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

ĐIỆN TỬ CHO CÔNG NGHỆ THÔNG TIN

Electronics for Information Technology

IT3420

Đỗ Công Thuần

Bộ môn Kỹ thuật Máy tính

Email: thuandc@soict.hust.edu.vn

Thông tin chung

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

Nội dung

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

Chương 4:
Cơ sở lý thuyết mạch số

1. Giới thiệu về Hệ thống số
2. Hệ đếm
3. Các phép toán số học

Bài giảng có sử dụng hình vẽ, text từ các tài liệu tham khảo:
Digital electronics: Principles, Devices, and Applications, Anil Kumar Maini 2007 John Wiley & Sons
Fundamentals of Logic Design, Seventh Edition, Charles H. Roth, Jr. and Larry L. Kinney
Digital Fundamentals, Thomas L. Floyd, Eleventh Edition, Pearson Education Limited 2015

Chương 4:
Cơ sở lý thuyết mạch số

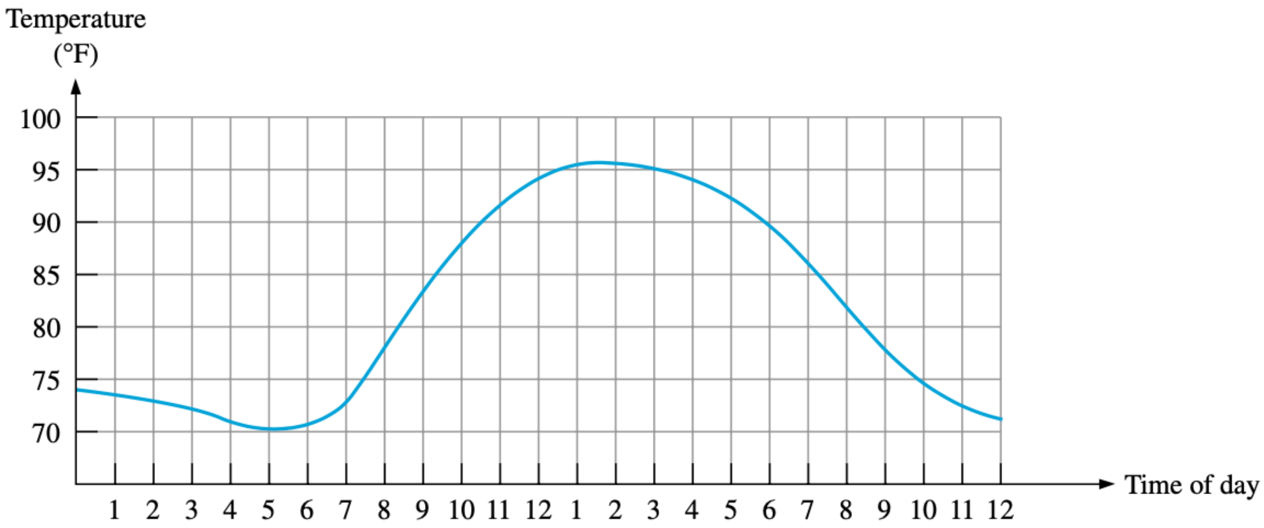
1. Giới thiệu về hệ thống số
2. Hệ đếm
3. Các phép toán số học

Giới thiệu về hệ thống số

- Các hệ thống điện tử được chia thành 2 loại: số và tương tự.
- Điện tử tương tự xử lý các tín hiệu/đại lượng tương tự
- Điện tử số xử lý các tín hiệu/đại lượng số

