵn sẽ có $2^{n/2}$ hàng, n lẻ sẽ có $2^{(n-1)/2}$ hàng. Các giá trị này đánh số theo mã Gray
- Mép trên các cột biểu diễn tổ hợp giá trị của nửa sau trong toàn bộ n biến; nếu n chẵn sẽ có $2^{n/2}$ cột, n lẻ sẽ có $2^{(n+1)/2}$ cột. Các giá trị này được đánh số theo mã Gray
- 2 tổ hợp biến kế cận trên mỗi hàng hay mỗi cột chỉ khác nhau 1 bit
- Các ô trong bảng ghi giá trị của hàm tương ứng với tổ hợp biến. Nếu ghi theo dạng CTT bắt buộc ghi số 1 tại các ô mà hàm nhận giá trị bằng 1, tại các ô hàm nhận giá trị bằng 0 ghi số 0 hoặc để trống. Nếu ghi theo dạng CTH bắt buộc ghi số 0 tại các ô mà hàm nhận giá trị bằng 0, tại các ô hàm nhận giá trị bằng 1 ghi số 1 hoặc để trống.



3.3. Biểu diễn bằng bảng Karnaugh

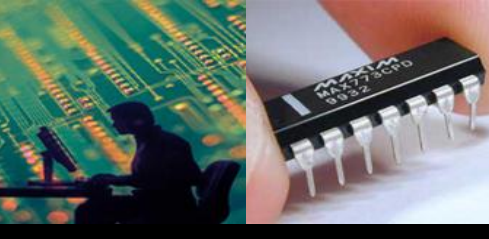
□ Ví dụ bảng Karnaugh

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

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

□ Ưu nhược điểm của bảng Karnaugh

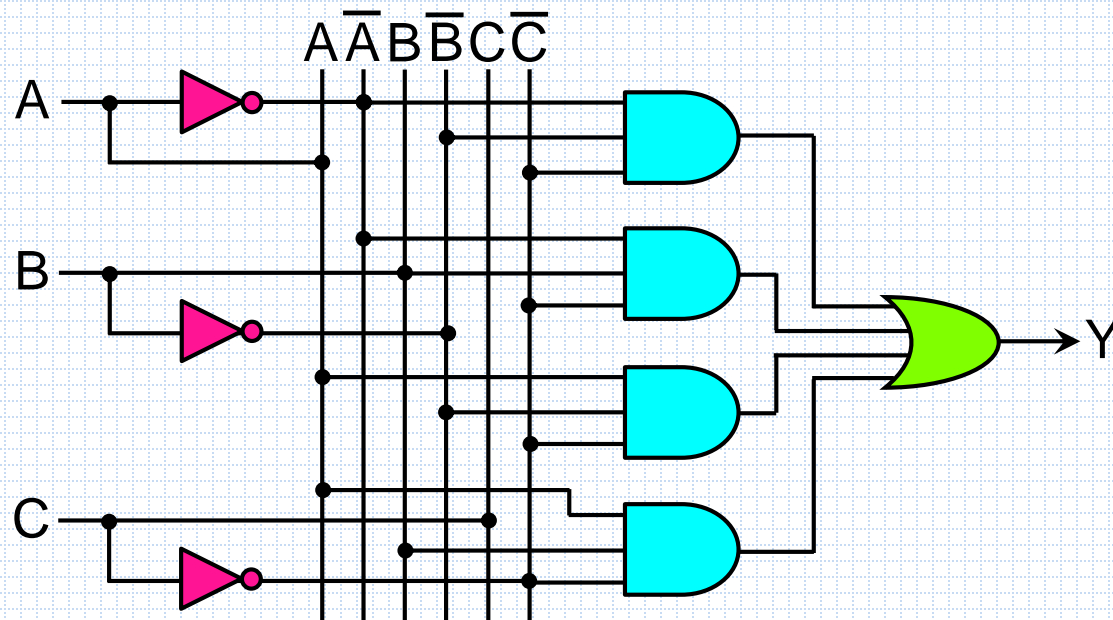
- Tính kế cận và đối xứng của các tổ hợp biến được thể hiện một cách rõ nét nhờ sắp xếp theo mã Gray
- Hỗ trợ trực quan và nhanh chóng cho quá trình tối thiểu hóa hàm logic khi số biến logic không quá lớn (không quá 6 biến)
- Độ phức tạp và công kênh của bảng tăng nhanh khi số biến tăng

