

# Phần 1: Hệ thống số

1. Giới thiệu

2. Hệ thống số và mã



# 1. Giới thiệu

1.1 Các đại lượng số và tương tự

1.2 Số nhị phân, mức logic và dạng sóng số

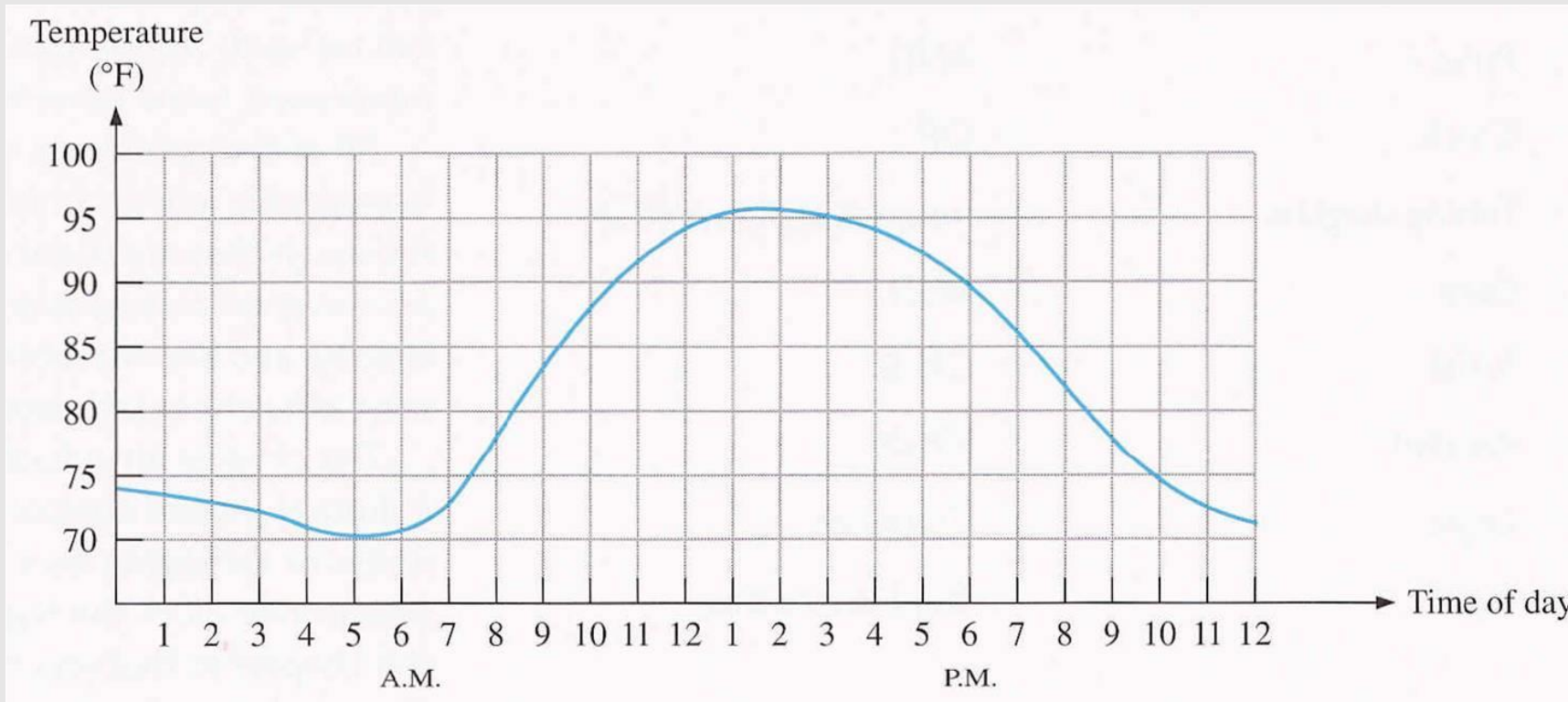
1.3 Các phép toán logic cơ bản

1.4 Tổng quan về hàm logic cơ bản



## 1.1 Các đại lượng tương tự-số

- Tín hiệu tương tự (analog signal)



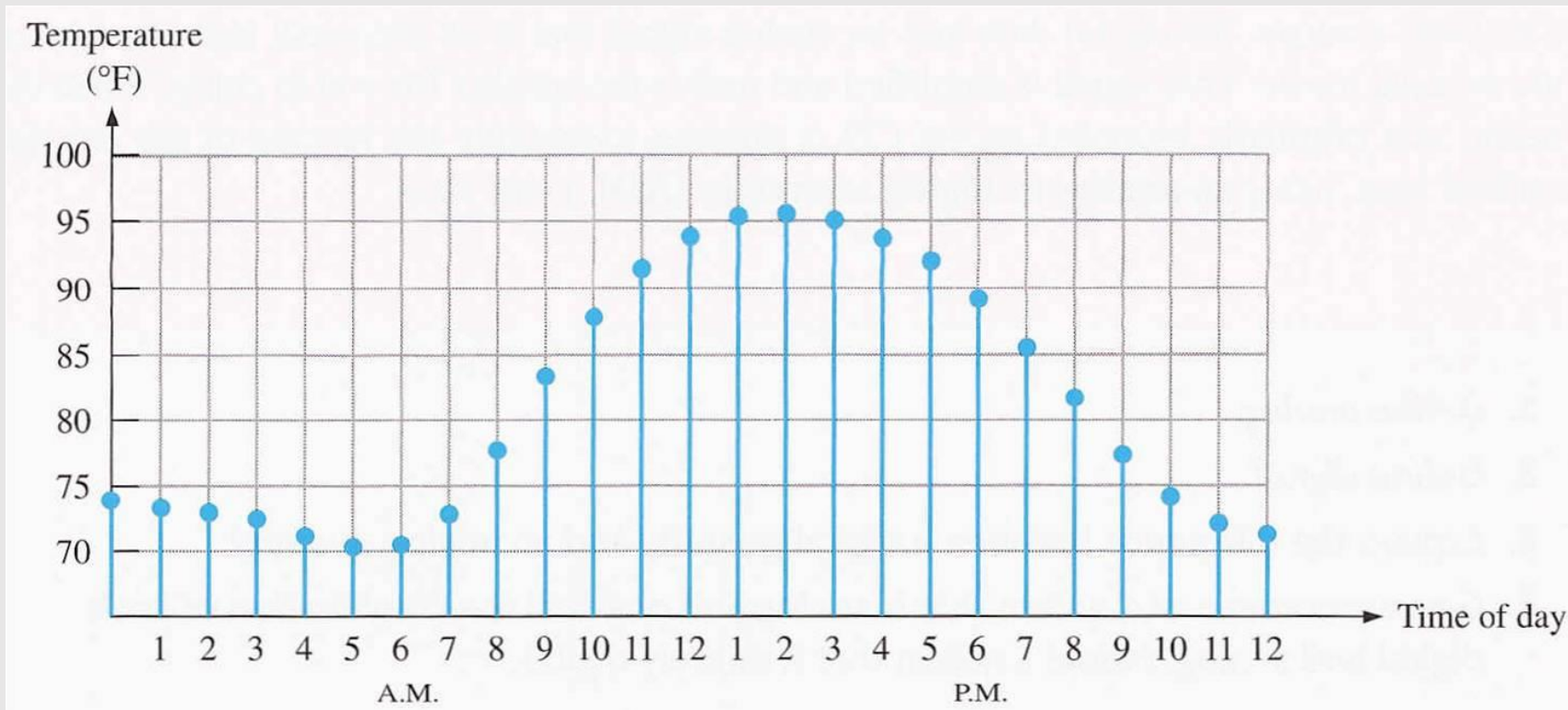
Tín hiệu có biên độ liên tục theo thời gian.

Chúng ta sẽ thấy loại tín hiệu này hoạt động xung quanh chúng ta: tần số, âm thanh, gió, sóng, v.v.



## 1.1 Các đại lượng tương tự-số

- Tín hiệu rời rạc (discrete signal)



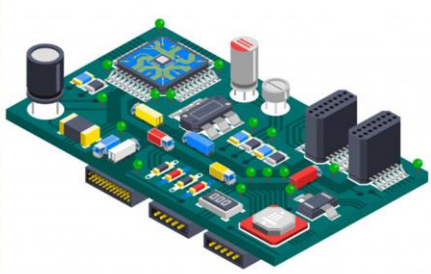
Biểu diễn giá trị đã lấy mẫu của đại lượng tương tự. Mỗi giá trị được biểu diễn bằng một điểm. Có vô số giá trị trong khoảng thời gian quan sát.



## 1.1 Các đại lượng tương tự-số

- Tín hiệu số (digital signal)

- Ta lượng tử hóa các giá trị của tín hiệu rời rạc: số lượng giá trị là  $\infty$  của tín hiệu rời rạc được quy về một số giới hạn các giá trị.
- Sử dụng mã số (digital code) bao gồm một chuỗi các số (bit) 1 và 0 để “mã hóa” số giới hạn các giá trị nêu trên  $\rightarrow$  ta có tín hiệu số (digital signal).



## 1.1 Các đại lượng tương tự-số

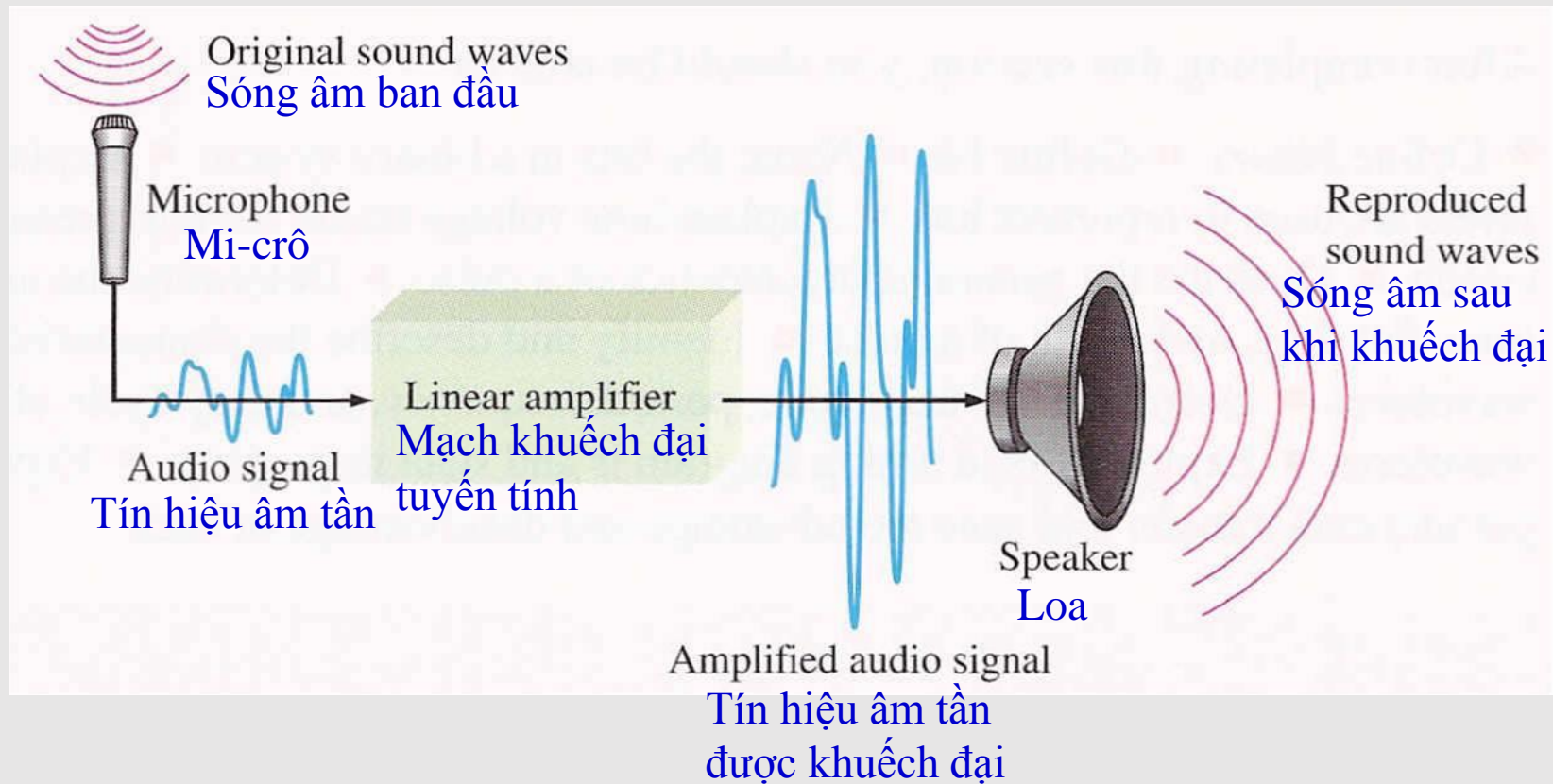
- Ưu điểm của mạch số so với mạch tương tự
  - Dễ thiết kế, phân tích.
  - Hoạt động theo chương trình lập sẵn.
  - Ít bị ảnh hưởng của nhiễu
  - Dễ chế tạo thành mạch tích hợp.

