



Kiểm tra 15 phút

- Thực hiện phép chuyển đổi giữa các hệ thống số

Decimal	Binary	Octal	Hexadecimal
5.3			
	1011011		
		7.12	
			1A.F

COMPUTER ENGINEERING



Kiểm tra 15 phút

- Thực hiện phép chuyển đổi giữa các hệ thống số

Decimal	Binary	Octal	Hexadecimal
5.3	101.01001100110	5.2314	5.4CC
91	1011011	133	5B
7.15625	111.00101	7.12	7.28
26.9375	11010.1111	32.74	1A.F

COMPUTER ENGINEERING



Kiểm tra 15 phút

Decimal	Binary	Octal	Hexadecimal
73.4	*		
	110.11011		

* Lưu ý: số nhị phân có 8bit thập phân sau dấu .

Biểu diễn các số sau dưới dạng bù 2, 8bit (gồm cả bit dấu):

17, -26

Sau đó thực hiện phép toán cộng 17 và -26



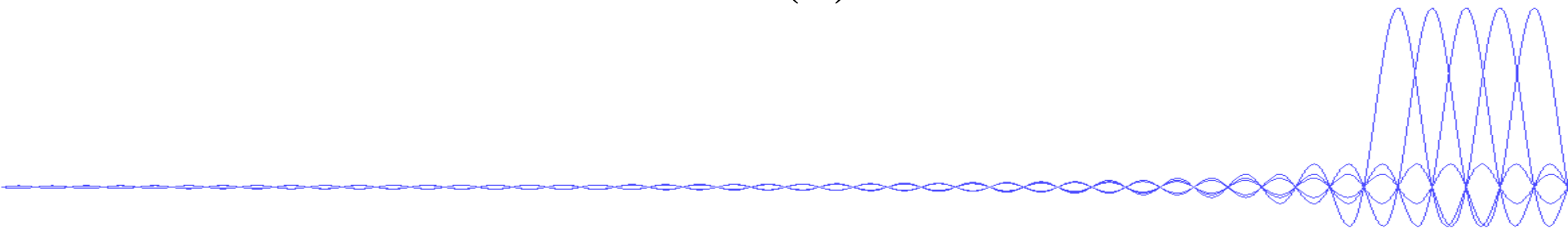
COMPUTER ENGINEERING



UIT
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

NHẬP MÔN MẠCH SỐ

CHƯƠNG 2: CÁC DẠNG BIỂU DIỄN SỐ (tt)





Nội dung

- Tổng quan
- Các hệ thống số
- Chuyển đổi giữa các hệ thống số
- Biểu diễn số phân số thập phân dưới dạng nhị phân
- Các phép tính số nhị phân không dấu
 - Phép cộng
 - Phép nhân
 - Phép trừ
- Biểu diễn số nhị phân có dấu
- Biểu diễn các loại số khác



Phép cộng

■ Cộng 2 số nhị phân 1-bit

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	10



Phép cộng

■ Phép cộng 2 số nhị phân không dấu

a)

$$\begin{array}{r} 11 \quad (3) \\ +110 \quad (6) \\ \hline 1001 \quad (9) \end{array}$$

b)

$$\begin{array}{r} 11.011 \quad (3.375) \\ +10.110 \quad (2.750) \\ \hline 110.001 \quad (6.125) \end{array}$$

COMPUTER ENGINEERING



Phép nhân

■ Nhân 2 số nhị phân 1-bit

A	B	$A * B$
0	0	0
0	1	0
1	0	0
1	1	1



Phép nhân

■ Phép nhân 2 số nhị phân không dấu

$$\begin{array}{r} 1110 \\ \times 1011 \\ \hline 1110 \\ 1110 \\ 0000 \\ 1110 \\ \hline 10011010 \end{array}$$



Phép trừ

- Quy tắc thực hiện phép trừ như sau:

$$0 - 0 = 0$$

$$1 - 1 = 0$$

$$1 - 0 = 1$$

$$[1]0 - 1 = 1 \text{ Mượn 1}$$

- VD: Thực hiện phép trừ 2 số nhị phân 5 bits:
00111 từ 10101

$$\begin{array}{r} 10101 \\ - 00111 \\ \hline 01110 \end{array} \quad \begin{array}{r} 21 \\ - 7 \\ \hline 14 \end{array}$$