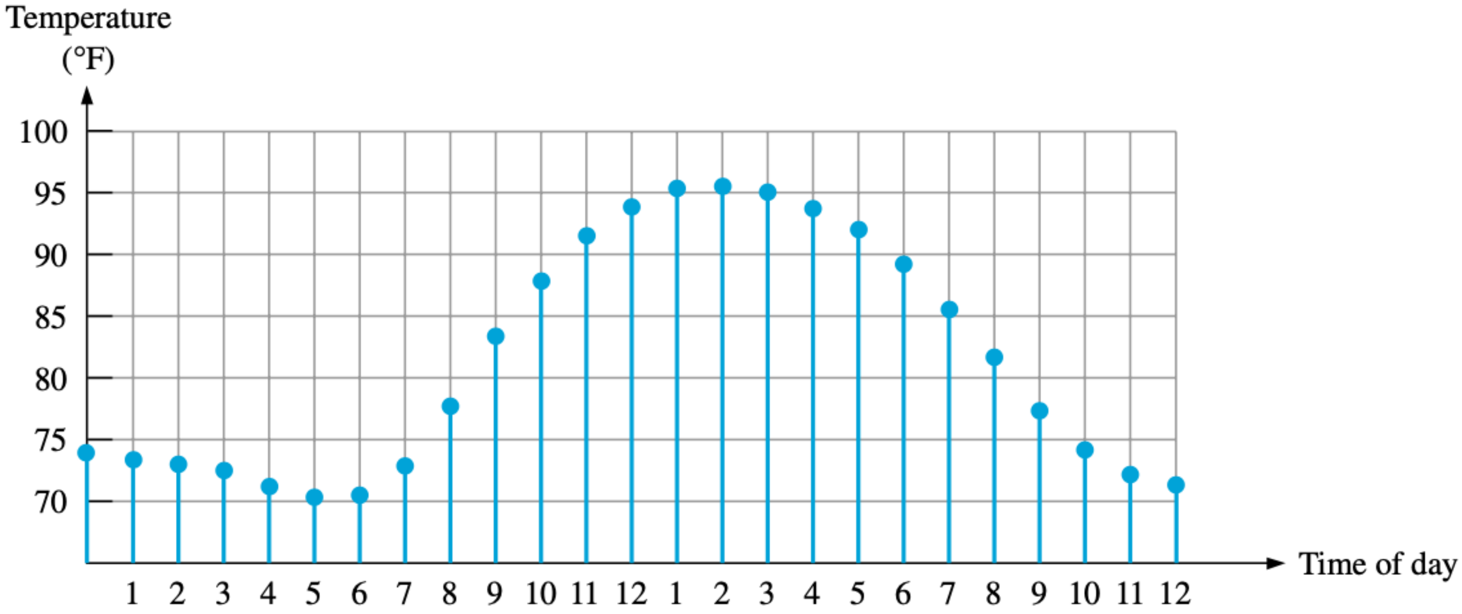
Tín hiệu và chuyển đổi tín hiệu

- Tín hiệu tương tự là tín hiệu có một tập các giá trị liên tục trong một khoảng thời gian xác định.
- Có thể là một dạng sóng lặp đi lặp lại như sóng hình sin hoặc là một tín hiệu biến đổi liên tục chứa thông tin như sóng âm thanh…



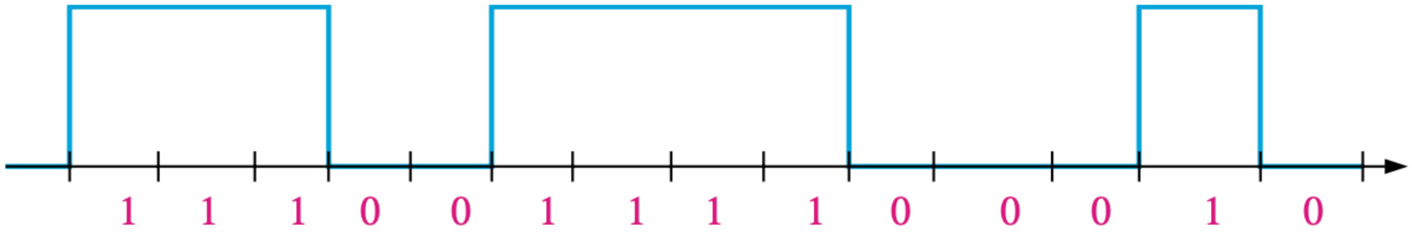
Tín hiệu và chuyển đổi tín hiệu

- Tín hiệu số là tín hiệu có một tập các giá trị rời rạc trong một khoảng thời gian xác định.



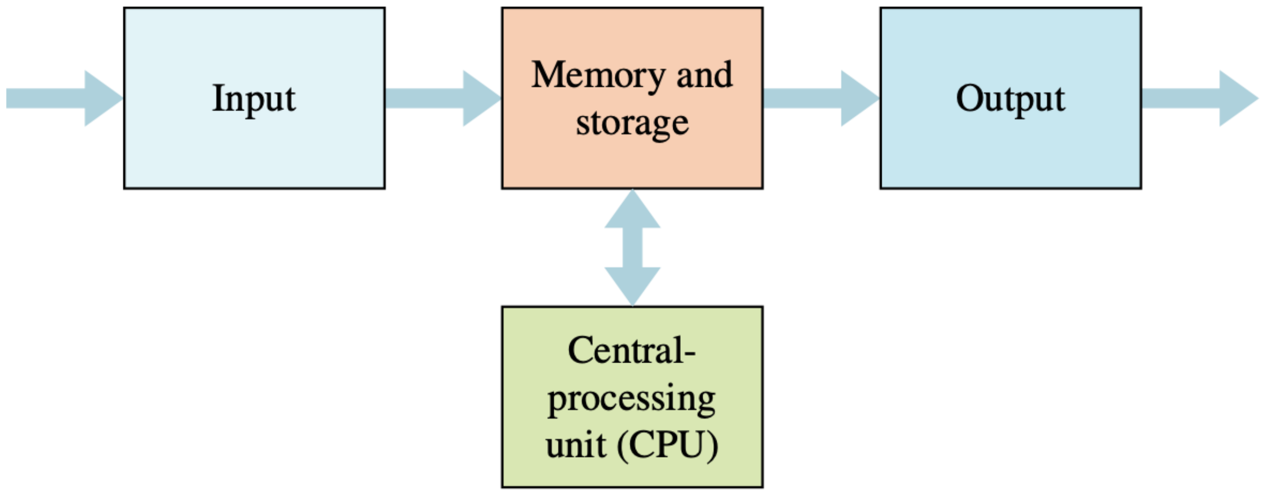
Tín hiệu và chuyển đổi tín hiệu

- Tín hiệu số được mã hoá thành một chuỗi liên tục các bit 0 và 1.
- Chuỗi bit được truyền đi dưới dạng các xung hay các mức điện áp, trong đó mức điện áp cao biểu diễn bit 1 và mức điện áp thấp biểu diễn bit 0.