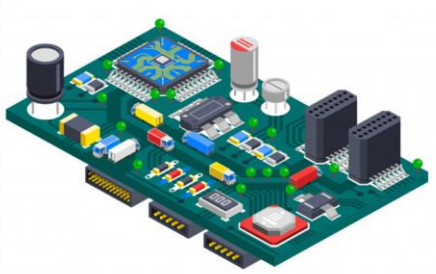




## 1.1 Các đại lượng tương tự-số

- Hệ thống tương tự điển hình





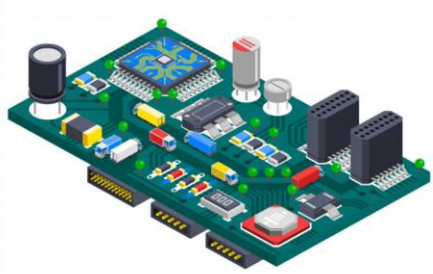
## 1.1 Các đại lượng tương tự-số

- Hệ thống số điển hình



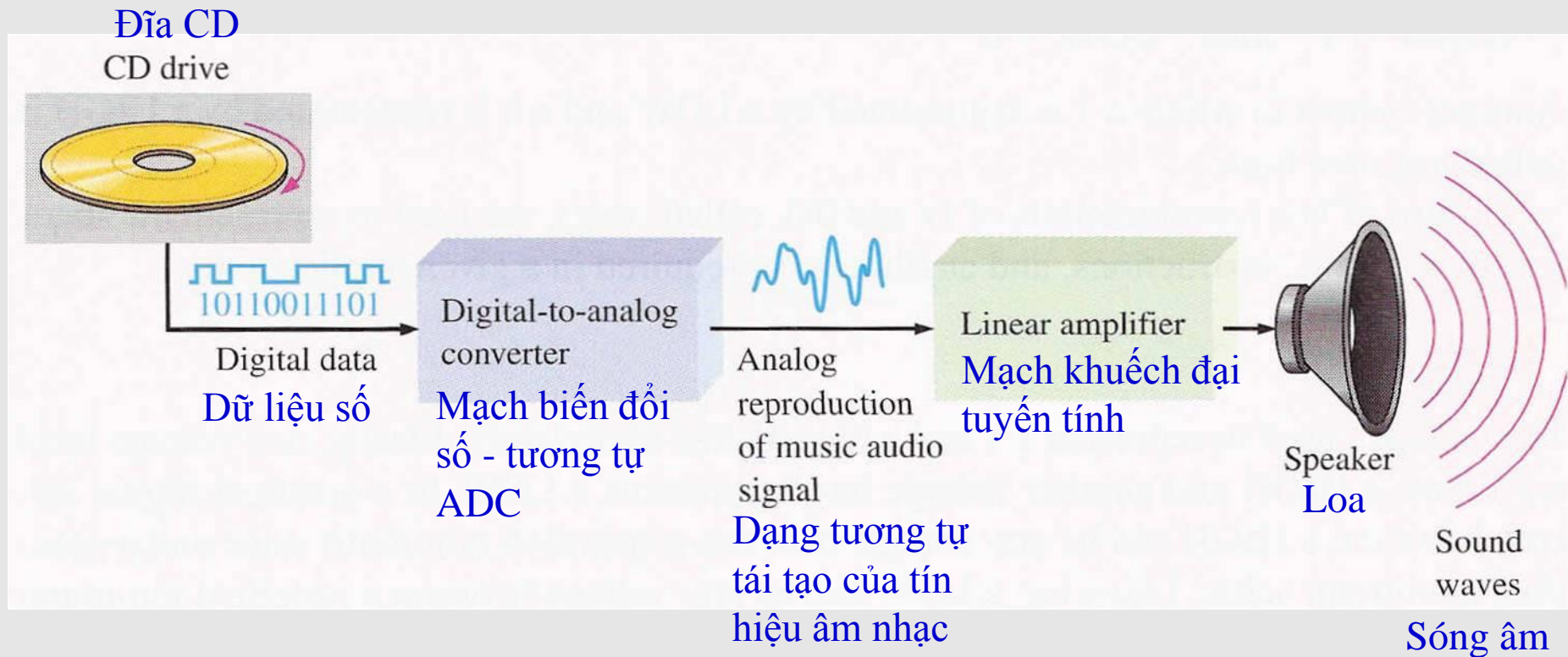
Mạch điện tử số điều khiển các đèn sáng – tắt theo thời gian

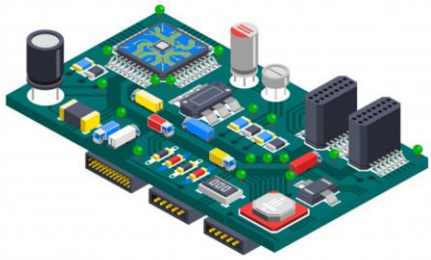




## 1.1 Các đại lượng tương tự-số

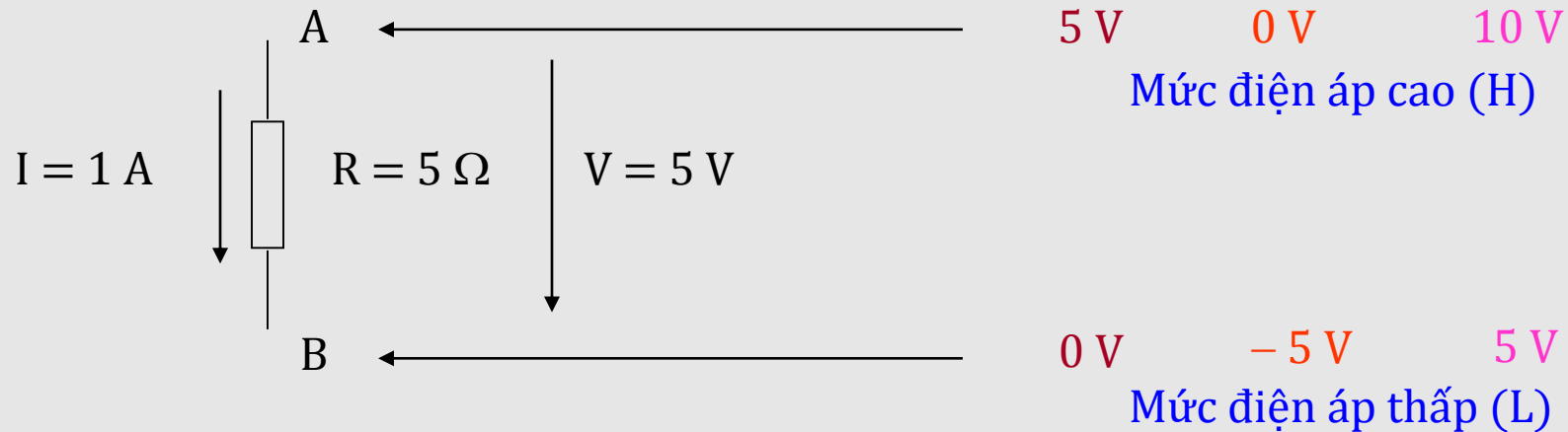
- Hệ thống số - tương tự điển hình





## 1.2 Số nhị phân, mức logic và dạng sóng số

- Khái niệm logic trong mạch điện



H → logic 1 (hay 1)

L → logic 1 (hay 1)

L → logic 0 (hay 0)

H → logic 0 (hay 0)

$V_{\text{logic1}} > V_{\text{logic0}}$

$V_{\text{logic1}} < V_{\text{logic0}}$

(Logic dương)

(Logic âm)



## 1.2 Số nhị phân, mức logic và dạng sóng số

- Số nhị phân

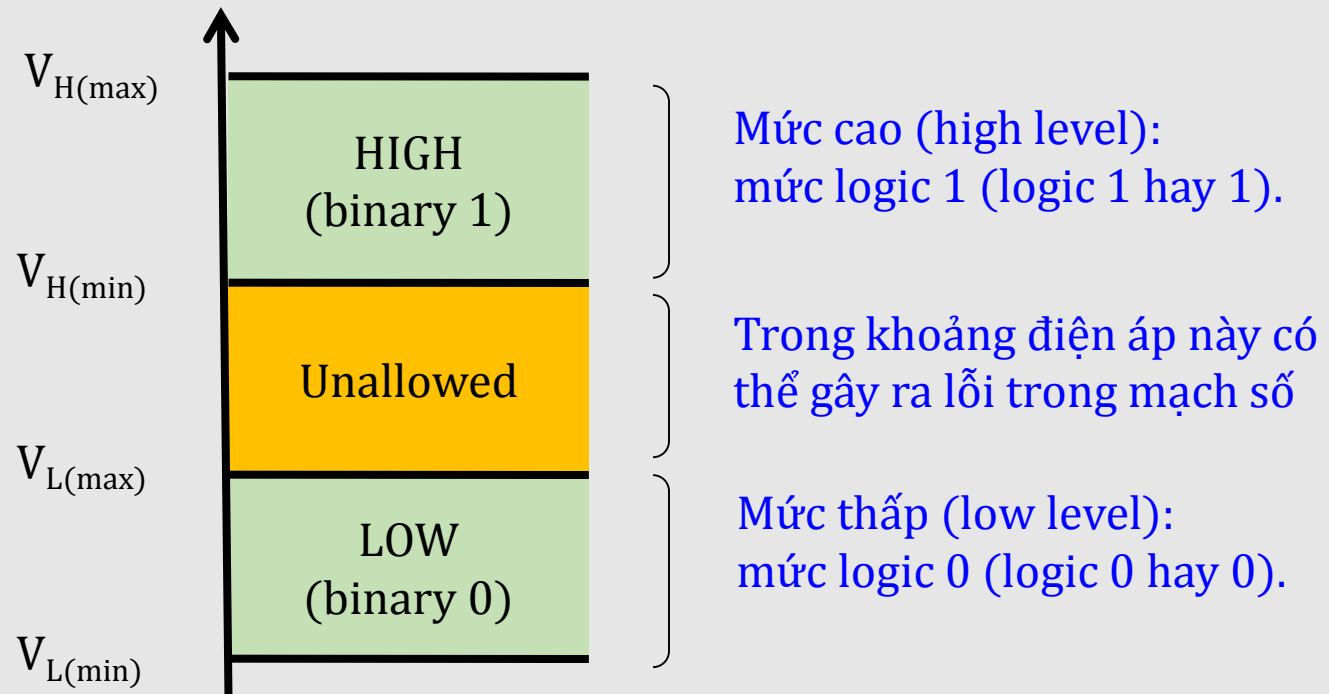
Hệ thống số nhị phân sử dụng hai **digit nhị phân** là 0 và 1 để biểu diễn số, ta còn gọi là hai **bit** 0 và 1.