Nội dung

- Tổng quan
- Các hệ thống số
- Chuyển đổi giữa các hệ thống số
- Biểu diễn số phân số thập phân dưới dạng nhị phân
- Các phép tính số nhị phân không dấu
- Biểu diễn số nhị phân có dấu
 - Số dấu và độ lớn
 - Số bù 1
 - Số bù 2
 - Phép cộng, phép trừ số bù 2
 - Hiện tượng tràn số học



Biểu diễn số có dấu

- Số dương (+) và Số âm (-)
- Sử dụng thêm 1 bit (sign bit) để thể hiện dấu của số:
 - 0: dương
 - 1: âm
- Bit thể hiện dấu nằm ở ngoài cùng bên trái của số



Biểu diễn số có dấu

■ Có 3 dạng phổ biến để biểu diễn số có dấu:

□ Dạng số “dấu và độ lớn”

□ Dạng số “bù 1”

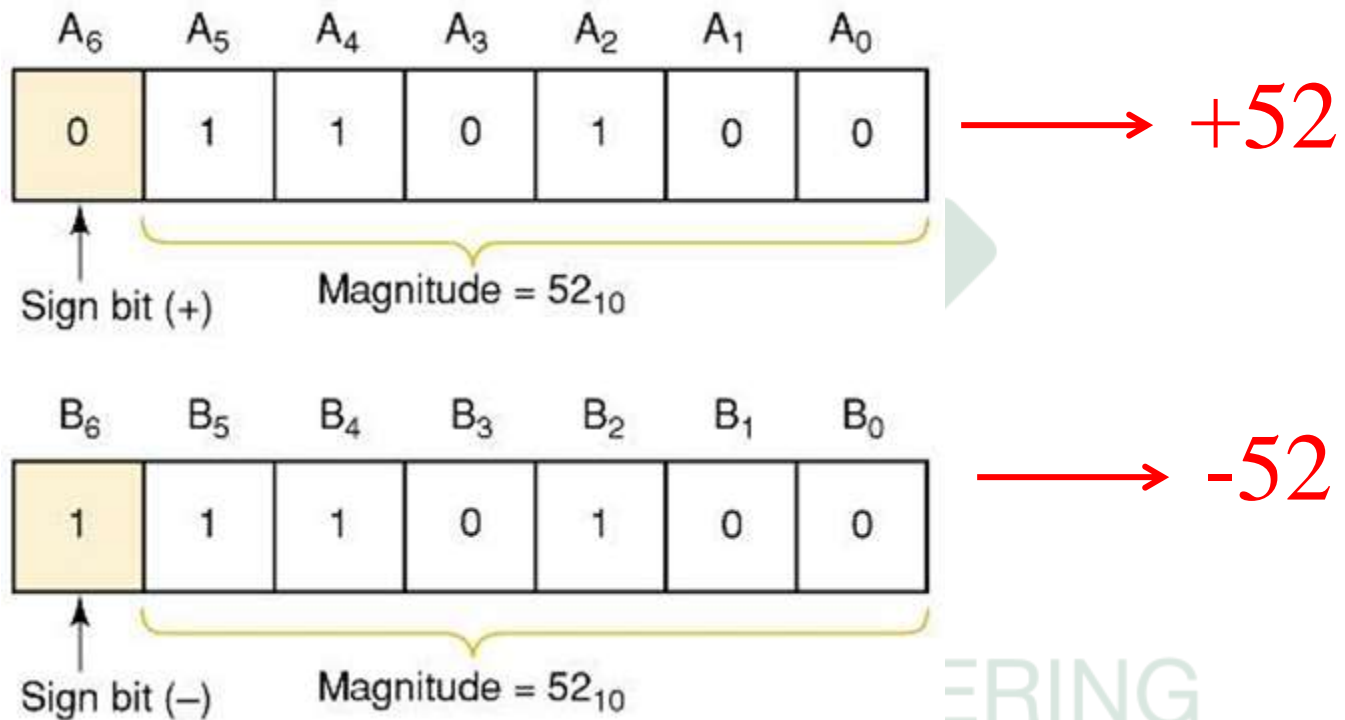
□ Dạng số “bù 2”

COMPUTER ENGINEERING



Dạng số “dấu và độ lớn”