Hệ thống số

- Một hệ thống số bao gồm các chức năng logic riêng lẻ được kết nối với nhau để thực hiện một thao tác cụ thể hoặc tạo ra một đầu ra xác định trước.
- Ví dụ máy tính được tạo thành từ các khối như hình mô tả dưới đây:



Ưu điểm của hệ thống số

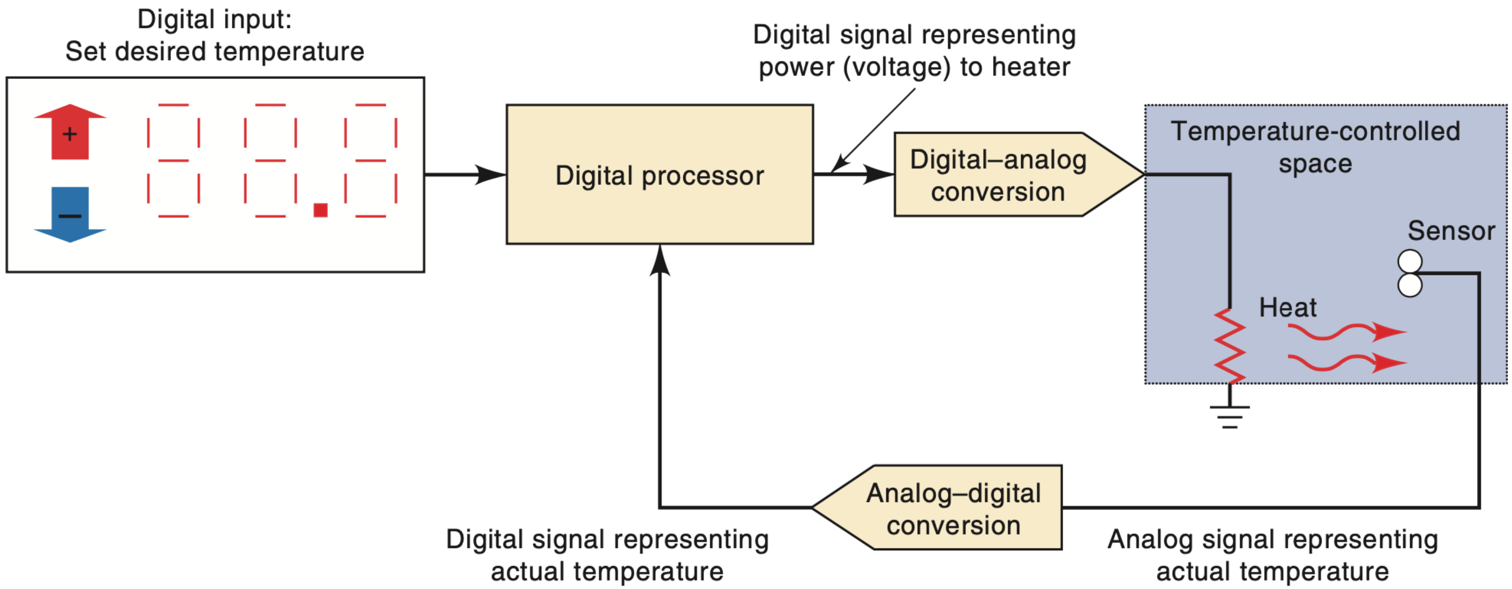
- Các hệ thống số nói chung được thiết kế dễ dàng
- Thông tin được lưu trữ dễ dàng
- Độ chính xác được duy trì một cách dễ dàng trên toàn bộ hệ thống
- Có thể lập trình được
- Ít bị ảnh hưởng bởi nhiễu
- Mạch số có thể đưa lên chip bán dẫn

Hạn chế của hệ thống số

- Thế giới thực là tương tự và số hoá tín hiệu tương tự luôn dẫn đến sai số
- Việc xử lý tín hiệu số hoá cần thời gian
- Để có thể tận dụng được ưu điểm của kỹ thuật số khi xử lý tín hiệu vào và ra tương tự cần thực hiện 4 bước sau:
  - Chuyển đổi biến vật lý thành tín hiệu điện (tương tự)
  - Chuyển đổi tín hiệu điện (tương tự) về dạng số
  - Xử lý thông tin số
  - Chuyển đổi tín hiệu số về dạng tín hiệu tương tự trong thế giới thực.

Kết hợp hệ thống số và ttương tự

- Phần lớn các ứng dụng thực tế là sự kết hợp hệ thống số và tương tự



Chương 4:
Cơ sở lý thuyết mạch số

1. Giới thiệu về Hệ thống số
2. Hệ đếm
3. Các phép toán số học

Hệ đếm

- Tập hợp các **ký hiệu** và **qui tắc** sử dụng tập ký hiệu để **biểu diễn** và xác định các giá trị.
  - Hệ La mã: I, V, X, L, C,..
  - Quy tắc: IX, XV, XXX
- Mỗi hệ đếm sử dụng một số ký hiệu (ký tự, chữ số,..) hữu hạn
  - Tổng số ký số của mỗi hệ đếm được gọi là **cơ số** (*base*, *radix*), ký hiệu là ***r***.
  - **Ví dụ:** Hệ đếm cơ số 10: sử dụng các chữ số từ 0 - 9.