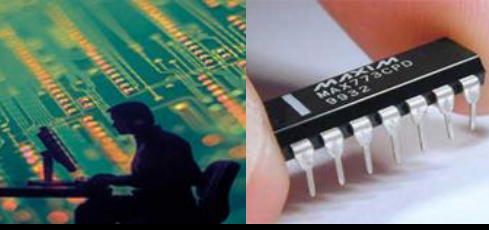


3.4. Biểu diễn bằng sơ đồ mạch logic

- Mọi hàm logic đều có thể biểu diễn thông qua các hàm logic cơ bản AND, OR và NOT và được thể hiện dưới dạng sơ đồ mạch logic sử dụng các cổng logic cơ bản AND, OR và NOT

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





NỘI DUNG CHƯƠNG 2

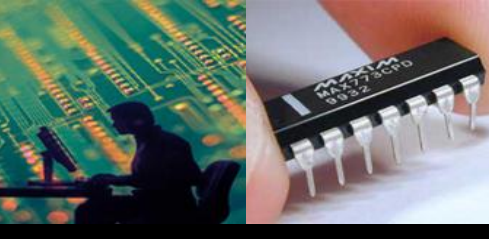
1. Cơ bản về đại số logic

2. Giới thiệu về cổng logic

3. Biểu diễn hàm logic

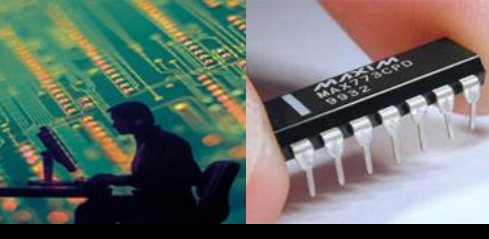
4. Tối thiểu hàm logic

5. Hệ hàm logic



4. Tối thiểu hàm logic

- ❑ **Lý do phải tối thiểu hàm logic:** hàm logic rút ra từ bảng chân lý thường phức tạp dẫn đến mạch logic tương ứng cũng phức tạp
- ❑ **Khái niệm đơn giản hàm logic:** tìm hàm logic mới đơn giản hơn mà không làm thay đổi chức năng của hàm
- ❑ **Khái niệm tối thiểu hàm logic:** tìm hàm logic mới đơn giản nhất mà không làm thay đổi chức năng của hàm
- ❑ **Bài toán tối thiểu hóa:** tìm các tích cực tiểu và phủ tối thiểu
 - **Tích cực tiểu** của hàm là một tích các biến tại đó hàm nhận giá trị 1 mà không bỏ bớt được một biến nào nữa
 - **Phủ tối thiểu** của hàm là tập hợp các tích cực tiểu ít nhất phủ đủ hết tất cả các tích đầy đủ làm cho hàm nhận giá trị 1
- ❑ **Các phương pháp tối thiểu hóa:** thông dụng nhất là phương pháp đại số, phương pháp bảng Karnagh và phương pháp Mc-Cluski



4.1 Phương pháp đại số

- ❑ **Bước 1:** Biểu diễn hàm logic dưới dạng biểu thức đại số
- ❑ **Bước 2:** Đơn giản và tối thiểu hàm logic dựa vào các định lý và tính chất của đại số logic
- ❑ **Phương pháp nhóm số hạng**