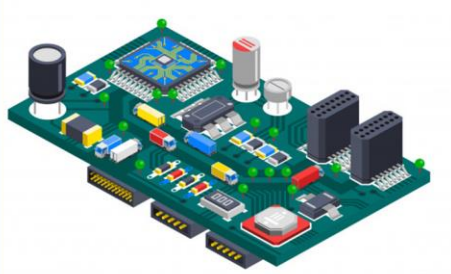
- Mức logic

Logic dương

$$V_{\text{logic1}} > V_{\text{logic0}}$$

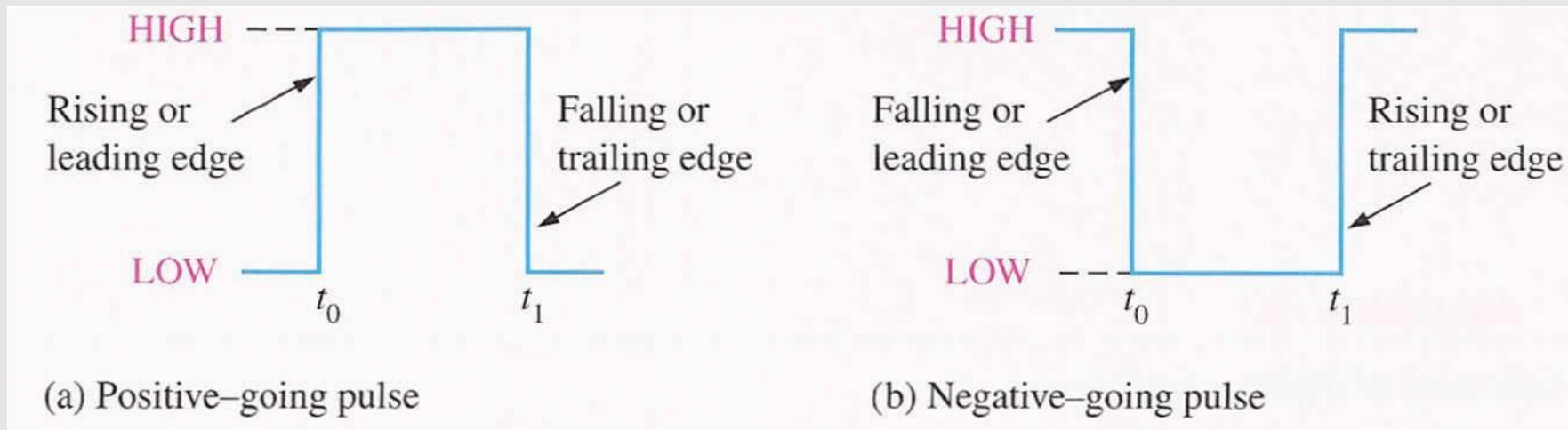
Ngược lại, ta có  
logic âm





## 1.2 Số nhị phân, mức logic và dạng sóng số

- Dạng sóng số



Xung dương được tạo ra khi điện áp đi từ mức thấp lên mức cao rồi trở xuống mức thấp

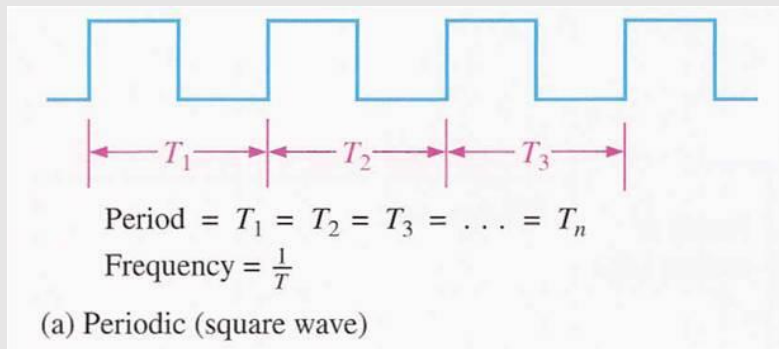
Xung âm được tạo ra khi điện áp đi từ mức cao đến mức thấp rồi lại trở lại mức cao



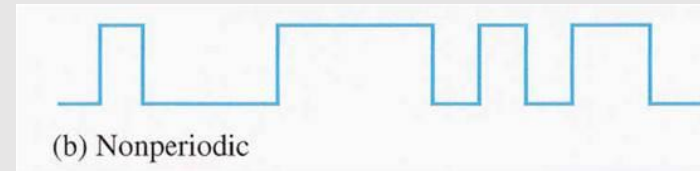
## 1.2 Số nhị phân, mức logic và dạng sóng số

- Dạng sóng số

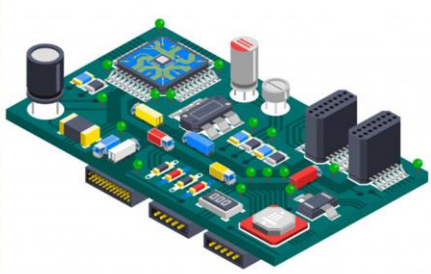
### Xung lý tưởng



Xung tuần hoàn: xung tự lặp lại sau những khoảng thời gian cố định (chu kỳ T)

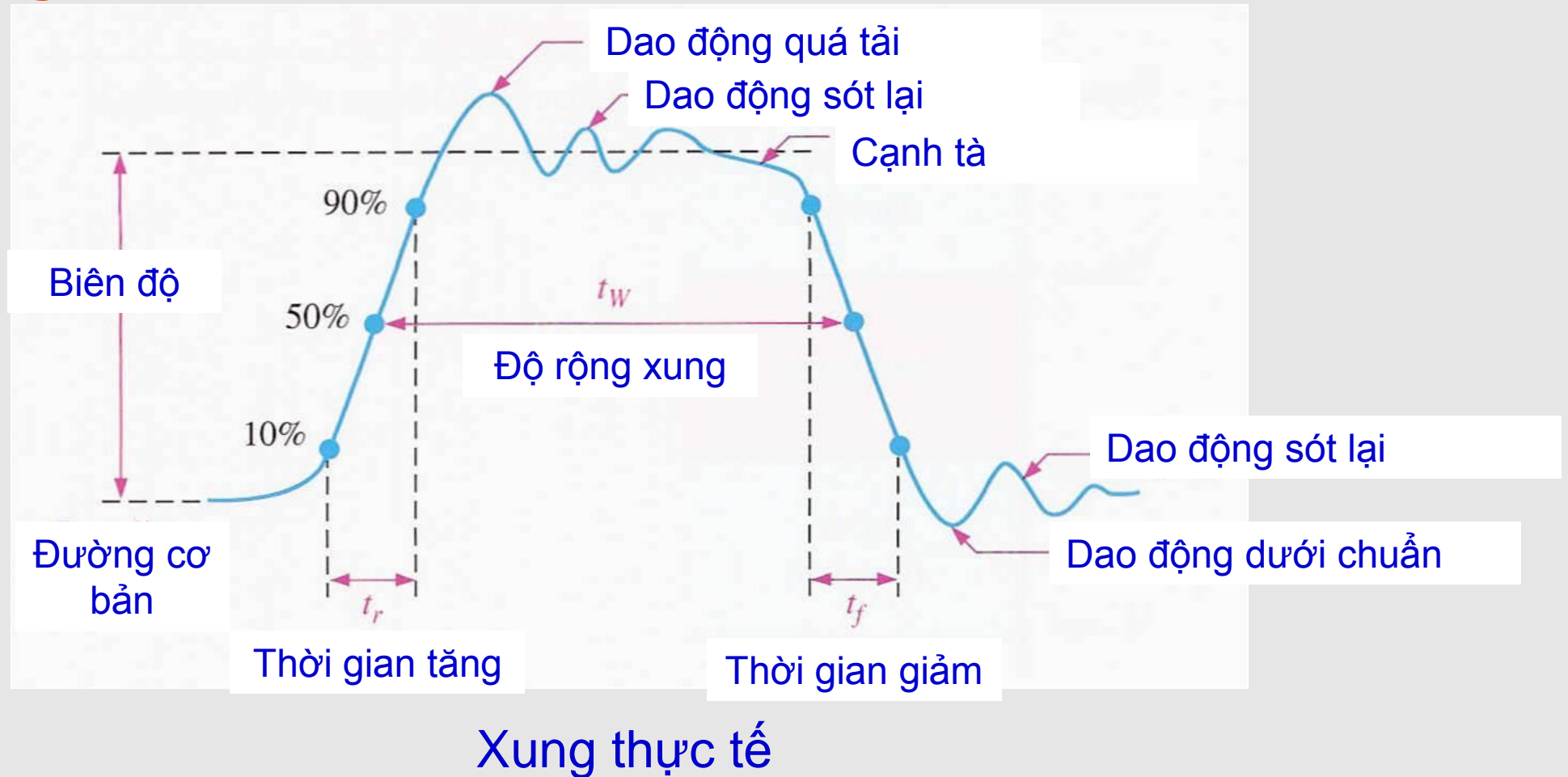


Xung không tuần hoàn: xung không lặp lại sau những khoảng thời gian cố định và có thể bao gồm những xung có độ rộng khác nhau



## 1.2 Số nhị phân, mức logic và dạng sóng số

- Dạng sóng số

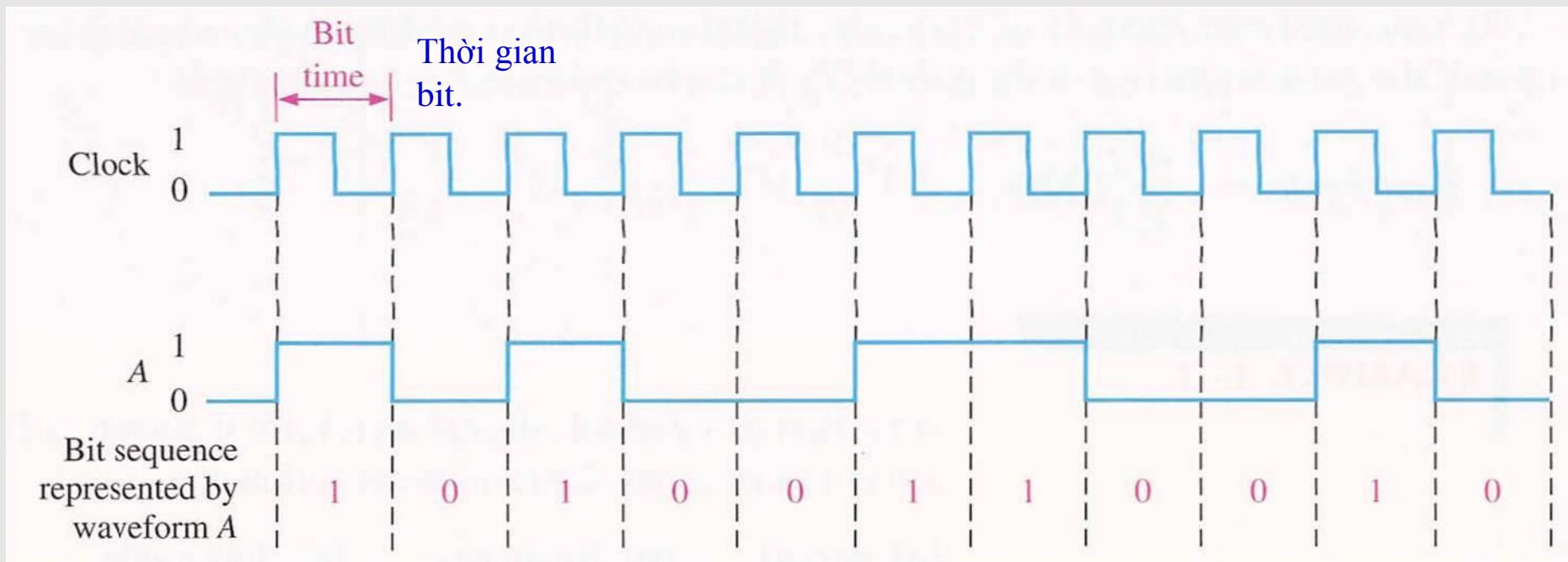






## 1.2 Số nhị phân, mức logic và dạng sóng số

- Dạng sóng số mang thông tin nhị phân



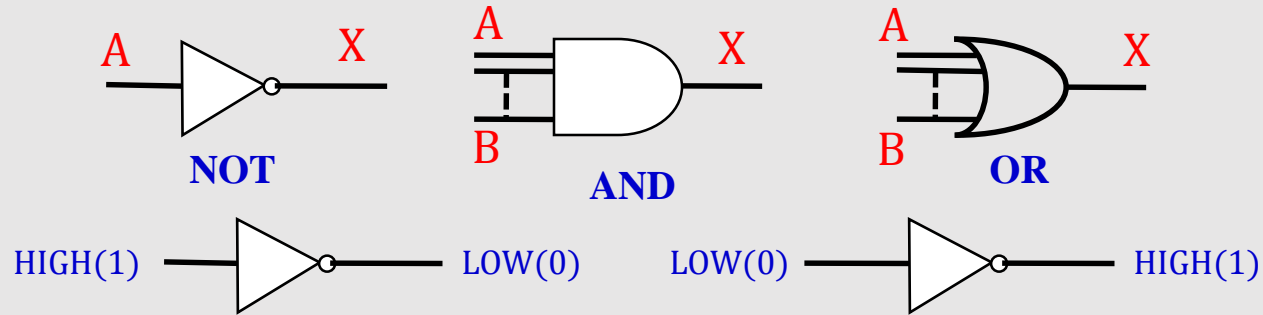
Chuỗi bit biểu  
diễn dạng sóng  
của A.