Hệ đếm

- Trên lý thuyết, có thể biểu diễn một giá trị theo hệ đếm cơ số bất kì.
- Trong tin học, quan tâm đến các hệ đếm:
  - Hệ thập phân (*Decimal System*)

    → Con người sử dụng
  - Hệ nhị phân (*Binary System*)

    → Máy tính sử dụng
  - Hệ đếm bát phân/hệ cơ số 8 (*Octal System*)

    → Dùng để viết gọn số nhị phân.
  - Hệ mười sáu (*Hexadecimal System*)

    → Dùng để viết gọn số nhị phân

Hệ đếm

- Hệ đếm cơ số ***r***
- Các hệ cơ số thông dụng
- Biểu diễn số âm
- Chuyển đổi giữa các hệ cơ số
- Bốn định lý trong chuyển đổi giữa các hệ cơ số
- Số dấu phẩy động

Hệ đếm cơ số *r*

- Sử dụng ***r*** chữ số từ 0 → (*r*-1)để biểu diễn
- Một số N trong hệ cơ số ***r*** được biểu diễn dưới dạng: $a_5\, a_4\, a_3\, a_2\, a_1\, a_0\, .\, a_{-1}\, a_{-2}\, a_{-3}$
- Tổng quát, N được viết dưới dạng:

$$N = \mathrm{a_{n-1}} \times \mathrm{r^{n-1}} + \cdots + \mathrm{a_0} \times \mathrm{r^0} + \mathrm{a_{-1}} \times \mathrm{r^{-1}} + \cdots + \mathrm{a_{-m}} \times \mathrm{r^{-m}}$$

$$= \sum_{\mathrm{n-1}}^{\mathrm{-m}} \mathrm{a_i} \times \mathrm{r^i}$$

- Biểu diễn giá trị phần nguyên: r^0, r^1, r^2…
- Biểu diễn phân số: r^{-1}, r^{-2}, r^{-3}…

Các hệ cơ số thông dụng

- Hệ cơ số 10
- Hệ cơ số 2
- Hệ cơ số 8
- Hệ cơ số 16

Hệ cơ số 10

- Sử dụng 10 chữ số từ 0→9
- Phần nguyên: 10^0, 10^1, 10^2, 10^3…
- Phân số: 10^{-1}, 10^{-2}, 10^{-3}…
- Ký hiệu: $(3456.265)_{10}$

Hệ cơ số 2

- Sử dụng 2 chữ số: 0, 1
- Phần nguyên: 2^0, 2^1, 2^2, 2^3…
- Phân số: 2^{-1}, 2^{-2}, 2^{-3}…
- Ký hiệu: $(0011.0111)_2$
- Ưu điểm:
  - Sử dụng được các phép toán logic
  - Tất cả các loại dữ liệu được biểu diễn dưới dạng 0 và 1
  - Các mạch điện sử dụng cho các phép toán cho 0 và 1 được đơn giản hóa

Hệ cơ số 8

- Sử dụng 8 chữ số: $0 \rightarrow 7$
- Phần nguyên: 8^0, 8^1, 8^2, 8^3…
- Phân số: 8^{-1}, 8^{-2}, 8^{-3}…
- Ký hiệu: $(123)_8$

Hệ cơ số 16

- Sử dụng 16 chữ số: 0→9, A, B, C, D, E, F
- Phần nguyên: 16^0, 16^1, 16^2, 16^3…
- Phân số: 16^{-1}, 16^{-2}, 16^{-3}…
- Ký hiệu: $(2ABE)_{16}$
- Được sử dụng để biểu diễn các số lớn
  - Hệ cơ số 10: 0 – 65536
  - Hệ cơ số 2: 00000000 00000000 - 11111111 11111111
  - Hệ cơ số 16: 0000 - FFFF

Một số khái niệm chung – Hệ cơ số 2

- **Bit**: chỉ đơn vị nhỏ nhất của thông tin, gồm bit 0 và 1.
- **Byte**: 1 chuỗi 8 bit, là một đơn vị lưu trữ thông tin trong máy tính.
- **Word**: 1 chuỗi bit, tùy theo máy tính mà chiều dài 1 từ có thể là 1 byte, 2 bytes, 3 bytes, 4 bytes hay hơn.
- **Số bù 1**: đạt được khi đảo tất cả các bit
  - Ví dụ: Số bù 1 của $(10010110)_2$ là $(01101010)_2$
- **Số bù 2**: đạt được khi cộng thêm 1 vào số bù 1
  - Ví dụ: Số bù 2 của $(10010110)_2$ là $(01101011)_2$

Một số khái niệm chung – Hệ cơ số 10

- Số bù 9: đạt được bằng cách trừ mỗi số bởi 9
  - Ví dụ: Số bù 9 của $(2496)_{10}$ là $(7503)_{10}$

- Số bù 10: đạt được bằng cách cộng 1 vào số bù 9
  - Ví dụ: Số bù 10 của $(2496)_{10}$ là $(7504)_{10}$

Một số khái niệm chung – Hệ cơ số 8

- Số bù 7: đạt được bằng cách trừ mỗi số bởi 7
  - Ví dụ: Số bù 7 của $(562)_8$ là $(215)_8$

- Số bù 8: đạt được bằng cách cộng 1 vào số bù 7
  - Ví dụ: Số bù 8 của $(562)_8$ là $(216)_8$

Một số khái niệm chung – Hệ cơ số 16

- Số bù 15: đạt được bằng cách trừ mỗi số bởi 15
  - Ví dụ: Số bù 15 của $(3BF)_{16}$ là $(C40)_{16}$

- Số bù 16: đạt được bằng cách cộng 1 vào số bù 15
  - Ví dụ: Số bù 16 của $(2AE)_{16}$ là $(D52)_{16}$

Ví dụ 4.1

- Số bù 7 của một số trong hệ cơ số 8 là $(5264)_8$. Tìm số nhị phân và số hexa tương đương của số đó.
  - Số bù 7: $(5264)_8$
  - Số octal: $(2513)_8$
  - Số nhị phân $= (010\ 101\ 001\ 011)_2$
    $= (10101001011)_2$
  - Số hexa $= (10101001011)_2$
    $= (101\ 0100\ 1011)_{16}$
    $= (54B)_{16}$