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

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

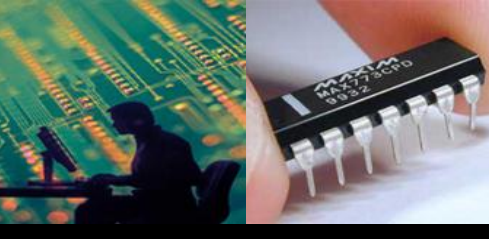
$$Y = \overline{B}\overline{C} + B\overline{C} = \overline{C}(\overline{B} + B) = \overline{C}$$

- ❑ **Phương pháp thêm số hạng**

$$Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

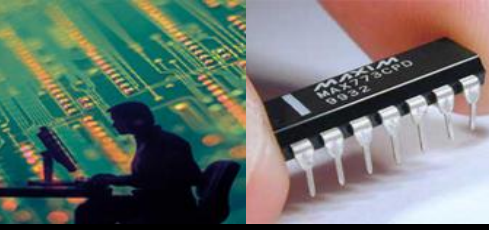
$$Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC + ABC + ABC$$

$$Y = AB + AC + BC$$



4.2 Phương pháp bảng Karnaugh

- ❑ **Bước 1:** Biểu diễn hàm bằng bảng Karnaugh
- ❑ **Bước 2:** Gộp nhóm. Gộp các ô bằng 1 thành các nhóm sao cho số ô trong nhóm lớn nhất có thể (tích cực tiểu) và thỏa mãn điều kiện:
 - Số ô trong nhóm là lũy thừa của 2 (2, 4, 8,...)
 - Các ô trong nhóm phải là kế cận hoặc đối xứng
 - Số nhóm phải là ít nhất (phủ tối thiểu)
- ❑ **Bước 3:** Viết biểu thức hàm logic tối thiểu
 - Hàm logic tối thiểu bằng tổng các tích cực tiểu, mỗi tích tương ứng với một nhóm vừa gộp
 - Trong tích cực tiểu biến có giá trị 1 tại tất cả các ô của nhóm được viết dưới dạng nguyên biến; biến có giá trị 0 viết ở dạng đảo biến; biến nhận cả giá trị 0 và 1 được loại bỏ và không có mặt trong tích



4.2 Phương pháp bảng Karnaugh

□ Ví dụ: Biểu diễn hàm bằng bảng Karnaugh

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

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

Y \ CD	AB							
	000	001	011	010	110	111	101	100
00	0	1	0	1	1	0	1	0
01	0	1	0	1	1	0	1	0
11	0	0	0	0	0	0	1	0
10	0	1	0	1	0	0	1	0



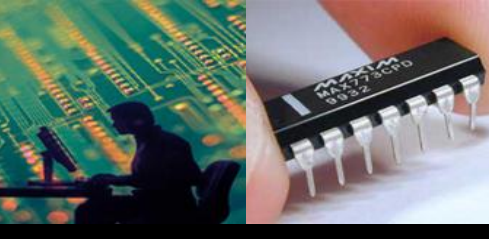
4.3 Phương pháp MC-CLUSKI

□ **Nhiệm vụ 1:** Tìm các tích cực tiểu

- **Bước 1:** Phân nhóm các tổ hợp biến làm cho hàm bằng 1 theo thứ tự tăng dần số bit bằng 1 trong tổ hợp
- **Bước 2:** Thay thế lần đầu tập hợp nhóm gốc bằng các nhóm mới theo nguyên tắc: so sánh lần lượt từng tổ hợp của nhóm i với tất cả các tổ hợp của nhóm $(i + 1)$, i chạy từ 1 đến áp chót; nếu chúng chỉ khác nhau tại một vị trí thì thay hai tổ hợp này bằng tổ hợp mới có dấu '—' tại vị trí khác nhau. Đánh dấu các tổ hợp đã được thay thế.
- **Bước 3:** Tiếp tục thay thế lần 1, lần 2 ... cho các tập hợp nhóm tương tự như ở bước 2 cho tới khi hết khả năng thay thế thì thôi

