Hệ nhị phân

Bách khoa toàn thư mở Wikipedia

Hệ nhị phân (hay **hệ đếm cơ số hai**) là một hệ đếm dùng hai ký tự để biểu đạt một giá trị số, bằng tổng số các lũy thừa của 2. Hai ký tự đó thường là 0 và 1; chúng thường được dùng để biểu đạt hai giá trị hiệu điện thế tương ứng (có hiệu điện thế, hoặc hiệu điện thế cao là 1 và không có, hoặc thấp là 0). Do có ưu điểm tính toán đơn giản, dễ dàng thực hiện về mặt vật lý, chẳng hạn như trên các mạch điện tử, hệ nhị phân trở thành một phần kiến tạo căn bản trong các máy tính đương thời.

Mục lục

Lịch sử

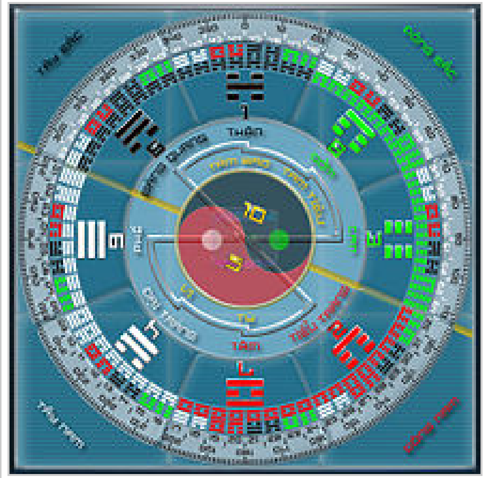
Hệ nhị phân được nhà toán học cổ người Ấn Độ Pingala phác thảo từ thế kỷ thứ ba trước Công Nguyên.

Một bộ trọn 8 hình bát quái với 64 hình sao sáu cạnh, tương đồng với 3 bit và 6 bit trong hệ số nhị phân, đã được ghi lại trong điển tịch cổ Kinh Dịch.

Nhiều tổ hợp nhị phân tương tự cũng được tìm thấy trong hệ thống bói toán truyền thống của châu Phi, ví dụ như Ifá, và trong môn bói đất của phương Tây.

Tổ hợp thứ tự của những hình sao sáu cạnh trong Kinh Dịch, đại diện cho một dãy số nguyên thập phân từ 0 đến 63, cùng với một công thức để sinh tạo dãy số ấy, đã được học giả và nhà triết học người Trung Hoa tên là Thiệu Ung (邵雍), thế kỷ 11, thiết lập. Dầu vậy, không có ghi chép nào để lại, thể hiện bằng chứng là Thiệu Ung thông hiểu cách tính toán, dùng hệ nhị phân.

Trong thế kỷ 17, nhà triết học người Đức tên là Gottfried Leibniz đã ghi chép lại một cách trọn vẹn hệ thống nhị phân trong bài viết "Giải thích về toán thuật trong hệ nhị phân" (*Explication de l'Arithmétique Binaire*). Hệ thống số mà Leibniz dùng chỉ bao gồm số 0 và số 1, tương đồng với hệ số nhị phân đương đại.[1]



Hệ 64 quẻ Tiên thiên và Hà đồ trong Kinh dịch

Năm 1854, nhà toán học người Anh, George Boole đã cho xuất bản một bài viết chi tiết về một hệ thống lôgic mà sau này được biết là đại số Boole, đánh dấu một bước ngoặt trong lịch sử toán học. Hệ thống lôgic của ông đã trở thành nền tảng trong việc kiến tạo hệ nhị phân, đặc biệt trong việc thực thi hệ thống này trên bảng điện tử.[2]

Vào năm 1937, nhà toán học và kỹ sư điện tử người Mỹ, Claude Elwood Shannon, viết một luận án cử nhân tại MIT, trình bày phương thức kiến tạo hệ thống đại số Boole và số học nhị phân dùng các rơ-le và công tắc lần đầu tiên trong lịch sử. Bài viết với đầu đề "Bản phân tích tượng hình của mạch điện dùng rơ-le và công tắc" (*A Symbolic Analysis of Relay and Switching Circuits*). Bản luận án của ông đã được chứng minh là có tính khả thi trong việc thiết kế mạch điện kỹ thuật số.[3]