Chuyển đổi giữa các hệ cơ số

1. Khi chuyển sang hệ cơ số 10, phải phân tách phần nguyên và phân số, với hệ cơ số r,
   - Biểu diễn giá trị phần nguyên dưới dạng: r^0, r^1, r^2, r^3…
   - Biểu diễn phân số dưới dạng: r^{-1}, r^{-2}, r^{-3}…

Chuyển đổi từ hệ cơ số 2

- Ví dụ: $(1101.0101)_2 = (13.3125)_{10}$
- Phần nguyên: (1101)
  - Chuyển sang hệ cơ số 10 tương đương:

  $= 1x2^0 + 0x2^1 + 1x2^2 + 1x2^3$

  $= 1 + 0 + 4 + 8 = 13$
- Phân số: $(.0101)_2$
  - Chuyển sang hệ cơ số 10 tương đương:

  $= 0x2^{-1} + 1x2^{-2} + 0x2^{-3} + 1x2^{-4}$

  $= 0 + 0.25 + 0 + 0.0625 = 0.3125$

Chuyển đổi từ hệ cơ số 8

- Ví dụ: $(135.21)_8 = (93.265625)_{10}$
- Phần nguyên: (135)
  - Chuyển sang hệ cơ số 10 tương đương:

  $= 5x8^0 + 3x8^1 + 1x8^2$

  $= 5 + 24 + 64 = 93$
- Phân số: $(.21)_2$
  - Chuyển sang hệ cơ số 10 tương đương:

  $= 2x8^{-1} + 1x8^{-2}$

  $= 0.25 + 0.015625 = 0.265625$

Chuyển đổi từ hệ cơ số 16

- Ví dụ: $(1BF.A1)_{16} = (447.62890625)_{10}$
- Phần nguyên: (1BF)
  - Chuyển sang hệ cơ số 10 tương đương:

  $= 15x16^0 + 11x16^1 + 1x16^2$

  $= 15 + 176 + 256 = 447$
- Phân số: $(.A1)_2$
  - Chuyển sang hệ cơ số 10 tương đương:

  $= 10x16^{-1} + 1x16^{-2}$

  $= 0.625 + 0.00390625 = 0.62890625$

Chuyển đổi giữa các hệ cơ số

1. Khi chuyển sang hệ cơ số 10, phải phân tách phần nguyên và phân số, với hệ cơ số r,
   - Biểu diễn giá trị phần nguyên dưới dạng: r^0, r^1, r^2, r^3…
   - Biểu diễn phân số dưới dạng: r^{-1}, r^{-2}, r^{-3}…
2. Khi chuyển từ hệ cơ số 10 sang hệ cơ số ***r***, chia phần nguyên cho ***r*** cho đến khi kết quả phép chia bằng 0, phần chuyển đổi là tập hợp phần dư viết theo chiều ngược lại.

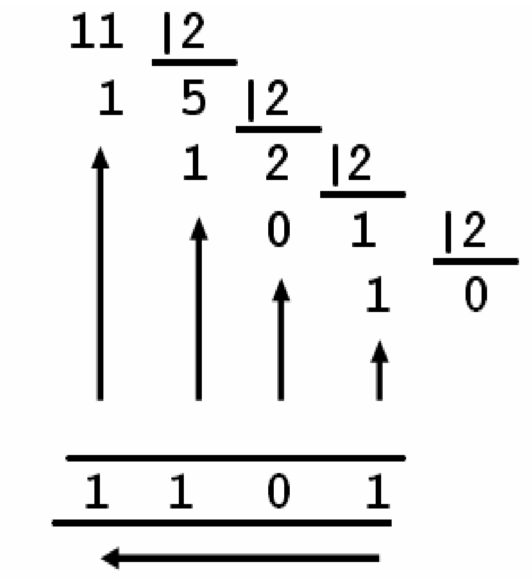
Chuyển đổi từ hệ cơ số 10 sang hệ cơ số 2

- Ví dụ: $(11.73)_{10} = (1011.1011)_2$

- Phần nguyên: $(11)_{10}$
- Phân số: $(0.73)_{10}$

	0.73
1	0.46
0	0.92
1	0.84
1	0.68

0.1011



Chuyển đổi từ hệ cơ số 10 sang hệ cơ số 8

- Ví dụ: $(23.75)_{10} = (27.6)_8$

- Phần nguyên: $(23)_{10}$
  - Chuyển sang hệ cơ số 8 tương đương: $(27)_8$
- Phân số: $(0.75)_{10}$
  - Chuyển sang hệ cơ số 8 tương đương: $(0.6)_8$

Chuyển đổi từ hệ cơ số 10 sang hệ cơ số 16

- Ví dụ: $(23.75)_{10} = (17.12)_{16}$

- Phần nguyên: $(23)_{10}$
  - Chuyển sang hệ cơ số 8 tương đương: $(17)_{16}$
- Phân số: $(0.75)_{10}$
  - Chuyển sang hệ cơ số 8 tương đương: $(0.12)_{16}$

Chuyển đổi giữa các hệ cơ số

1. Khi chuyển sang hệ cơ số 10, phải phân tách phần nguyên và phân số, với hệ cơ số r,
   - Biểu diễn giá trị phần nguyên dưới dạng: r^0, r^1, r^2, r^3…
   - Biểu diễn phân số dưới dạng: r^{-1}, r^{-2}, r^{-3}…
2. Khi chuyển từ hệ cơ số 10 sang hệ cơ số ***r***, chia phần nguyên cho ***r*** cho đến khi kết quả phép chia bằng 0, phần chuyển đổi là tập hợp phần dư viết theo chiều ngược lại.
3. Chuyển đổi giữa hệ cơ số 2 và 8 bằng cách chuyển các số tương đương trong hệ cơ số 8 sang hệ cơ số 2 và ngược lại.
4. Chuyển đổi giữa hệ cơ số 8 và 16 bằng cách chuyển sang hệ cơ số 2 hoặc 10 tương đương.