□ **Nhiệm vụ 2:** Tìm phủ tối thiểu của các tích cực tiểu

- **Bước 4:** Lập bảng Karnaugh với các cột là các tổ hợp biến làm cho hàm bằng 1, các hàng là các tích cực tiểu thu được ở bước 3, đánh dấu 'x' vào ô nếu tích cực tiểu tại hàng bao phủ tổ hợp biến tại cột
- **Bước 5:** Tìm các cột có 1 dấu nhân, từ đó suy ra các tích cực tiểu bao phủ. Nếu các tổ hợp biến chưa được bao phủ hết chọn thêm các tích cực tiểu có nhiều dấu 'x' cho đến khi bao phủ toàn bộ



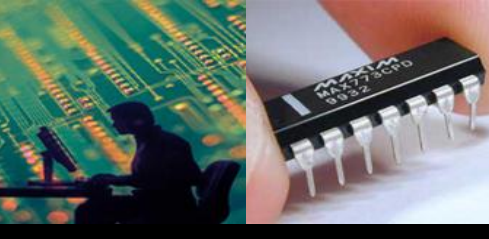
4.3 Phương pháp MC-CLUSKI

□ Ví dụ: Tối thiểu hàm logic bốn biến

$$Y = f(A, B, C, D) = f^1(0, 1, 2, 3, 4, 6, 7, 8, 9, 11, 13, 14, 15)$$

➤ Tìm các tích cực tiểu

Nhóm	Sắp xếp lần đầu	Thay thế lần 1	Thay thế lần 2
Có 0 bit 1	0000*	000—*, 00—0* 0—00*, —000*	00—, 0—0, —00—
Có 1 bit 1	0001*, 0010* 0100*, 1000*	00—1*, —001*, 001—* 0—10*, 01—0*, 100—*	—0—1, 0—1—
Có 2 bit 1	0011*, 0110*, 1001*	0—11*, —011*, 011—*, —110*, 10—1*, 1—01*	—11, —11—, 1—1
Có 3 bit 1	0111*, 1011* 1101*, 1110*	—111*, 1—11* 11—1*, 111—*	
Có 4 bit 1	1111*		



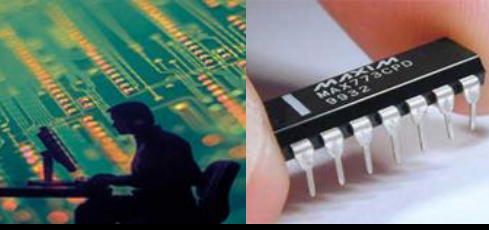
4.3 Phương pháp MC-CLUSKI

□ Ví dụ: Tối thiểu hàm logic bốn biến

$$Y = f(A, B, C, D) = f^1(0, 1, 2, 3, 4, 6, 7, 8, 9, 11, 13, 14, 15)$$

➤ Tìm phủ tối thiểu của các tích cực tiểu

	0000	0001	0010	0011	0100	0110	0111	1000	1001	1011	1101	1110	1111
00—	x	x	x	x									
0—0	x		x		x	x							
—00	x	x						x	x				
—0—1		x		x					x	x			
0—1—			x	x		x	x						
—11				x			x			x			
—11—						x	x					x	x
1—1									x	x	x		x