■ Ví dụ: biểu diễn 1 số 6 bits có dấu



Giá trị số dấu và độ lớn

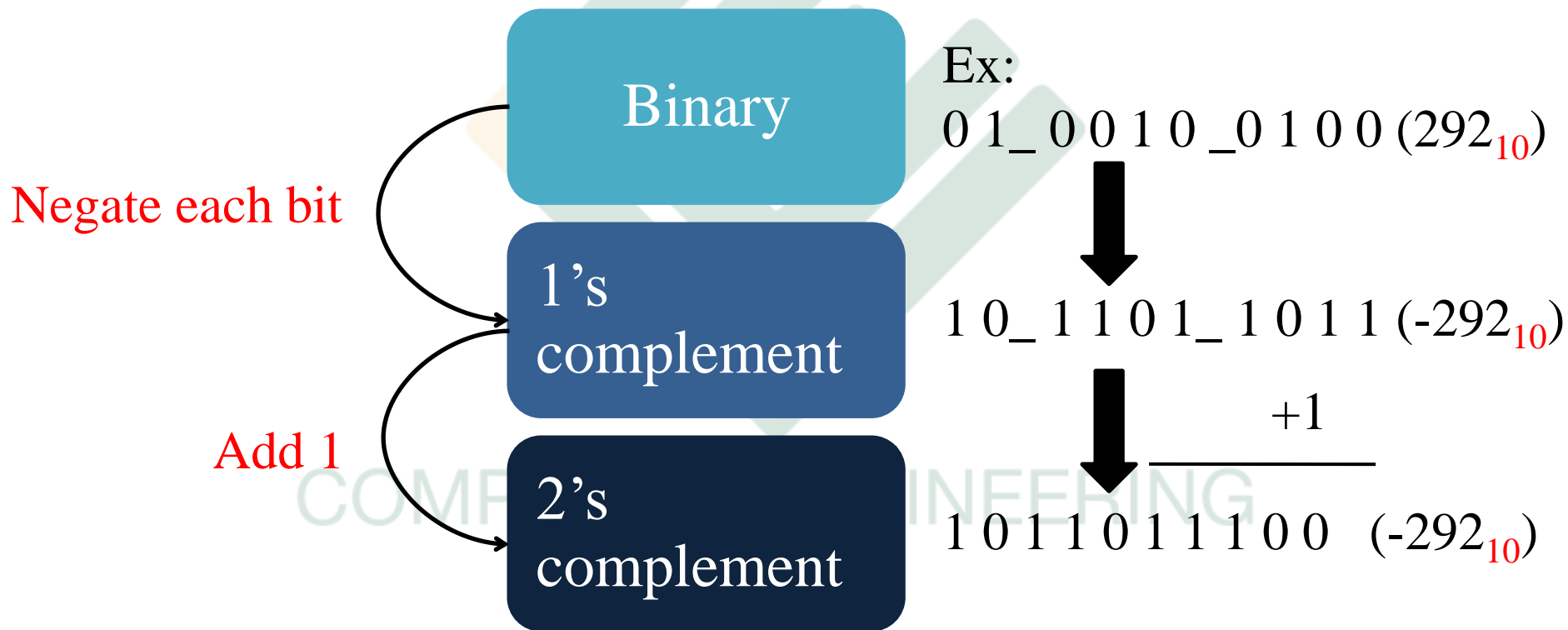
$$= (-1)^{b_{n-1}} \times \sum_{i=0}^{n-2} b_i 2^i$$

n: số bit biểu diễn số bù 2
b: giá trị của bit (0, 1)