Chuyển đổi giữa hệ cơ số 2 và hệ cơ số 8

- Ví dụ: $(375.2)_8 = (?)_2$

  $(1100110001.0111)_2 = (?)_8$

- Từ hệ cơ số 8 sang hệ cơ số 2: $(375.2)_8 = (?)_2$

$$
\begin{aligned}
(375.2)_8 &= (011\ 111\ 101.010)_2 \\
&= (011111101.010)_2 \\
&= (11111101.01)_2
\end{aligned}
$$

- Từ hệ cơ số 2 sang hệ cơ số 8:

$$
\begin{aligned}
(1100110001.0111)_2 &= (1\ 100\ 110\ 001.011\ 1)_2 \\
&= (001\ 100\ 110\ 001.011\ 100)_2 \\
&= (1461.34)_8
\end{aligned}
$$

Chuyển đổi giữa hệ cơ số 8 và hệ cơ số 16

- Ví dụ: $(375.2)_8 = (?)_{16}$

  $(3BF.1A)_{16} = (?)_8$

- Từ hệ cơ số 8 sang hệ cơ số 16: $(375.2)_8 = (?)_{16}$

$$
\begin{aligned}
(375.2)_8 &= (011\ 111\ 101.010)_2 \\
&= (011111101.010)_2 \\
&= (1111\ 1101.01\textcolor{red}{00})_2 = (FD.4)_{16}
\end{aligned}
$$

- Từ hệ cơ số 16 sang hệ cơ số 8: $(3BF.1A)_{16} = (?)_8$

$$
\begin{aligned}
(3BF.1A)_{16} &= (0011\ 1011\ 1111.0001\ 1010)_2 \\
&= (1110111111.0001101)_2 \\
&= (001\ 110\ 111\ 111.000\ 110\ 10\textcolor{red}{0})_2 \\
&= (1677.064)_8
\end{aligned}
$$

Các phương pháp biểu diễn số âm

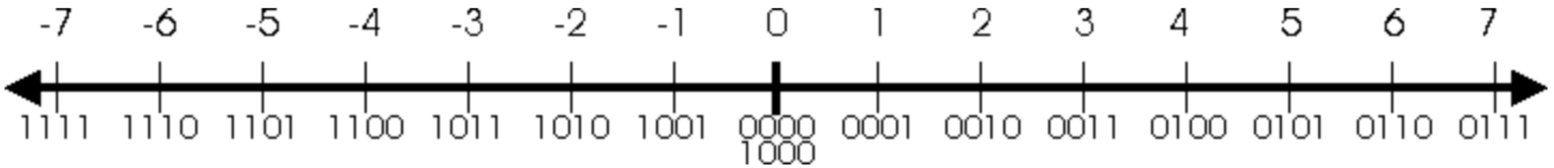
- Các phương pháp biểu diễn số âm phổ biến nhất bao gồm:
  - Phương pháp dấu lượng
  - Phương pháp bù 1
  - Phương pháp bù 2
- Trong các phương pháp trên, bit bên trái nhất của một số (bit có trọng số lớn nhất) được sử dụng để biểu diễn dấu: bit 0 cho số dương và bit 1 cho số âm.

Phương pháp dấu lượng

- Với n bit, sử dụng 1 bit ngoài cùng để biểu diễn dấu và (n-1) bit còn lại để biểu diễn độ lớn của số.
  - Bit 0: biểu diễn dấu '+'
  - Bit 1: biểu diễn dấu '-'
- Ví dụ: với 8 bit
  - +9 được biểu diễn bởi $(00001001)_2$
  - -9 được biểu diễn bởi $(10001001)_2$
- Dải biểu diễn: với n bit, dải biểu diễn khả dụng từ $-(2^{n-1}-1)$ đến $+(2^{n-1}-1)$ bao gồm số (+0) và (−0)
  - Ví dụ: với 8 bit, dải biểu diễn từ -127_{10} đến $+127_{10}$ bao gồm số 00000000 (+0) và 10000000 (−0)

Phương pháp dấu lượng

- Ưu điểm: dễ dàng nhận biết số âm/dương bằng cách kiểm tra bit có trọng số lớn nhất
- Nhược điểm:
  - Một chuỗi bit bị lãng phí (-0: 10000000) và (+0: 00000000)



  - Phép cộng nhị phân không hiệu quả

Phương pháp sử dụng số bù 1

- Số âm được biểu diễn bằng cách lấy phần bù 1 của số dương
- Ví dụ: với 8 bit
  - +9 được biểu diễn bởi $(00001001)_2$
  - -9 được biểu diễn bởi $(11110110)_2$
- Với n bit, biểu diễn từ $-(2^{n-1}-1)$ đến $+(2^{n-1}-1)$
  - Ví dụ: 8 bit biểu diễn từ -127 đến +127