NỘI DUNG CHƯƠNG 2

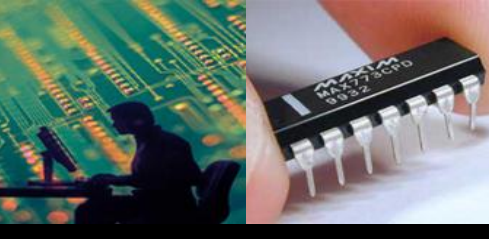
1. Cơ bản về đại số logic

2. Giới thiệu về cổng logic

3. Biểu diễn hàm logic

4. Tối thiểu hàm logic

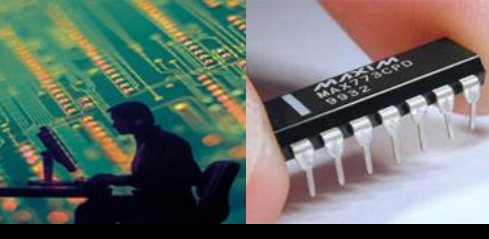
5. Hệ hàm logic



5.1. Khái niệm hệ hàm logic

□ **Định nghĩa:** Hệ hàm logic là tập hợp các hàm logic có cùng chung biến logic

$$\left\{ \begin{array}{l} y_0 = f_0(x_{n-1}, x_{n-2}, \dots, x_1, x_0) \\ y_1 = f_1(x_{n-1}, x_{n-2}, \dots, x_1, x_0) \\ \dots \\ y_{m-1} = f_{m-1}(x_{n-1}, x_{n-2}, \dots, x_1, x_0) \end{array} \right.$$



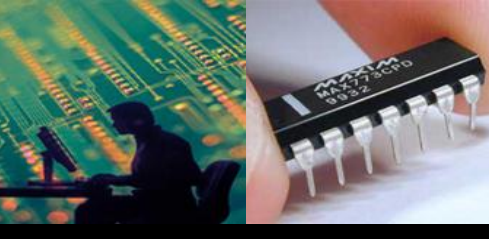
5.2. Các phương pháp biểu diễn hệ hàm

❑ **Phương pháp bảng giá trị hàm:** lập bảng có số hàng bằng 2^n tương ứng với 2^n tổ hợp biến, số cột bằng $(m+n)$ với n cột đầu ghi giá trị của n biến, m cột sau ghi giá trị của m hàm

➤ **Ví dụ:** Hệ hàm logic chuyển đổi mã BCD sang mã dư 3

Số hệ 10	Mã BCD D C B A	Mã dư 3 $d_3 d_2 d_1 d_0$
0	0 0 0 0	0 0 1 1
1	0 0 0 1	0 1 0 0
2	0 0 1 0	0 1 0 1
3	0 0 1 1	0 1 1 0
4	0 1 0 0	0 1 1 1
5	0 1 0 1	1 0 0 0
6	0 1 1 0	1 0 0 1
7	0 1 1 1	1 0 1 0

Số hệ 10	Mã BCD D C B A	Mã dư 3 $d_3 d_2 d_1 d_0$
8	1 0 0 0	1 0 1 1
9	1 0 0 1	1 1 0 0
10	x x x x	x x x x
11	x x x x	x x x x
12	x x x x	x x x x
13	x x x x	x x x x
14	x x x x	x x x x
15	x x x x	x x x x

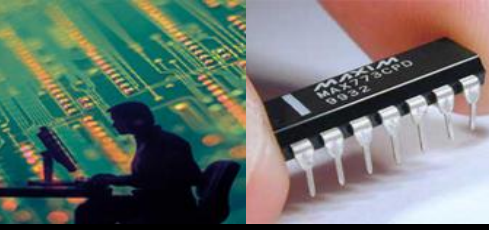


5.2. Các phương pháp biểu diễn hệ hàm

□ **Phương pháp đại số:** dùng m biểu thức đại số để biểu diễn cho m hàm, mỗi biểu thức được rút ra theo từng hàm logic đơn