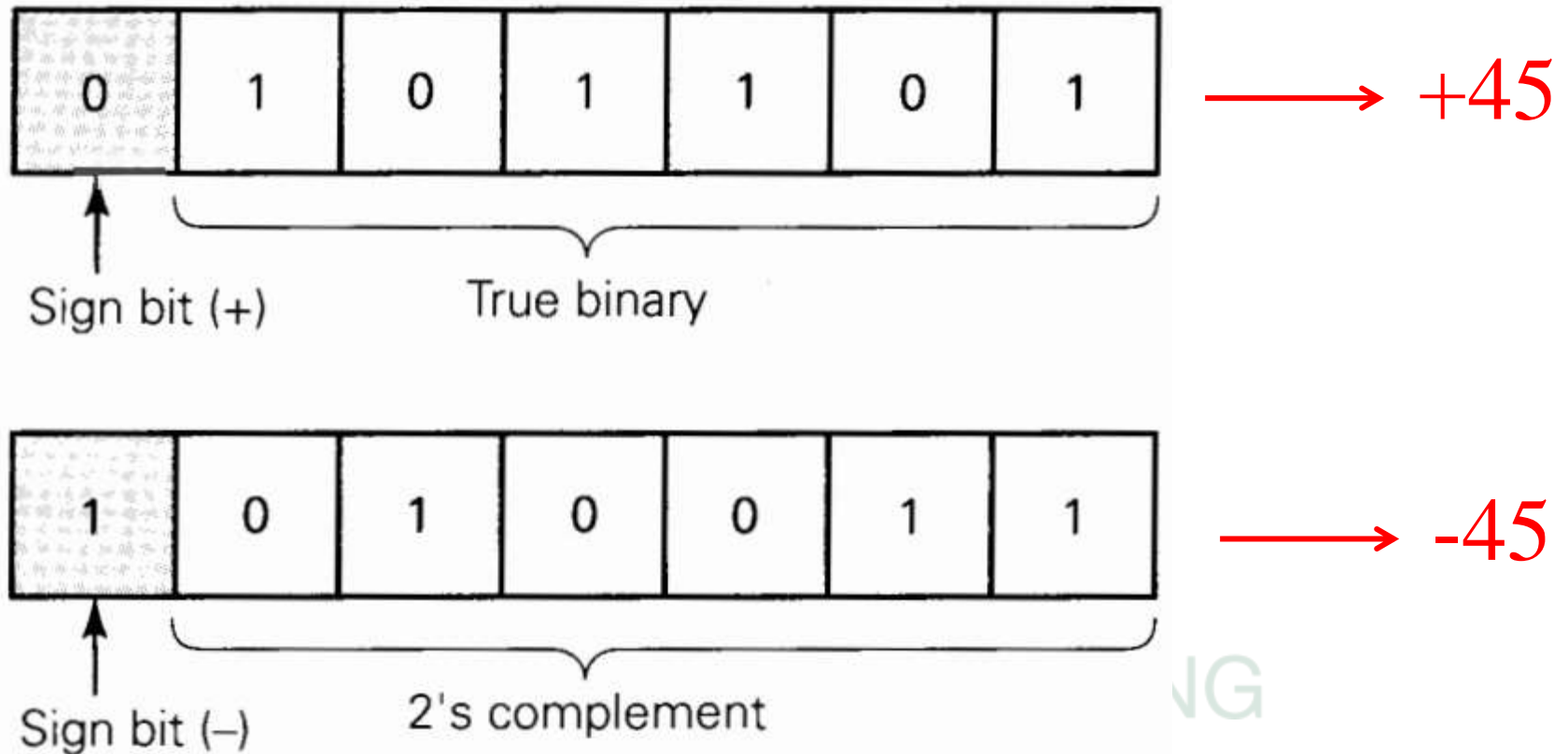
Dạng số “bù 1” và “bù 2”

- Phương pháp tìm số âm của một số dưới dạng số “bù 1” và dưới dạng số “bù 2”:





Biểu diễn số có dấu dưới dạng số bù 2



$$\text{Giá trị số bù 2} = -b_{n-1}2^{n-1} + \sum_{i=0}^{n-2} b_i 2^i$$

n: số bit biểu diễn số bù 2
b: giá trị của bit (0, 1)



Ví dụ

■ Biểu diễn số có dấu áp dụng phương pháp dạng số bù 2

(a) +13

(b) -9

(c) -2

(d) -8



COMPUTER ENGINEERING



Tầm trị biểu diễn

■ Cho số có dấu n bit

$b_3b_2b_1b_0$	Sign-magnitude	1's complement	2's complement
0111	+7	+7	+7
0110	+6	+6	+6
0101	+5	+5	+5
0100	+4	+4	+4
0011	+3	+3	+3
0010	+2	+2	+2
0001	+1	+1	+1
0000	+0	+0	+0
1000	-0	-7	-8
1001	-1	-6	-7
1010	-2	-5	-6
1011	-3	-4	-5
1100	-4	-3	-4
1101	-5	-2	-3
1110	-6	-1	-2
1111	-7	-0	-1

Số dương được biểu diễn giống nhau ở cả 3 dạng

$$-(2^{n-1}-1) \rightarrow 2^{n-1}-1 \quad -(2^{n-1}-1) \rightarrow 2^{n-1}-1 \quad -2^{n-1} \rightarrow 2^{n-1}-1$$



Phép tính sử dụng số bù 2

■ Tại sao trong máy tính sử dụng số bù 2 để thực hiện các phép toán mà không sử dụng số dấu và độ lớn hoặc bù 1.