Thông tin nhị phân đưa vào hệ thống số là dạng sóng được biểu diễn bằng một chuỗi các bit. Khi dạng sóng cao, số nhị phân 1 được đưa vào; khi dạng sóng thấp, số nhị phân 0 được đưa vào

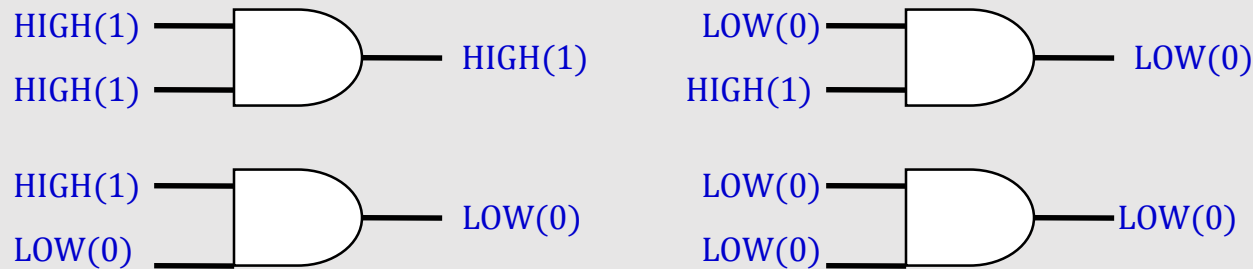


## 1.3 Các phép toán logic cơ bản

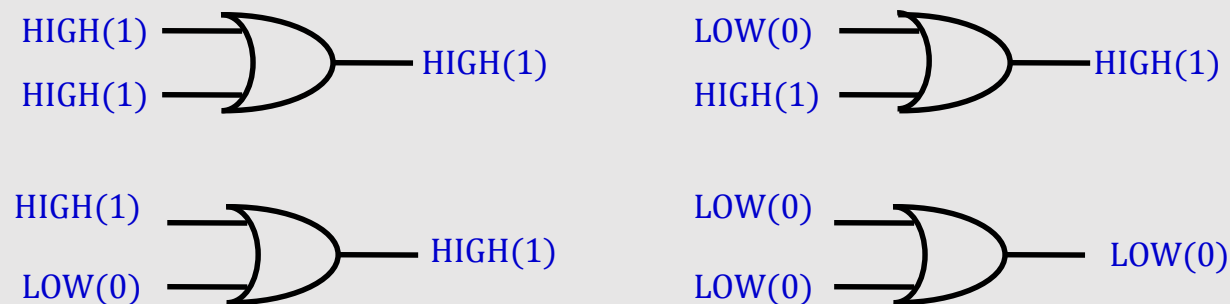
$$X = \bar{A}$$



$$X = A * B$$



$$X = A + B$$





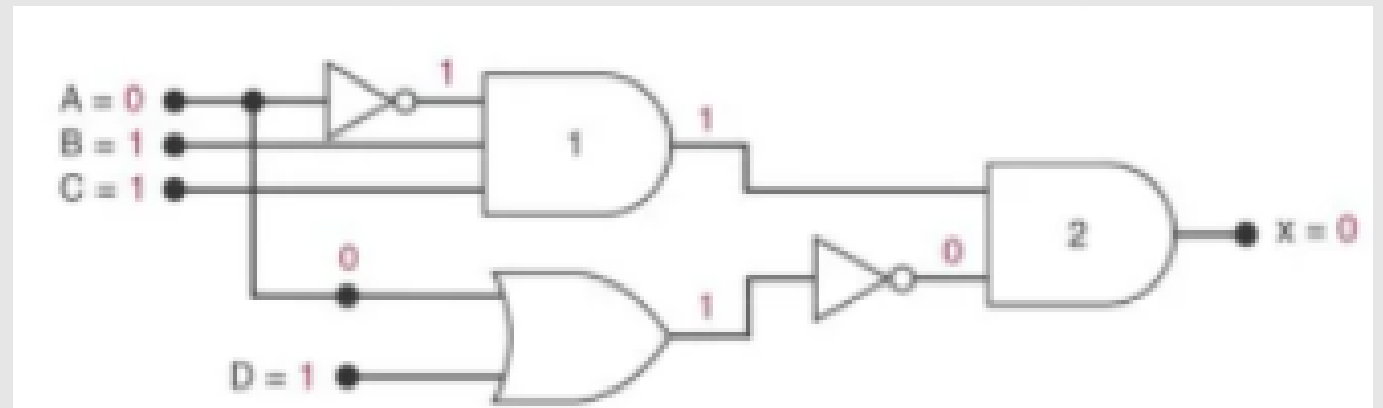
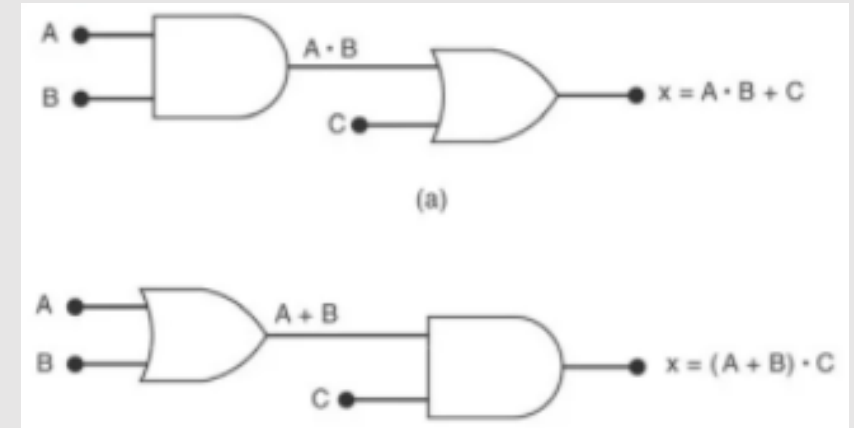
## 1.3 Các phép toán logic cơ bản

- Bất kỳ mạch logic nào cũng có thể được xây dựng từ 3 cổng logic cơ bản: AND NOT OR

- VD:  $X = AB + C$

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

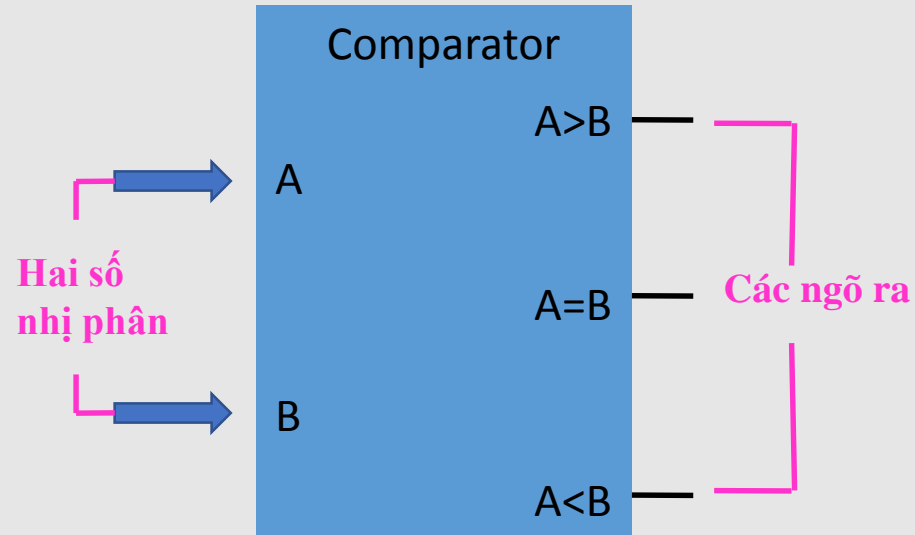
$$X = \overline{A}BC(\overline{A + D})$$



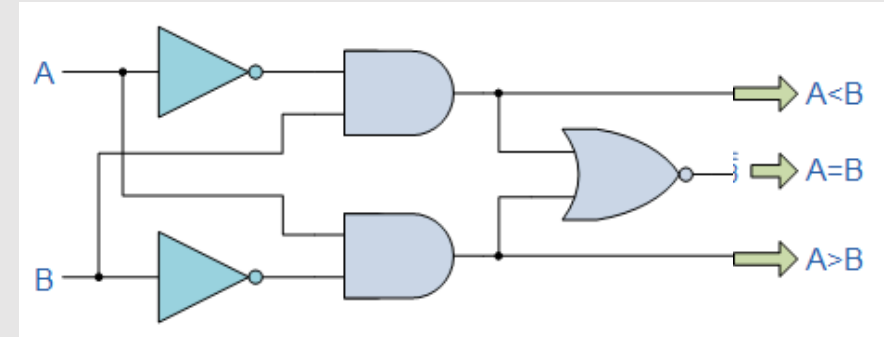


## 1.4 Tổng quan về hàm logic cơ bản

- Hàm so sánh



Mạch so sánh độ lớn dạng cơ bản



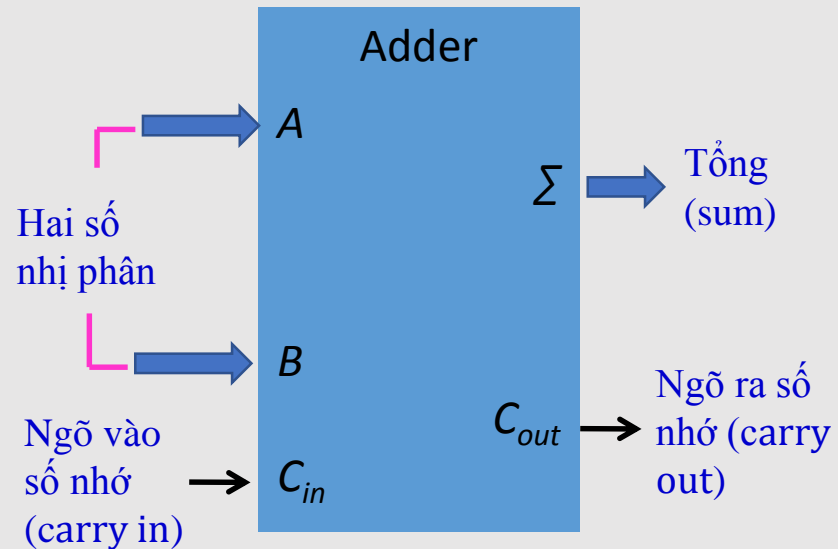
Bộ so sánh 1 bit

Đầu vào		Kết quả đầu ra		
B	A	A > B	A = B	A < B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