Phương pháp sử dụng số bù 1

- Ưu điểm: dễ dàng nhận biết số âm/dương bằng cách kiểm tra bit có trọng số lớn nhất
- Nhược điểm:
  - Một chuỗi bit bị lãng phí (-0: 11111111) và (+0: 00000000)
  - Phép cộng nhị phân không hiệu quả

```
–3          1100
–4          1011
           ------
      (1)   0111
               1     (end-around carry)
           ------
            1000     (correct answer, no overflow)
```

Phương pháp sử dụng số bù 1

- Ưu điểm: dễ dàng nhận biết số âm/dương bằng cách kiểm tra bit có trọng số lớn nhất
- Nhược điểm:
  - Một chuỗi bit bị lãng phí (-0: 11111111) và (+0: 00000000)
  - Phép cộng nhị phân không hiệu quả

–5		1010	
–6		1001	
	(1)	0111	
	→	1	(end-around carry)
		0100	(wrong answer because of overflow)

Phương pháp sử dụng số bù 2

- Ở dạng bù 2:
  - Bit 0: biểu diễn dấu '+'
  - Bit 1: biểu diễn dấu '-'
  - Các bit còn lại được dùng để biểu diễn độ lớn
  - Số âm được biểu diễn bằng số bù 1 của số dương và cộng thêm 1
- Ví dụ: với 8 bit
  - +9 được biểu diễn bởi: $(00001001)_2$
  - -9 được biểu diễn bởi: $(11110111)_2$
- Với n bit, biểu diễn từ $-(2^{n-1})$ đến $+(2^{n-1}-1)$
  - Ví dụ: 8 bit biểu diễn từ -128 đến +127

Phương pháp sử dụng số bù 2

- Ưu điểm:
  - Có một cách biểu diễn duy nhất cho số 0
- Nhược điểm:
  - Một chuỗi bit bị lãng phí (-0: 11111111) và (+0: 00000000)
  - Phép cộng nhị phân không hiệu quả
- Để thực hiện phép trừ:
  - Chuyển thành phép cộng số bù 2 của số trừ
  - Loại bỏ bit nhớ nếu có

$$
\begin{array}{rl}
11111000 & (-8) \\
\underline{00010011} & \underline{+19} \\
\cancel{(1)}00001011 = & +11
\end{array}
$$

(discard last carry)

Dấu chấm động

- Được sử dụng để biểu diễn các số rất lớn và rất nhỏ
- Biểu thị dưới dạng: $N = M \times b^E$

  M: giá trị phần định trị

  b: cơ số

  E: số mũ
- Hệ cơ số 10: $N = M \times 10^E$
- Hệ cơ số 16: $N = M \times 16^E$
- Hệ cơ số 2: $N = M \times 2^E$

Chương 4:
Cơ sở lý thuyết mạch số

1. Giới thiệu về Hệ thống số
2. Hệ đếm
3. Các phép toán số học

Các phép toán số học

- Quy luật cơ bản của phép cộng và trừ
- Phép cộng số nhị phân lớn
- Phép trừ số nhị phân lớn
- Phép nhân số nhị phân
- Phép chia số nhị phân
- Dấu chấm động

Quy luật cơ bản của phép cộng

1. 0 + 0 = 0
2. 0 + 1 = 1
3. 1 + 0 = 1
4. 1 + 1 = 0 (nhớ 1)
5. 1 + 1 + 1 = 1 (nhớ 1)

Phép cộng sử dụng phương thức bù 2

- Cả hai số đều dương
- Số lớn hơn là số dương
- Số lớn hơn là số âm
- Cả hai số đều âm

Phép cộng sử dụng phương thức bù 2

- Trường hợp 1: +37 và +18
  - Số bù 2 của + 37 = 00100101
  - Số bù 2 của + 18 = 00010010
  - Kết quả = $(00110111)_2 = (+55)_{10}$

- Trường hợp 2: +37 và -18
  - Số bù 2 của + 37 = 00100101
  - Số bù 2 của - 18 = 11101110
  - Kết quả = $(00010011)_2 = (+19)_{10}$

Phép cộng sử dụng phương thức bù 2

- Trường hợp 3: +18 và -37
  - Số bù 2 của - 37 = 11011011
  - Số bù 2 của + 18 = 00010010
  - Kết quả = $(11101101)_2$ = $(-19)_{10}$

- Trường hợp 4: -18 và -37
  - Số bù 2 của -18 = 11101110
  - Số bù 2 của -37 = 11011011
  - Kết quả = $(11001001)_2$ = $(-55)_{10}$

Phép cộng sử dụng phương thức bù 2

- Có xảy ra tràn số, kết quả sai:

$$
\begin{array}{rl}
 & X = 0100\ 1011 = +75 \\
+ & \underline{Y = 0101\ 0001 = +81} \\
 & S = 1001\ 1100 \neq +156 \\
 & (S = -2^7 + 2^4 + 2^3 + 2^2 = -100)
\end{array}
$$

$$
\begin{array}{rl}
 & X = 1001\ 1000 = -104 \\
+ & \underline{Y = 1011\ 0110 = -74} \\
 & S = 0100\ 1110 \neq -178 \\
 & C_{out} = 1 \rightarrow \text{bỏ qua} \\
 & (S = 2^6 + 2^3 + 2^2 + 2^1 = 78)
\end{array}
$$

Phép cộng số nhị phân lớn

$$A + B + C_{in}$$

A	B	Carry-in (C_{in})	Sum	Carry-out (C_o)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Quy luật cơ bản của phép trừ

1. 0 - 0 = 0
2. 1 - 0 = 1
3. 1 - 1 = 0
4. 0 - 1 = 1

   (mượn 1)