□ Ví dụ 1: Dấu và độ lớn: $(-1) + 5 = 4$ (dương)

$$\begin{array}{r} + 1001 \\ 0101 \\ \hline 1110 \end{array} \text{ (âm)} \rightarrow \text{Sai}$$

□ Ví dụ 2: Bù 1: $(-5) + (-2) = -7$ (âm)

$$\begin{array}{r} + 1010 \text{ (-5)} \\ 1101 \text{ (-2)} \\ \hline 10111 \text{ (-8)} \end{array} \rightarrow \text{Sai}$$



Phép cộng số bù 2

■ Thực hiện như phép cộng số bù 2

- ☐ Bit dấu được xử lý dựa theo cách tương tự như các bit độ lớn
- ☐ Bit nhớ ở vị trí cuối cùng sẽ được loại bỏ
- ☐ **Kết quả của phép cộng sử dụng số bù 2 luôn đúng**

COMPUTER ENGINEERING



Ví dụ

$$\begin{array}{rcl} +9 \rightarrow & 0 & 1001 \quad (\text{augend}) \\ +4 \rightarrow & 0 & 0100 \quad (\text{addend}) \\ \hline & 0 & 1101 \quad (\text{sum} = +13) \end{array}$$

↑ sign bits

$$\begin{array}{rcl} -9 \rightarrow & 10111 \\ +4 \rightarrow & 00100 \\ \hline & 11011 \quad (\text{sum} = -5) \end{array}$$

↑ negative sign bit



Ví dụ

$-9 \rightarrow 10111$

$-4 \rightarrow 11100$

$\begin{array}{r} 1 \\ \uparrow \\ 10011 \end{array}$

sign bit

This carry is disregarded; the result is 10011 (sum = -13).

- Thực hiện phép cộng 2 số thập phân: +9 và -9?

COMPUTER ENGINEERING



Phép trừ số bù 2

- Trong ví dụ $4 + (-9)$, phép cộng trong hệ thống số bù 2 thực chất là phép trừ

$$\begin{array}{r} -9 \rightarrow 10111 \\ +4 \rightarrow 00100 \\ \hline 11011 \quad (\text{sum} = -5) \\ \uparrow \text{negative sign bit} \end{array}$$

- Quy tắc thực hiện phép trừ trong hệ thống số bù 2:

$$\begin{aligned} -B &= \text{bù 2 của } B \\ A - B &= A + (-B) = A + (\text{bù 2 của } B) \end{aligned}$$



Ví dụ

$$\blacksquare 9 - 4 = ?$$

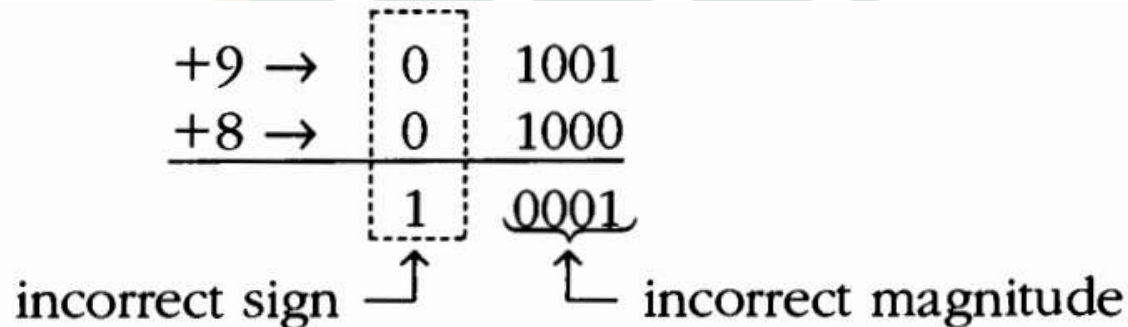


COMPUTER ENGINEERING



Hiện tượng tràn số học (Overflow)

- Tràn: Khi số bit của kết quả vượt quá số bit cho phép
- 1 số có dấu bù 2 n-bit biểu diễn trong tầm: -2^{n-1} đến $+2^{n-1}-1$
- Hiện tượng Overflow luôn cho 1 kết quả sai hoàn toàn