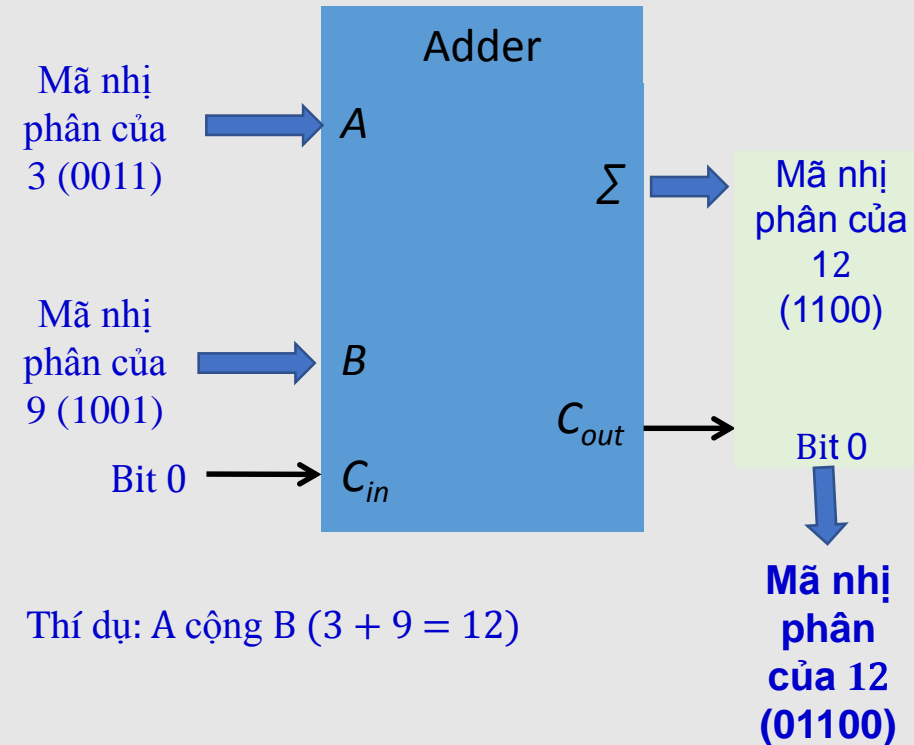


## 1.4 Tổng quan về hàm logic cơ bản

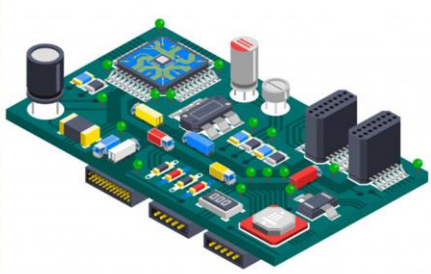
- Hàm số học  
Cộng



Mạch cộng cơ bản

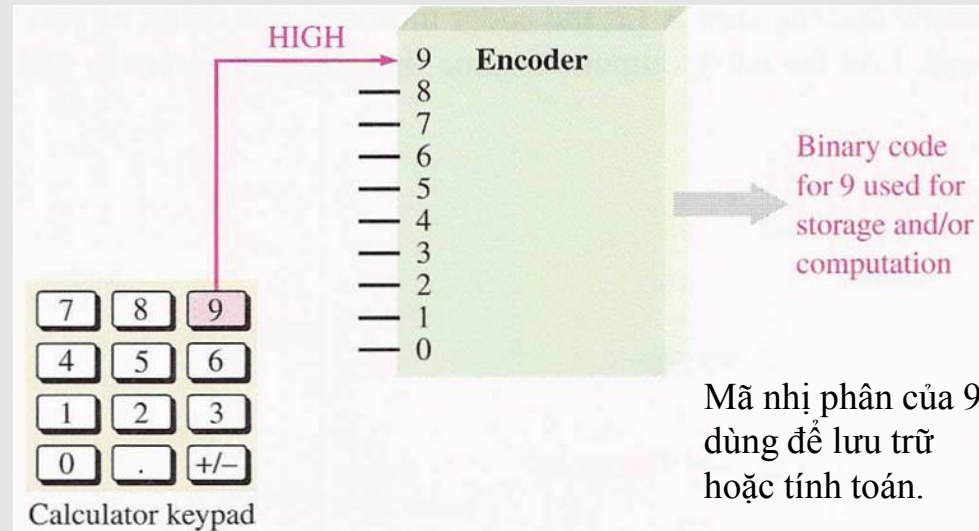


Trừ, nhân: xem các chương sau.



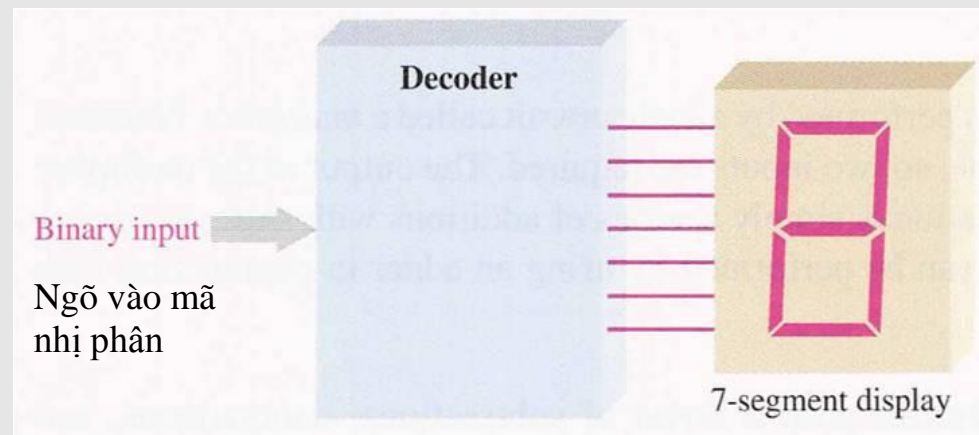
## 1.4 Tổng quan về hàm logic cơ bản

- Hàm biến đổi mã  
Mã hóa  
(encoder)



Mã nhị phân của 9 dùng để lưu trữ hoặc tính toán.

Giải mã  
(Decoder)



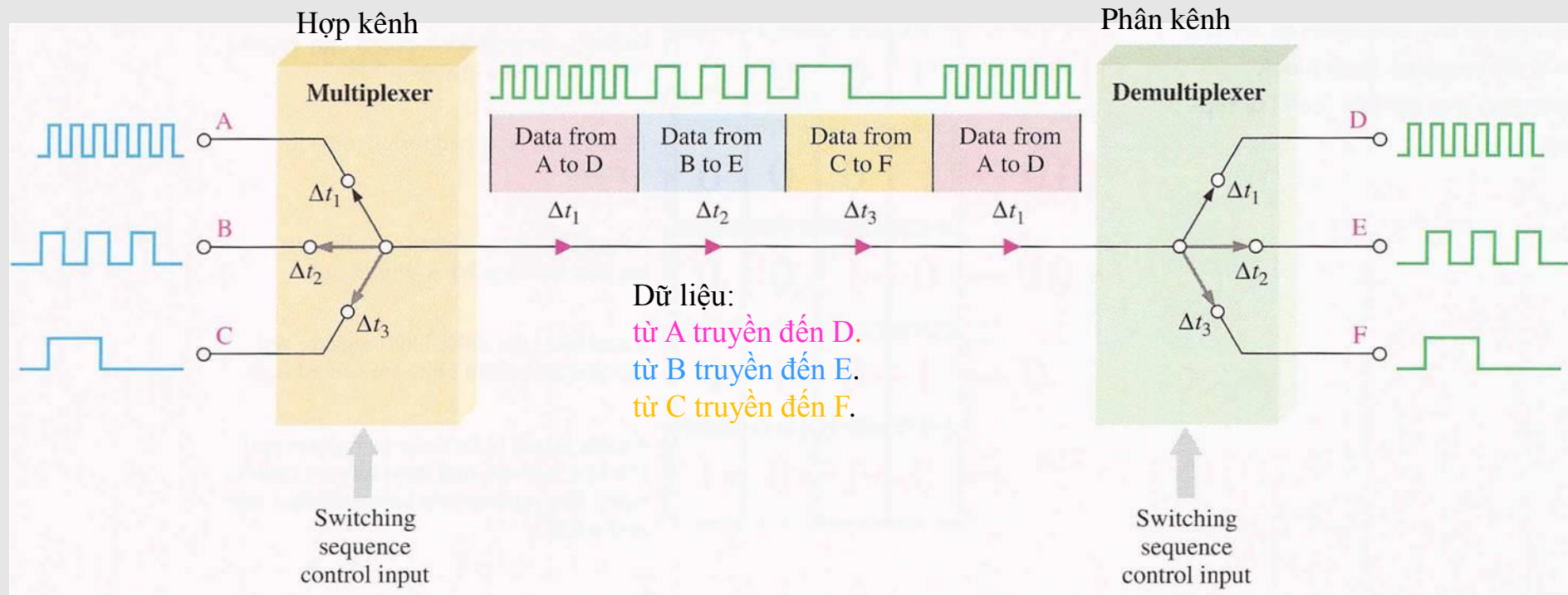
Đèn hiển thị 7-đoạn





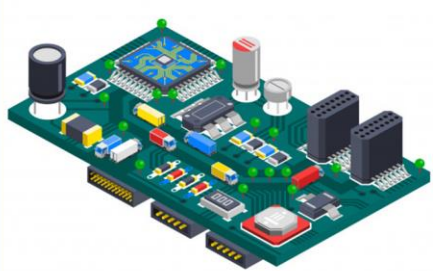
## 1.4 Tổng quan về hàm logic cơ bản

- Hàm lựa chọn dữ liệu



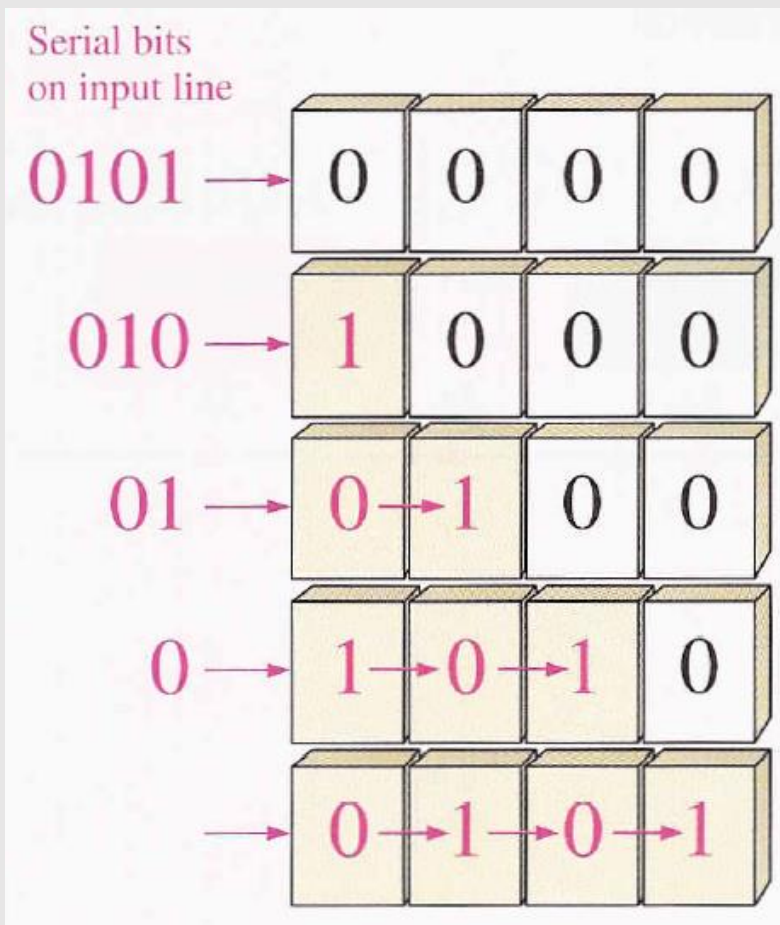
Ngõ vào điều khiển  
trình tự chuyển mạch

Ngõ vào điều khiển  
trình tự chuyển mạch



## 1.4 Tổng quan về hàm logic cơ bản

### Lưu trữ: flipflop, thanh ghi, bộ nhớ



Ban đầu, thanh ghi chỉ chứa dữ liệu không hợp lệ hoặc bằng 0 như hình vẽ.