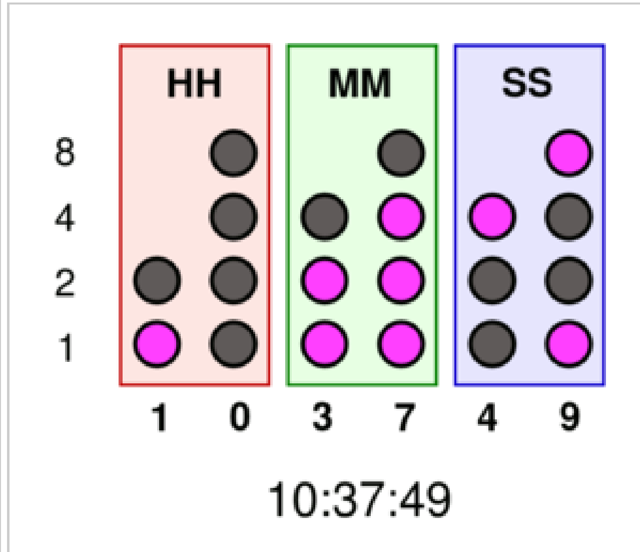
Tháng 11 năm 1937, ông George Stibitz, lúc đó đang làm việc tại Bell Labs, hoàn thành việc thiết kế một máy tính dùng các rơ-le và đặt tên cho nó là "Mô hình K" (*Model K*) - chữ K ở đây là chữ cái đầu tiên của từ *kitchen* trong tiếng Anh, nghĩa là "nhà bếp", nơi ông lắp ráp máy tính của mình. Máy tính của ông có thể tính toán dùng phép tính cộng của hệ nhị phân.[4] Cơ quan Bell Labs vì thế đã ra lệnh và cho phép một chương trình nghiên cứu tổng thể được thi hành vào cuối năm 1938 dưới sự chỉ đạo của ông Stibitz. Máy tính số phức hợp (*Complex Number Computer*) của họ, được hoàn thành vào ngày 8 tháng 1 năm 1940, có thể giải trình số phức hợp. Trong một cuộc luận chứng tại hội nghị của Hội Toán học Mỹ (*American Mathematical Society*), được tổ chức tại Dartmouth College vào ngày 11 tháng 9 năm 1940, ông Stibitz đã có thể truyền lệnh cho Máy tính số phức hợp từ xa, thông qua đường dây điện thoại, bằng một máy điện báo đánh chữ (*teletype*). Đây là máy tính đầu tiên được sử dụng với phương pháp điều khiển từ xa dùng đường dây điện thoại. Một số thành viên tham gia hội nghị và được chứng kiến cuộc thuyết trình bao gồm John von Neumann, John Mauchly và Norbert Wiener, đã viết lại sự kiện này trong hồi ký của mình.[5][6][7]

Biểu thức

Bất cứ số nào cũng có thể biểu đạt được trong hệ nhị phân bằng một dãy đơn vị bit (*binary digit*, số ký nhị phân), do đó có thể được diễn giải bằng bất cứ một cơ cấu có khả năng thể hiện hai thể trạng biệt lập. Bản liệt kê những dãy ký tự sau đây cho phép sự giải nghĩa tương đồng với những giá trị số trong hệ nhị phân:

```
1 0 1 0 0 1 1 0 1 0
| - | - - | | - | -
x o x o o x x o x o
y n y n n y y n y n
```

Giá trị số biểu đạt trong mỗi trường hợp trên phụ thuộc vào giá trị mà nó được gán ghép để đại diện. Trong máy tính, những giá trị số được biểu hiện bằng hai hiệu điện thế khác nhau; trong đĩa từ tính (*magnetic disk*) thì chiều phân của các lưỡng cực từ có thể được dùng để biểu hiện hai giá trị này. Một giá trị biểu đạt trạng thái "dương", "có" hoặc "chạy" không có nghĩa là giá trị tương tự với số 1 trong hệ số, song nó còn tuỳ thuộc vào kiến trúc của hệ thống đang được dùng.



Một cái đồng hồ nhị phân có thể dùng LED để biểu tả các số nhị phân. Trong đồng hồ này, mỗi cột các đèn LED biểu tả một con số thập phân mã hóa bằng hệ nhị phân (*binary-coded decimal*), đại diện cho thời gian, dùng hệ lục thập phân (*sexagesimal*) (60) truyền thống

Song hành với chữ số Ả Rập thường dùng, số nhị phân thường được biểu đạt bằng hai ký tự 0 và 1. Khi được viết, các số nhị phân thường được ký hiệu hóa gốc của hệ số. Những phương thức ký hiệu thường được dùng có thể liệt kê ở dưới đây:

> 100101 binary (đặc tả phân dạng hệ số)
> 100101b (chữ b nối tiếp ám chỉ phân dạng hệ số nhị phân - lấy chữ đầu của *binary* trong tiếng Anh, tức là "nhị phân")
> bin 100101 (dùng ký hiệu dẫn đầu để đặc tả phân dạng hệ số nhị phân - *bin* cũng được lấy từ *binary*)
> 100101_2 (ký hiệu viết nhỏ phía dưới ám chỉ gốc nhị phân)

Khi nói, mỗi ký tự số của các giá trị số nhị phân thường được phát âm riêng biệt, để phân biệt chúng với số thập phân. Chẳng hạn, giá trị "100" nhị phân được phát âm là "một không không", thay vì "một trăm", để biểu đạt cụ thể tính nhị phân của giá trị đang nói đến, đồng thời đảm bảo sự chính xác trong việc truyền tin qua lại. Vì giá trị "100" tương đương với giá trị "4" trong hệ bát phân, nên nếu được truyền đạt là "một trăm" thì nó sẽ gây sự hỗn loạn trong tư duy.