- Một mạch điện riêng biệt được thiết kế ra để phát hiện hiện tượng tràn



Ví dụ

- Số có 4 bit, gồm 3 bit độ lớn và 1 bit dấu

$$\begin{array}{r} (+7) \\ + (+2) \\ \hline (+9) \end{array}$$

$$\begin{array}{r} 0111 \\ + 0010 \\ \hline 1001 \end{array}$$

$c_4 = 0$
 $c_3 = 1$

$$\begin{array}{r} (-7) \\ + (+2) \\ \hline (-5) \end{array}$$

$$\begin{array}{r} 1001 \\ + 0010 \\ \hline 1011 \end{array}$$

$c_4 = 0$
 $c_3 = 0$

$$\begin{array}{r} (+7) \\ + (-2) \\ \hline (+5) \end{array}$$

$$\begin{array}{r} 0111 \\ + 1110 \\ \hline 10101 \end{array}$$

$c_4 = 1$
 $c_3 = 1$

$$\begin{array}{r} (-7) \\ + (-2) \\ \hline (-9) \end{array}$$

$$\begin{array}{r} 1001 \\ + 1110 \\ \hline 10111 \end{array}$$

$c_4 = 1$
 $c_3 = 0$

- Hiện tượng Tràn không xảy ra đối với những phép tính giữa 2 số khác dấu nhau



Nội dung

- Tổng quan
- Các hệ thống số
- Chuyển đổi giữa các hệ thống số
- Biểu diễn số phân số thập phân dưới dạng nhị phân
- Các phép tính số nhị phân không dấu
- Biểu diễn số nhị phân có dấu
- Biểu diễn các loại số khác
 - BCD
 - Số dấu chấm động
 - ASCII



BCD (Binary Coded Decimal)

- Mỗi chữ số của số thập phân được biểu diễn bằng số nhị phân **4 bits** tương ứng

0	1	2	3	4	5	6	7	8	9
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

- Công dụng: Hiển thị số thập phân trên các thiết bị máy tính
- Ex:

$$10_{10} \Rightarrow \text{BCD}$$

$$847_{10} \Rightarrow \text{BCD}$$



Ví dụ

$$137_{10} = 10001001_2 \quad (\text{Số Nhị Phân})$$

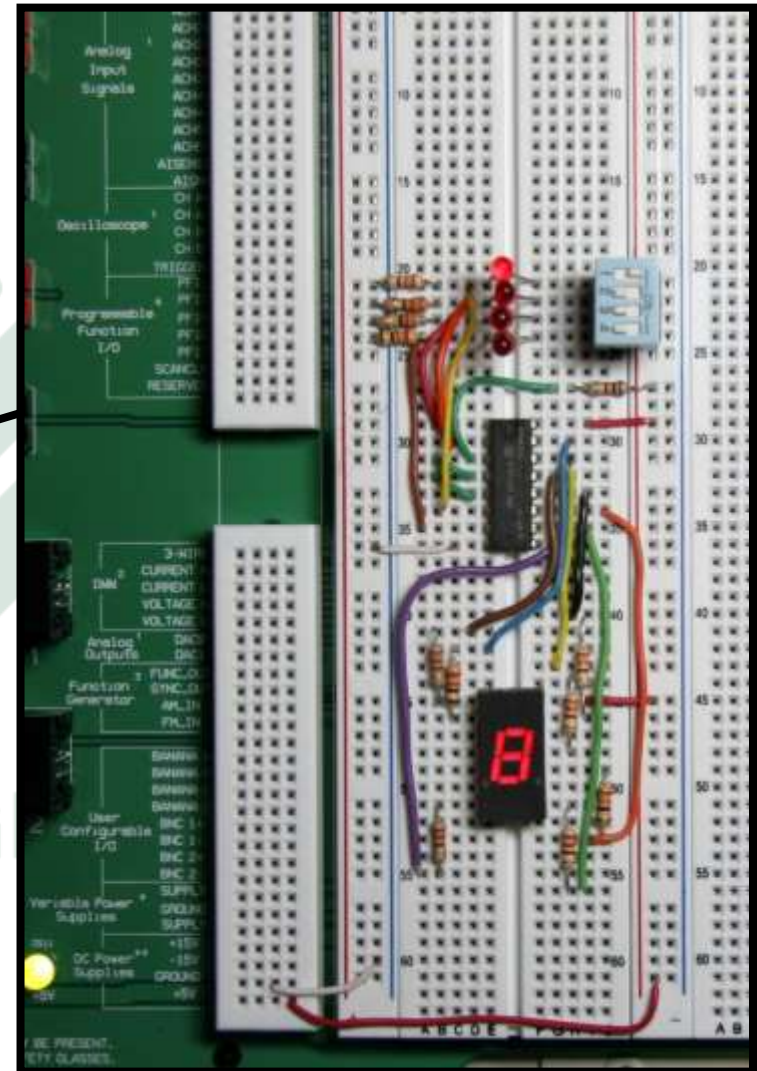
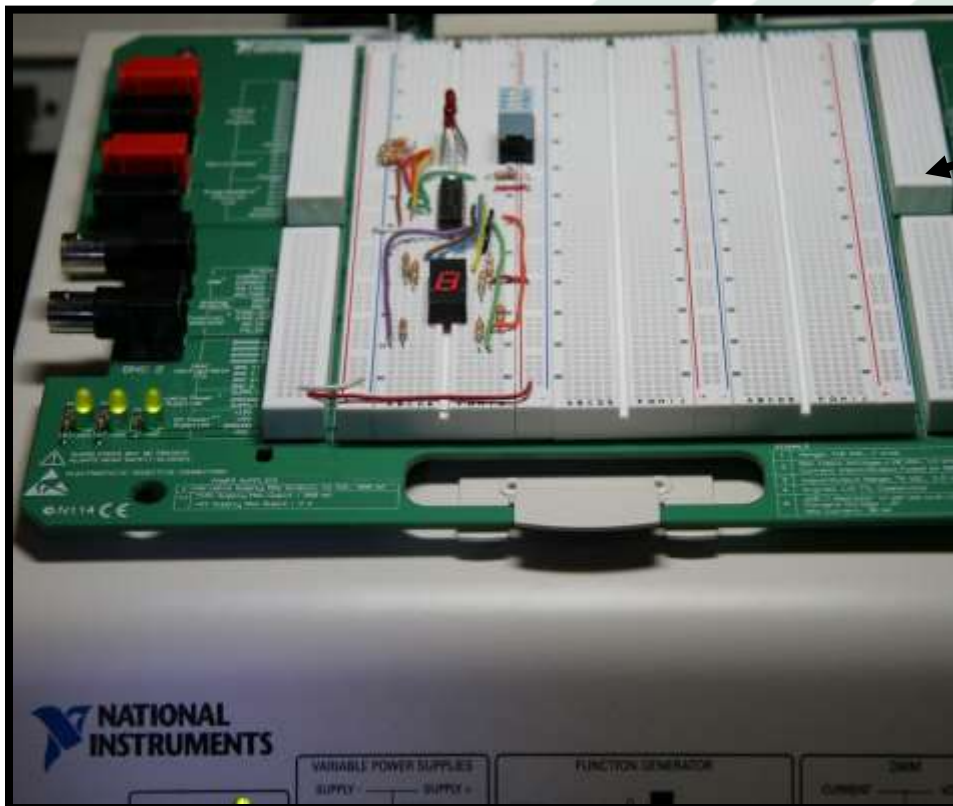
$$137_{10} = 0001_0011_0111 \quad (\text{BCD})$$

- BCD sử dụng nhiều bits hơn nhưng việc chuyển đổi đơn giản hơn



Hiện thị số BCD lên thiết bị điện tử

- Mạch thí nghiệm chuyển đổi từ số thập phân sang số BCD





Số dấu chấm động

Biểu diễn giá trị của tốc độ ánh sáng, c , bằng ký hiệu của số dấu chấm động có độ chính xác đơn ($c = 0.2998 \times 10^9$)

Số Nhị Phân, $c = 0001_0001_1101_1110_1001_0101_1100_0000_2$.

Ký hiệu khoa học,

$c = 1._0001_1101_1110_1001_0101_1100_0000 \times 2^{28}$.

$S = 0$ // số dương

$E = 28 + 127 = 155_{10} = 1001\ 1011_2$. (IEEE 754, bias = 127)

F là 23 bits tiếp theo sau khi bit có giá trị 1 đầu tiên xuất hiện.