Bit đầu tiên (1) được dịch nối tiếp vào thanh ghi.

Bit thứ hai (0) được dịch nối tiếp vào thanh ghi và bit đầu tiên được dịch phải.

Bit thứ ba (1) được dịch nối tiếp vào thanh ghi, bit đầu tiên và bit thứ hai được dịch phải.

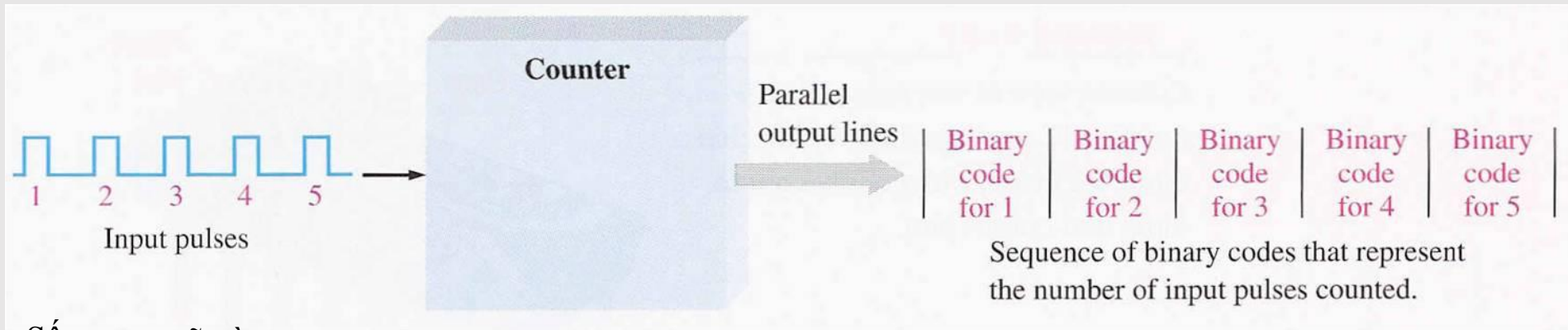
Bit thứ tư (0) được dịch nối tiếp vào thanh ghi, bit đầu tiên, bit thứ hai và bit thứ ba được dịch phải. Thanh ghi lúc này chứa đủ 4-bit.

Thanh ghi dịch bit



## 1.4 Tổng quan về hàm logic cơ bản

- **Đếm**



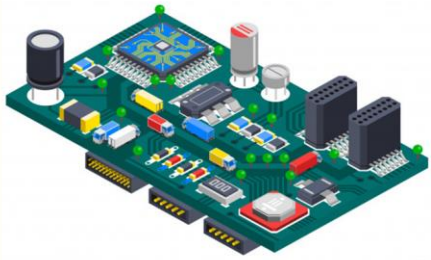
Số xung ngõ vào

Trình tự các mã nhị phân biểu diễn số xung ngõ vào được đếm.



## 2. Hệ thống số và mã

- 2.1 Số thập phân
- 2.2 Số nhị phân
- 2.3 Bù-1 và bù-2
- 2.4 Biến đổi thập phân – nhị phân
- 2.5 Phép toán số nhị phân
- 2.6 Số nhị phân có dấu
- 2.7 Phép toán số nhị phân có dấu
- 2.8 Số hexa (thập lục phân)
- 2.9 Số octal (bát phân)
- 2.10 Số BCD (thập phân mã hóa nhị phân)
- 2.11 Các mã khác



## 2.1 Số thập phân (decimal number)

- Hệ thống số thập phân có 10 ký tự số (10 digit) phân biệt: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
- Hệ thống số thập phân có số cơ sở (base) là 10, còn gọi là cơ số (radix) 10.
- Giá trị của một số được xác định bởi vị trí của số này trong con số

586.23  
↓ ↓ ↓ ↓ ↓  
Vị trí i: 2 1 0 -1 -2

5, 8, 6 là các digit (còn gọi là decade) của phần nguyên; 2, 3 là các digit của phần phân (lẻ).





## 2.1 Số thập phân

Digit 5 ở vị trí 2, trọng số là  $10^2$ . Do vậy digit 5 có giá trị là  $5 \times 10^2$ .

$$\begin{aligned} 568.23 &= (5 \times 10^2) + (6 \times 10^1) + (8 \times 10^0) + (2 \times 10^{-1}) + (3 \times 10^{-2}) \\ &= (5 \times 100) + (6 \times 10) + (8 \times 1) + (2 \times 0.1) + (3 \times 0.01) \\ &= \mathbf{500} + \mathbf{60} + \mathbf{8} + \mathbf{0.2} + \mathbf{0.03} \end{aligned}$$

- Cơ số (radix) lũy thừa vị trí ( $10^i$ ) được gọi là **trọng số** (weight) của digit.
- Ở thí dụ trên, ta đã biểu diễn số theo tổng giá trị của từng digit.



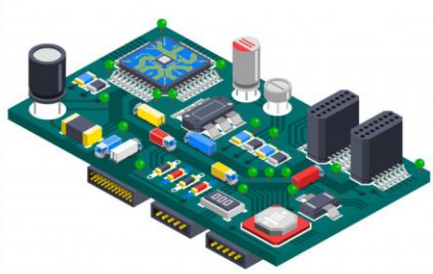


## 2.2 Số nhị phân (binary number)

- Hệ thống số nhị phân có 2 ký tự số (2 digit) phân biệt: 0, 1.
- Hệ thống số nhị phân có số cơ sở (base) là 2, còn gọi là **cơ số (radix) 2**.
- Giá trị của một digit được xác định bởi **vị trí** của digit này trong con số

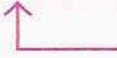
101.011  
↓ ↓ ↓ ↓ ↓  
Vị trí i: 2 1 0 -1 -2 -3

1, 0, 1 là các digit (còn gọi là **bit**) của phần nguyên; 0, 1, 1 là các bit của phần phân (lẻ).



## 2.2 Số nhị phân

$$2^{n-1} \cdot \dots \cdot 2^3 2^2 2^1 2^0 \cdot 2^{-1} 2^{-2} \cdot \dots \cdot 2^{-n}$$

 **Binary point**  
(dấu chấm nhị phân)

Cơ số (radix) lũy thừa vị trí ( $2^i$ ) được gọi là **trọng số** (weight) của bit.

POSITIVE POWERS OF TWO (WHOLE NUMBERS)								
$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
256	128	64	32	16	8	4	2	1

NEGATIVE POWERS OF TWO (FRACTIONAL NUMBER)					
$2^{-1}$	$2^{-2}$	$2^{-3}$	$2^{-4}$	$2^{-5}$	$2^{-6}$
1/2	1/4	1/8	1/16	1/32	1/64
0.5	0.25	0.125	0.0625	0.03125	0.015625



# 2.2 Số nhị phân

Bảng lũy thừa của 2

n	2 <sup>n</sup>	n	2 <sup>n</sup>	n	2 <sup>n</sup>
0	1	8	256	16	65,536
1	2	9	512	17	131,072
2	4	10	1,024	18	262,144
3	8	11	2,048	19	524,288
4	16	12	4,096	20	1,048,576
5	32	13	8,192	21	2,097,152
6	64	14	16,384	22	4,194,304
7	128	15	32,768	23	8,388,608

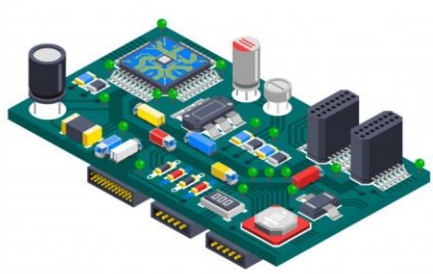


## 2.3 Bù-1 và bù-2

- Bù-1 (1's complement)

Đổi  $0 \rightarrow 1$  và  $1 \rightarrow 0$  đối với từng bit của số nhị phân, ta có số bù-1.

1 0 1 1 0 0 1 0	Binary number	(số nhị phân 8-bit)
↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓		
0 1 0 0 1 1 0 1	1's complement	(số bù-1)



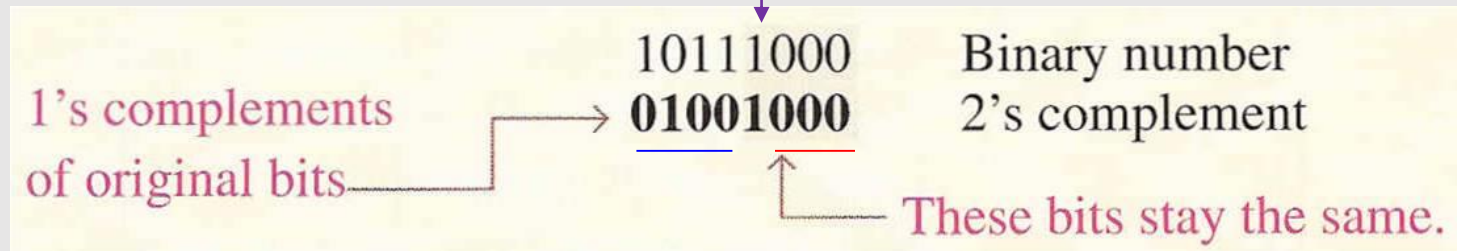
## 2.3 Bù-1 và bù-2

- Bù-2 (2's complement)

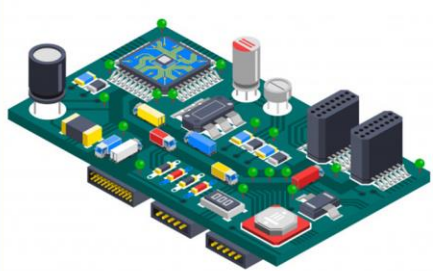
$$2\text{'s complement} = (1\text{'s complement}) + 1$$

10110010	Binary number
01001101	1's complement
$\begin{array}{r} + \phantom{000000} 1 \\ \hline \end{array}$	Add 1
<b>01001110</b>	2's complement

Bit 1 có trọng số thấp nhất



Đổi  $0 \rightarrow 1$  và  $1 \rightarrow 0$  tất cả các bit ở bên trái của bit 1 có trọng số thấp nhất, ta sẽ có số bù-2.