Phép trừ sử dụng phương thức bù 2

- Cả hai số đều dương, số bị trừ nhỏ hơn
- Số lớn hơn là số dương, số bị trừ lớn hơn
- Số trừ dương, số bị trừ âm và giá trị tuyệt đối lớn hơn
- Số trừ dương, số bị trừ âm và giá trị tuyệt đối nhỏ hơn
- Cả hai số đều âm, số trừ nhỏ hơn
- Cả hai số đều âm, số trừ lớn hơn

Phép trừ sử dụng phương thức bù 2

- Trường hợp 1: +24 – (+14)
  - Số bù 2 của +24 = 00011000
  - Số bù 2 của +14 = 00001110
  - Kết quả = $(00001010)_2 = (+10)_{10}$

- Trường hợp 2: +14 – (+24)
  - Số bù 2 của +24 = 00011000
  - Số bù 2 của +14 = 00001110
  - Số bù 2 của số bị trừ = 11101000
  - Kết quả = $(11110110)_2 = (-10)_{10}$

Phép trừ sử dụng phương thức bù 2

- Trường hợp 3: +24 – (– 14)
  - Số bù 2 của +24 = 00011000
  - Số bù 2 của -14 = 11110010
  - Số bù 2 của số bị trừ = 00001110
  - Kết quả = $(00100110)_2$ = $(+38)_{10}$

- Trường hợp 4: +14 – (– 24)
  - Số bù 2 của -24 = 11101000
  - Số bù 2 của +14 = 00001110
  - Số bù 2 của số bị trừ = 00011000
  - Kết quả = $(00100110)_2$ = $(+38)_{10}$

Phép trừ sử dụng phương thức bù 2

- Trường hợp 5: -24 – (– 14)
  - Số bù 2 của -24 = 11101000
  - Số bù 2 của -14 = 11110010
  - Số bù 2 của số bị trừ = 00001110
  - Kết quả = $(11110110)_2 = (-10)_{10}$

- Trường hợp 6: -14 – (– 24)
  - Số bù 2 của -14 = 11110010
  - Số bù 2 của -24 = 11101000
  - Số bù 2 của số bị trừ = 00011000
  - Kết quả = $(00001010)_2 = (+10)_{10}$

Phép trừ sử dụng phương thức bù 2

1. Biểu thị số trừ và số bị trừ dưới dạng số bù 2
2. Tìm số bù 2 của số bị trừ
3. Cộng số bù 2 của số bị trừ với số trừ
4. Bỏ qua bit nhớ cuối cùng nếu có
5. Kết quả ở dạng số bù 2
6. Số bù 2 được sử dụng khi kết quả mong muốn nằm trong khoảng -2^{n-1} đến $+(2^{n-1}-1)$, với n là số bit được sử dụng để biểu diễn số.

Phép trừ số nhị phân lớn

$$A - B - B_{in}$$

Inputs			Outputs	
Minuend (A)	Subtrahend (B)	Borrow-in (B_{in})	Difference (D)	Borrow-out (B_o)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Phép nhân

- Quy luật cơ bản của phép nhân

1. 0 x 0 = 0
2. 0 x 1 = 0
3. 1 x 0 = 0
4. 1 x 1 = 1

- Thực hiện phép nhân bằng máy tính

1. Thuật toán cộng và dịch bit trái
2. Thuật toán cộng và dịch bit phải

Thuật toán cộng và dịch bit trái

$$
\begin{array}{r}
1\;0\;1\;1\;1 \text{.............} (23)_{10} \\
\times\;1\;1\;0 \text{.............} (6)_{10} \\
\hline
0\;0\;0\;0\;0 \\
1\;0\;1\;1\;1 \\
1\;0\;1\;1\;1 \\
\hline
1\;0\;0\;0\;1\;0\;1\;0
\end{array}
$$

Thuật toán cộng và dịch bit phải

1 0 1 1 1	Multiplicand
1 1 0	Multiplier
0 0 0 0 0	Start
+ 0 0 0 0 0	
0 0 0 0 0	Result of first addition
0 0 0 0 0	0 (Result of addition shifted one bit to right)
+ 1 0 1 1 1	
1 0 1 1 1	Result of second addition
0 1 0 1 1	10 (Result of addition shifted one bit to right)
+ 1 0 1 1 1	
1 0 0 0 1 0	Result of third addition
0 1 0 0 0 1	010 (Result of addition shifted one bit to right)

Phép chia

- Thực hiện phép chia bằng máy tính

1. Thuật toán trừ và dịch bit phải
2. Thuật toán trừ và dịch bit trái

Thuật toán trừ và dịch bit phải

- Số bị chia: 100110
- Số chia: 1100

	Quotient		
First step	0	1 0 0 1 1 0	Dividend
		–1 1 0 0	Divisor
Second step	1	1 0 0 1 1	First five MSBs of dividend
		–1 1 0 0	Divisor shifted to right
		0 1 1 1	First subtraction remainder
Third step	1	0 1 1 1 0	Next MSB appended
		–1 1 0 0	Divisor right shifted
		0 0 1 0	Second subtraction remainder

Thuật toán trừ và dịch bit trái

Quotient	1 0 0 1 −1 1 0 0	1 0
0	1 1 0 1 +1 1 0 0	Borrow exists
	1 0 0 1	Final carry ignored
	1 0 0 1 1 −1 1 0 0	Next MSB appended
1	0 1 1 1	No borrow
	0 1 1 1 0 −1 1 0 0	Next MSB appended
1	0 0 0 1 0	No borrow