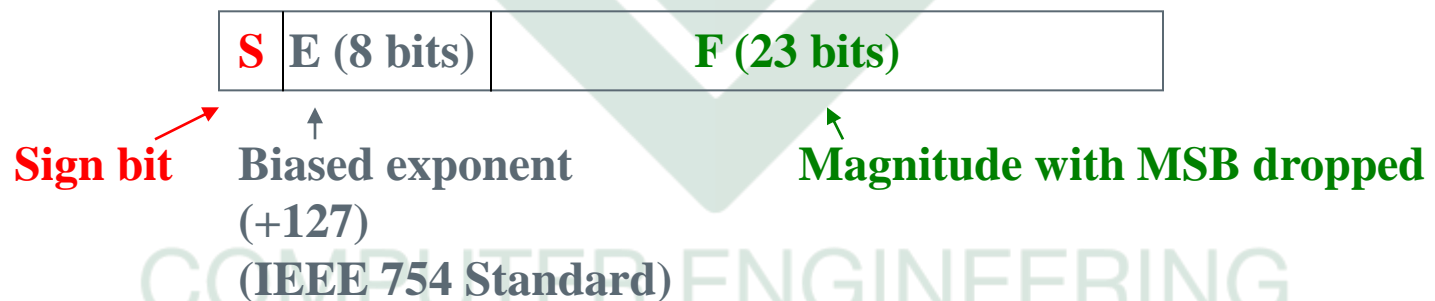
32-bit độ chính xác đơn (phần cứng)

$C = 0\ 10011011 \mid 0001_1101_1110_1001_0101_110$



Số dấu chấm động

- Ký hiệu dấu chấm động có thể biểu diễn cho một số có **giá trị rất lớn** hay **rất nhỏ** bằng cách sử dụng một hình thức ký hiệu khoa học
- Ví dụ minh họa 1 số dấu chấm động 32-bit có độ chính xác đơn.



- Giá trị thập phân = $(-1)^S \times 1.F \times 2^{E-127}$



Ví dụ

■ Số thập phân: -157.625

➔ Số dấu chấm động 32 bit ?

■ Số dấu chấm động 32 bit: 01001101001110000000000000000000

➔ Số thập phân ?

COMPUTER ENGINEERING



ASCII

- ASCII-7 (American Standard Codes for Information Interchange) (7 bit): dùng để biểu diễn 128 ký tự (character) dưới dạng số nhị phân 7 bit
- Ví dụ: Mã ASCII-7 được dùng thể hiện các ký tự từ bàn phím
 - $(1000001)_{\text{ASCII}} = (41\text{H}) = 'A'$
 - $(1000010)_{\text{ASCII}} = (42\text{H}) = 'B'$
 - $(1100001)_{\text{ASCII}} = (61\text{H}) = 'a'$
 - $(1100010)_{\text{ASCII}} = (62\text{H}) = 'b'$
 - $(0110000)_{\text{ASCII}} = (30\text{H}) = '0'$
 - $(0111001)_{\text{ASCII}} = (39\text{H}) = '9'$



Thuật ngữ

Byte	1 byte gồm có 8 bits
Floating-point number	Một số được đại diện dựa trên ký hiệu khoa học, trong đó bao gồm phần số mũ và phần định trị
Hexadecimal	Hệ thống số có cơ số là 16
Octal	Hệ số có cơ số nền là 8
BCD	Binary Coded Decimal: là các mã số, trong đó mỗi chữ số thập phân, từ 0 đến 9, được đại diện bởi một nhóm bốn bit
Alphanumeric (chữ-số)	Bao gồm các chữ số, chữ cái, và các ký hiệu khác
ASCII	Mã tiêu chuẩn của Mỹ dùng trong việc trao đổi thông tin, mã chữ và số được sử dụng rộng rãi nhất.



Tóm tắt nội dung chương học

- Qua Phần 2 - Chương 2, sinh viên cần nắm những nội dung chính sau:
 - ☐ Máy tính thực hiện các phép tính như thế nào
 - ☐ Máy tính biểu diễn số có dấu như thế nào?
 - ☐ Dấu chấm động là gì ? Máy tính biểu diễn số dấu chấm động như thế nào?
 - ☐ Số BCD là gì và Tại sao ta phải sử dụng số BCD ?
 - ☐ Mã ASCII là gì và Mã ASCII được sử dụng để làm gì trong máy tính?