## 2.4 Biến đổi thập phân – nhị phân

- Biến đổi nhị phân-thập phân (biểu diễn số)

$$2^n \dots 2^3 2^2 2^1 2^0 \cdot 2^{-1} 2^{-2} \dots 2^{-n}$$

↑ Binary point

Phần  
nguyên

$$\begin{aligned} \text{Weight: } & 2^6 \ 2^5 \ 2^4 \ 2^3 \ 2^2 \ 2^1 \ 2^0 \\ \text{Binary number: } & 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \\ 1101101 = & 2^6 + 2^5 + 2^3 + 2^2 + 2^0 \\ = & 64 + 32 + 8 + 4 + 1 = \mathbf{109} \end{aligned}$$

Phần  
phân  
(lẻ)

$$\begin{aligned} \text{Weight: } & 2^{-1} \ 2^{-2} \ 2^{-3} \ 2^{-4} \\ \text{Binary number: } & 0 \ . \ 1 \ 0 \ 1 \ 1 \\ 0.1011 = & 2^{-1} + 2^{-3} + 2^{-4} \\ = & 0.5 + 0.125 + 0.0625 = \mathbf{0.6875} \end{aligned}$$





## 2.4 Biến đổi thập phân – nhị phân

- **Biến đổi thập phân-nhị phân**

Phương pháp tổng các trọng số

Ví dụ:

$$(a) \ 12 = 8 + 4 = 2^3 + 2^2 \quad \rightarrow 1101$$

$$(b) \ 25 \qquad (c) \ 58 \qquad (d) \ 82$$

$$(b) \ 25 = 16 + 8 + 1 = 2^4 + 2^3 + 2^0 \quad \rightarrow 11001$$

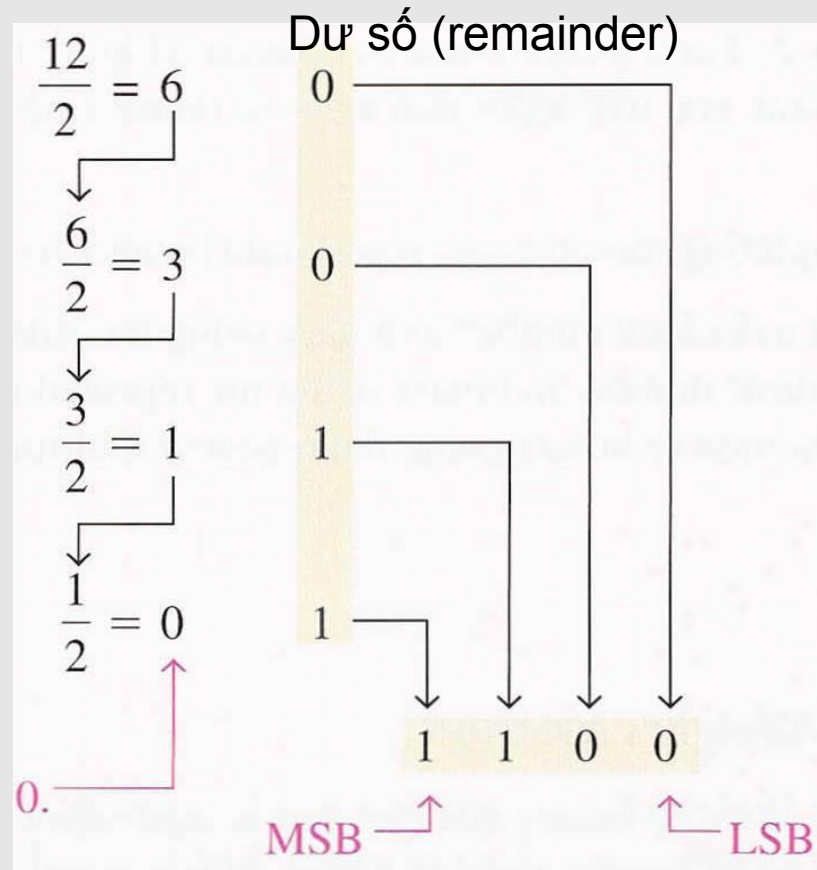
$$(c) \ 58 = 32 + 16 + 8 + 2 = 2^5 + 2^4 + 2^3 + 2^1 \quad \rightarrow 111010$$

$$(d) \ 82 = 64 + 16 + 2 = 2^6 + 2^4 + 2^1 \quad \rightarrow 1010010$$



## 2.3 Biến đổi thập phân – nhị phân

Phương pháp lặp lại phép chia 2 (chia cho cơ số)



Dừng khi thương số bằng 0

0.

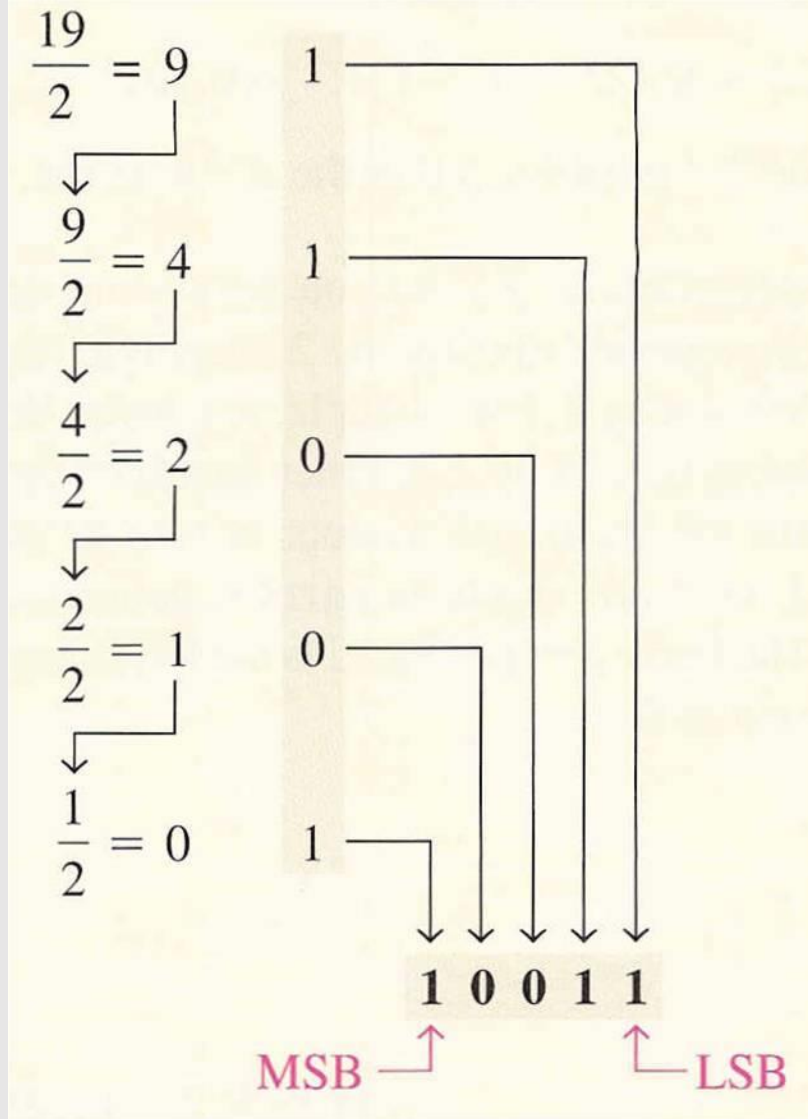
Most significant bit

Least significant bit

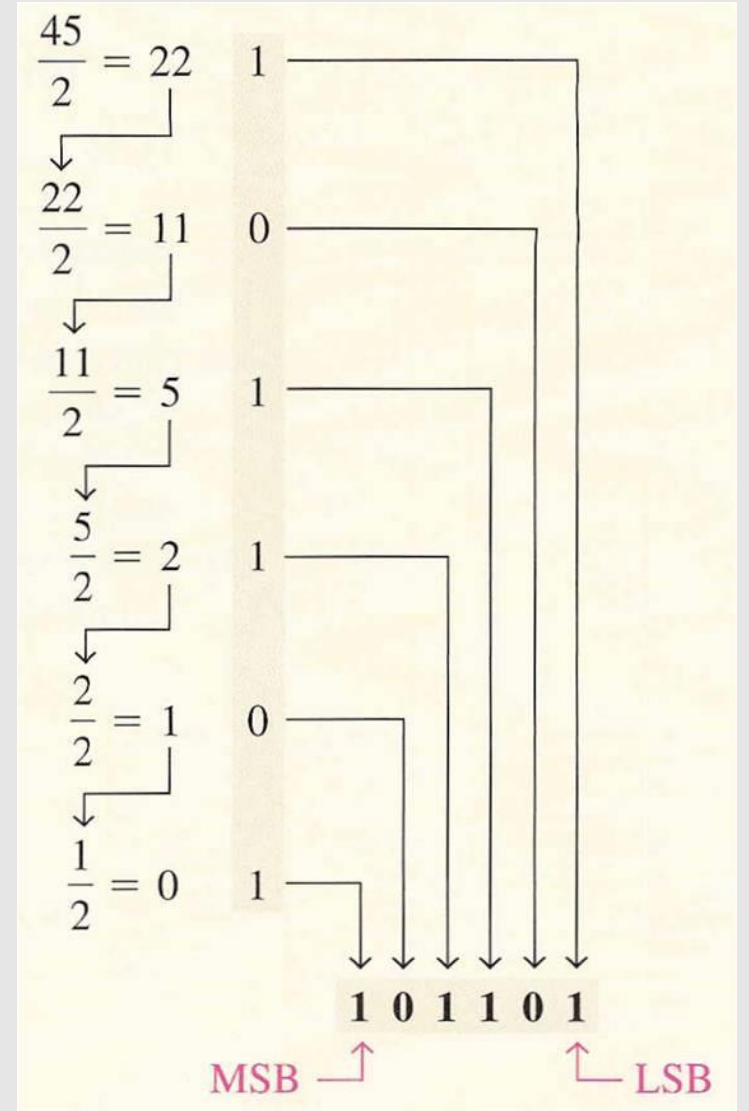


## 2.4 Biến đổi thập phân – nhị phân

19



45

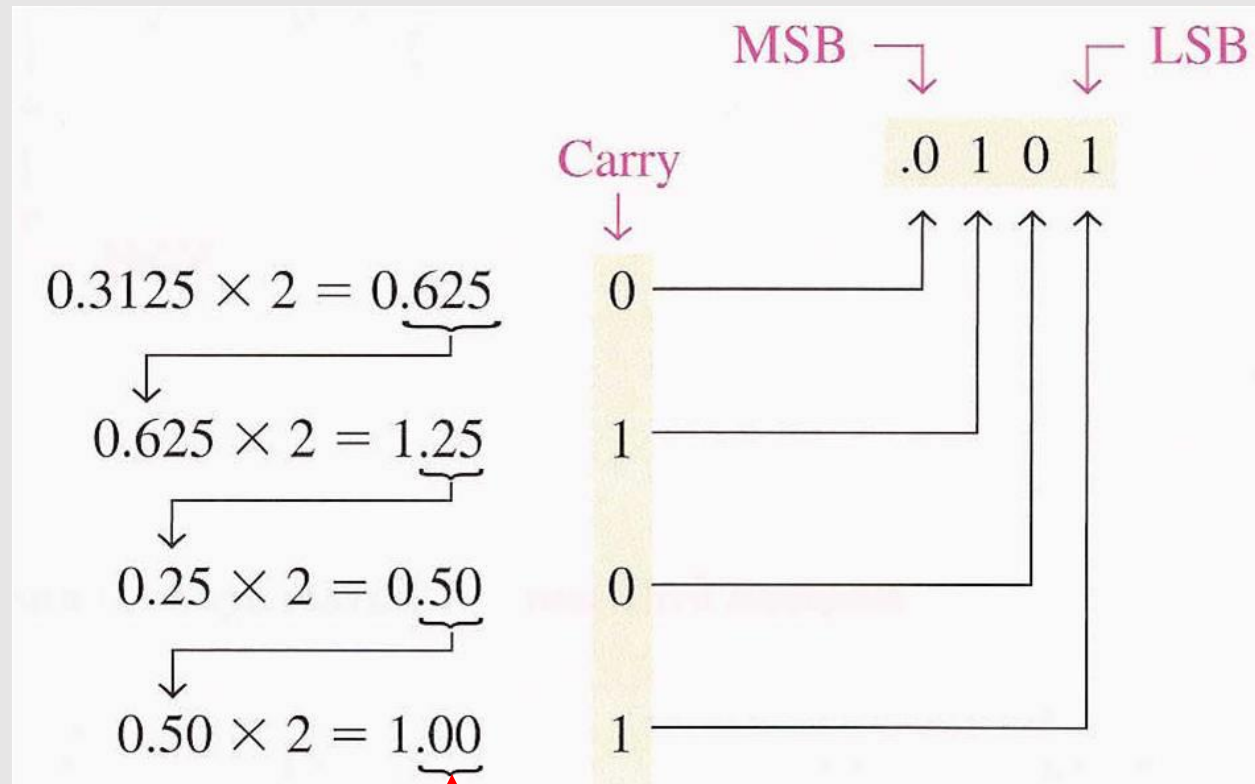




## 2.4 Biến đổi thập phân – nhị phân

- Biến đổi phần phân

$$0.3125 = 0.25 + 0.0625 = 2^{-2} + 2^{-4} = 0.0101$$



Tiếp tục đến sai số chấp nhận được hoặc dừng khi phần phân bằng 0.