➤ **Ví dụ:** Hệ hàm logic chuyển đổi mã BCD sang mã dư 3

$$\left\{ \begin{array}{l} d_3 = D \cdot \bar{C} \cdot \bar{B} \cdot A + D \cdot \bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{D} \cdot C \cdot B \cdot A + \bar{D} \cdot C \cdot B \cdot \bar{A} + \bar{D} \cdot C \cdot \bar{B} \cdot A \\ d_2 = D \cdot \bar{C} \cdot \bar{B} \cdot A + \bar{D} \cdot C \cdot \bar{B} \cdot \bar{A} + \bar{D} \cdot \bar{C} \cdot B \cdot A + \bar{D} \cdot \bar{C} \cdot B \cdot \bar{A} + \bar{D} \cdot \bar{C} \cdot \bar{B} \cdot A \\ d_1 = D \cdot \bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{D} \cdot C \cdot B \cdot A + \bar{D} \cdot C \cdot \bar{B} \cdot \bar{A} + \bar{D} \cdot \bar{C} \cdot B \cdot A + \bar{D} \cdot \bar{C} \cdot \bar{B} \cdot \bar{A} \\ d_0 = D \cdot \bar{C} \cdot \bar{B} \cdot \bar{A} + \bar{D} \cdot C \cdot B \cdot \bar{A} + \bar{D} \cdot C \cdot \bar{B} \cdot \bar{A} + \bar{D} \cdot \bar{C} \cdot B \cdot \bar{A} + \bar{D} \cdot \bar{C} \cdot \bar{B} \cdot \bar{A} \end{array} \right.$$



5.3. Tối thiểu hệ hàm logic

□ Phương pháp tách riêng từng hàm

➤ Ví dụ: Hệ hàm logic chuyển đổi mã BCD sang mã dư 3

d_3

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

d_1

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

d_2

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

d_0

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



5.3. Tối thiểu hệ hàm logic

❑ **Phương pháp sử dụng các phần chung:** Tách riêng phần chung cho các hàm và một số hàm. Một hàm bất kỳ sẽ bằng tổng các phần chung với các hàm khác và phần riêng còn lại

➤ **Ví dụ:** Tối thiểu hệ hàm logic sau

y_3

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

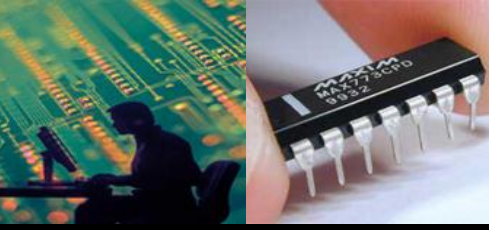
y_2

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

y_1

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

$$\begin{cases} y_3 = A \cdot \bar{D} + B \cdot C \cdot D + \bar{B} \cdot \bar{C} \cdot \bar{D} \\ y_2 = \bar{A} \cdot \bar{B} + B \cdot C \cdot D + A \cdot B \cdot \bar{D} \\ y_1 = A \cdot \bar{D} + A \cdot C + \bar{A} \cdot \bar{B} \cdot \bar{D} \end{cases}$$



5.3. Tối thiểu hệ hàm logic

y_3

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

y_2

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

y_1

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

y_{321}

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

y_{31}

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

y_{21}

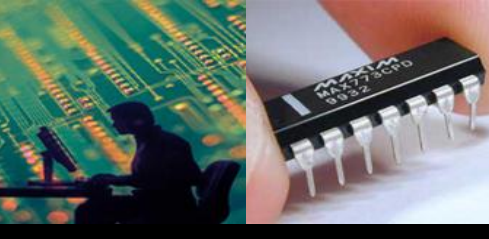
CD \ AB	00	01	11	10
00	1	0	0	1
01	0	0	x	0
11	1	0	x	1
10	0	0	0	0

$$y_{321} = B \cdot C \cdot D$$

$$y_{31} = A \cdot \bar{D}$$

$$y_{21} = A \cdot B \cdot \bar{D} + \bar{A} \cdot \bar{B} \cdot \bar{D}$$

$$= \bar{D} \cdot (\bar{A} \oplus B)$$



5.3. Tối thiểu hệ hàm logic

y_3

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

y_2

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

y_1

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

y'_3

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

y'_2

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

$$\begin{cases} y_3 = y'_3 + y_{321} + y_{31} \\ y_2 = y'_2 + y_{321} + y_{21} \\ y_1 = y_{321} + y_{31} \end{cases}$$