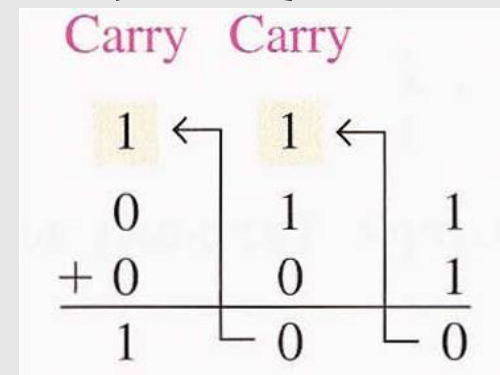
## 2.5 Phép toán số nhị phân

### • Cộng nhị phân

$0 + 0 = 0$	Sum of 0 with a carry of 0
$0 + 1 = 1$	Sum of 1 with a carry of 0
$1 + 0 = 1$	Sum of 1 with a carry of 0
$1 + 1 = 10$	Sum of 0 with a carry of 1

Số (bit) tổng

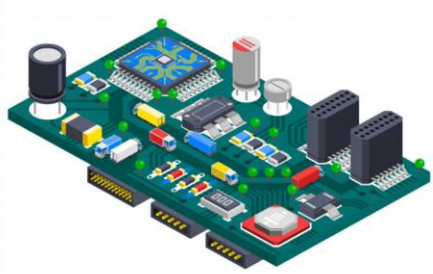
Số (bit) nhớ



Khi số nhớ chuyển đến là 1, ta có tình huống cộng 3 bit:

Carry bits				
Các (bit) nhớ	↓			
1	+ 0 + 0 = 01	Sum of 1 with a carry of 0		
1	+ 1 + 0 = 10	Sum of 0 with a carry of 1		
1	+ 0 + 1 = 10	Sum of 0 with a carry of 1		
1	+ 1 + 1 = 11	Sum of 1 with a carry of 1		





## 2.5 Phép toán số nhị phân

### • Trừ nhị phân

Số (bit) hiệu	Số (bit) mượn
0 - 0 = 0	
1 - 1 = 0	
1 - 0 = 1	
10 - 1 = 1	0 - 1 with a borrow of 1

(a)  $11 - 01$

(b)  $11 - 10$

(a) 
$$\begin{array}{r} 11 \\ - 01 \\ \hline 10 \end{array}$$

$$\begin{array}{r} 3 \\ - 1 \\ \hline 2 \end{array}$$

(b) 
$$\begin{array}{r} 11 \\ - 10 \\ \hline 01 \end{array}$$

$$\begin{array}{r} 3 \\ - 2 \\ \hline 1 \end{array}$$

Left column:

When a 1 is borrowed, a 0 is left, so  $0 - 0 = 0$ .

Middle column:

Borrow 1 from next column to the left, making a 10 in this column, then  $10 - 1 = 1$ .

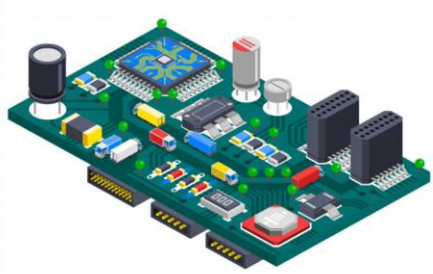
Right column:

$1 - 1 = 0$

$$\begin{array}{r} 0101 \\ - 011 \\ \hline 010 \end{array}$$

$$\begin{array}{r} 110110 \\ - 10111 \\ \hline 101011 \end{array}$$





## 2.5 Phép toán số nhị phân

- Nhân nhị phân

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

$$0 \times a = 0$$

$$1 \times a = a$$

(a)  $11 \times 11$

(b)  $101 \times 111$

(a)

$$\begin{array}{r} 11 \\ \times 11 \\ \hline 11 \\ + 11 \\ \hline 1001 \end{array}$$

Partial products {

$$\begin{array}{r} 3 \\ \times 3 \\ \hline 9 \end{array}$$

(b)

$$\begin{array}{r} 111 \\ \times 101 \\ \hline 111 \\ 000 \\ + 111 \\ \hline 10011 \end{array}$$

Partial products {

Các tích riêng phần



## 2.5 Phép toán số nhị phân

- Phép chia

Phép chia nhị phân cũng tương tự như phép chia trong hệ thập phân

$$\begin{array}{r|l}
 159 & 12 \\
 - 12 & \text{----} \\
 \hline
 39 & \\
 - 36 & \\
 \hline
 3 & 
 \end{array}$$

$$\begin{array}{r|l}
 10011111 & 1100 \\
 - 1100 & \text{----} \\
 \hline
 1111 & \\
 - 1100 & \\
 \hline
 1111 & \\
 - 1100 & \\
 \hline
 11 & 
 \end{array}$$

$$100101 / 1001$$

$$\begin{array}{r|l}
 100101 & 1001 \\
 - 1001 & \\
 \hline
 00 & \\
 - 0 & \\
 \hline
 01 & \\
 - 0 & \\
 \hline
 1 & 
 \end{array}$$

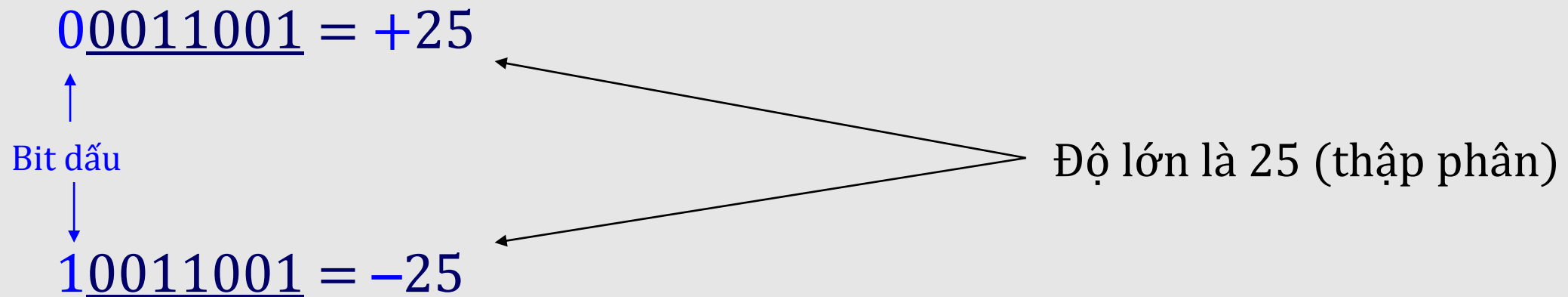


## 2.6 Số có dấu (signed number)

- Bit dấu (sign bit)

Trong số nhị phân có dấu, bit cực trái (bit có trọng số lớn nhất) là bit dấu.  
Bit dấu là 0 chỉ số dương, còn bit dấu là 1 chỉ số âm.

- Dạng dấu-độ lớn (sign-magnitude form)



Ở dạng dấu-độ lớn, số âm có cùng các bit độ lớn với số dương tương ứng, chỉ có bit dấu là 1 thay vì là 0.



## 2.6 Số có dấu

- **Dạng bù-1 (1's complement form)**

$$\underline{00011001} = +25$$

Lấy bù-1

Ở dạng bù-1, số âm là bù-1 của số dương tương ứng.

$$\underline{11100110} = -25$$

- **Dạng bù-2 (2's complement form)**

$$\underline{00011001} = +25$$

Lấy bù-2

Ở dạng bù-2, số âm là bù-2 của số dương tương ứng.

$$\underline{11100111} = -25$$



## 2.6 Số có dấu

- **Thí dụ:**

Biểu diễn số thập phân  $-39$  thành số nhị phân có dấu 8-bit dạng: dấu-độ lớn, bù-1 và bù-2.

**Giải:**

Số dương  $+39$  có dạng nhị phân 8-bit là: **00100111**.

a) Dạng dấu-độ lớn

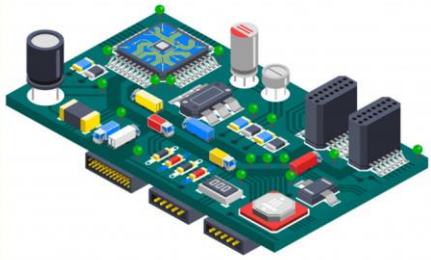
**10100111**

b) Dạng bù-1

**11011000**

c) Dạng bù-2

**11011001**



## 2.6 Số có dấu

- Giá trị thập phân của số có dấu

Số dương:

Các số có dấu dạng dấu-độ lớn, bù-1 và bù-2 đều có cùng giá trị thập phân khi chúng là số dương.

Số âm dạng dấu-độ lớn:

10010101

Số dương tương ứng là 00010101

$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
0	0	1	0	1	0	1

có giá trị thập phân là  $16 + 4 + 1 = 21$ .

Vậy 10010101 có giá trị thập phân là  $-21$ .





## 2.6 Số có dấu

### Số âm dạng bù-1

11101000

Số dương tương ứng là 01101000

↓ Lấy bù-1  
0010111

0010111 có giá trị là  $16 + 4 + 2 + 1 = +23$

Vậy 11101000 có giá trị là  $-23$ .

### Cách khác:

$-2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
1	1	1	0	1	0	0	0

Tổng các trọng số cộng với 1:

$$-128 + 64 + 32 + 8 + 1 = -23$$



## 2.6 Số có dấu

### Số âm dạng bù-2

10101010

Số dương tương ứng là 00101010

Lấy bù-2

1010110

1010110 có giá trị là  $64 + 16 + 4 + 2 = +86$

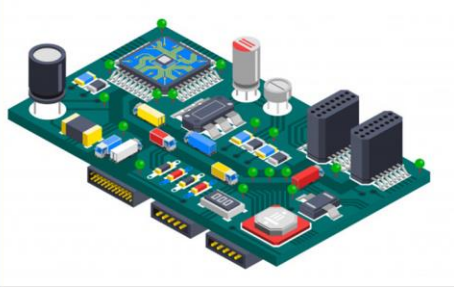
Vậy 10101010 có giá trị là  $-86$ .

### Cách khác:

$-2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
1	0	1	0	1	0	1	0

Tổng các trọng số:

$$-128 + 32 + 8 + 2 = -86$$



## 2.6 Số có dấu

- Số có dấu 4-bit

Nhị phân 4-bit	Không dấu	Thập phân Dấu-độ lớn	Bù-1	Bù-2
0000	0	+0	+0	+0
0001	1	+1	+1	+1
0010	2	+2	+2	+2
0011	3	+3	+3	+3
0100	4	+4	+4	+4
0101	5	+5	+5	+5
0110	6	+6	+6	+6
0111	7	+7	+7	+7
1000	8	-0	-7	-8
1001	9	-1	-6	-7
1010	10	-2	-5	-6
1011	11	-3	-4	-5
1100	12	-4	-3	-4
1101	13	-5	-2	-3
1110	14	-6	-1	-2
1111	15	-7	-0	-1



## 2.6 Số có dấu

- Tầm giá trị của số có dấu N-bit

Dấu – độ lớn:

$$-(2^{N-1} - 1) \text{ đến } +(2^{N-1} - 1)$$

Bù-1:

$$-(2^{N-1} - 1) \text{ đến } +(2^{N-1} - 1)$$

Bù-2:

$$-(2^{N-1}) \text{ đến } +(2^{N-1} - 1)$$