Biểu đạt giá trị dùng hệ nhị phân

Cách đếm trong hệ nhị phân tương tự như cách đếm trong các hệ thống số khác. Bắt đầu bằng số ở hàng đơn vị với một ký tự, việc đếm số được khai triển dùng các ký tự cho phép để ám chỉ giá trị, theo chiều tăng lên. Hệ thập phân được đếm từ ký tự 0 đến ký tự 9, trong khi hệ nhị phân chỉ được dùng ký tự 0 và 1 mà thôi.

Khi những ký tự cho một hàng đã dùng hết (như hàng đơn vị, hàng chục, hàng trăm trong hệ thập phân), thì con số tại hàng tiếp theo (về bên trái) được nâng giá trị lên một vị trí, và con số ở hàng hiện tại được hoàn trả lại vị trí đầu tiên dùng ký tự 0. Trong hệ thập phân, chu trình đếm tương tự như sau:

> 000, 001, 002, ... 007, 008, 009, (số cuối cùng ở bên phải quay trở lại vị trí ban đầu trong khi số tiếp theo ở bên trái được nâng cấp lên một giá trị)
> 0**1**0, 011, 012, ...
> ...
> 090, 091, 092, ... 097, 098, 099, (hai số bên phải chuyển về vị trí ban đầu trong khi số tiếp theo ở bên trái được nâng cấp lên một giá trị)
> **1**00, 101, 102, ...

Sau khi một con số đạt đến ký tự 9, thì con số ấy được hoàn trả lại vị trí ban đầu là số 0, đồng thời gây cho con số tiếp theo ở bên trái được nâng cấp lên một vị trí mới. Trong hệ nhị phân, quy luật đếm số tương đồng như trên cũng được áp dụng, chỉ khác một điều là số ký tự được dùng chỉ có 2 mà thôi, tức là ký tự 0 và 1 được dùng mà thôi. Vì vậy, khi một con số đã chuyển lên đến ký tự 1 trong hệ nhị phân, sự nâng cấp của giá trị bắt nó hoàn trả lại vị trí ban đầu, tức là số 0, và nâng cấp con số tiếp theo về bên trái lên một giá trị:

> 000, 001, (số cuối bên phải được hoàn trả lại vị trí ban đầu, trong khi số ở hàng bên cạnh về phía tay trái được nâng cấp lên một giá trị)
> 0**1**0, 011, (hai số cuối bên phải được hoàn trả lại vị trí ban đầu, trong khi số ở hàng bên cạnh về phía tay trái được nâng cấp lên một giá trị)
> **1**00, 101, ...

Nhị phân đơn giản hóa

Để đơn giản hoá hệ nhị phân, chúng ta có thể nghĩ như sau: Chúng ta dùng hệ thập phân. Điều này có nghĩa là các giá trị của mỗi hàng số (hàng đơn vị, hàng chục v.v..) chỉ được biểu đạt bởi một trong 10 ký tự mà thôi: 0, 1, 2, 3, 4, 5, 6, 7, 8, hoặc 9. Chúng ta ai cũng thông thuộc với những ký tự này và cách dùng của chúng trong hệ thập phân. Khi chúng ta đếm các giá trị, chúng ta bắt đầu bằng ký tự 0, luân chuyển nó đến ký tự 9. Chúng ta gọi nó là "một hàng".

Với những con số ở trên trong một hàng, chúng ta có thể liên tưởng đến vấn đáp về tính nhân. Số 5 có thể hiểu là 5×10^0 ($10^0=1$) tương đương với 5 x 1, vì bất cứ một số nào có mũ 0 cũng đều bằng 1 (tất nhiên là loại trừ số 0 ra). Khi khai triển sang bên trái một vị trí, chúng ta nâng số mũ của 10 lên một giá trị, vì vậy để biểu đạt 50, chúng ta dùng phương pháp tương tự và số này có thể được viết như 5×10^1, hoặc đơn giản hơn 5 x 10.

$$500 = (5 \times 10^2) + (0 \times 10^1) + (0 \times 10^0)$$
$$5834 = (5 \times 10^3) + (8 \times 10^2) + (3 \times 10^1) + (4 \times 10^0)$$

Khi chúng ta đã dùng hết các ký tự trong hệ thập phân, chúng ta chuyển vị trí sang bên trái và bắt đầu với số 1, đại diện cho hàng chục. Tiếp đó chúng ta hoàn trả hàng "đơn vị" về ký tự đầu tiên, số không.

Hệ nhị phân có gốc 2, cũng hoạt động trên cùng một nguyên lý như hệ thập phân, song chỉ dùng 2 ký tự để đại diện cho hai giá trị: 0 và 1. Chúng ta bắt đầu bằng hàng "đơn vị", đặt số 0 trước tiên, rồi nâng cấp lên số 1. Khi đã lên đến số 1, chúng ta không còn ký tự nào nữa để tiếp tục biểu đạt những giá trị cao hơn, do vậy chúng ta phải đặt số 1 ở "hàng hai" (tương tự như hàng chục trong hệ thập phân), vì chúng ta không có ký tự "2" trong hệ nhị phân để biểu đạt giá trị này như chúng ta có thể làm được trong hệ thập phân.

Trong hệ nhị phân, giá trị 10 có thể biểu đạt bằng hình thức tương tự như $(1 \times 2^1) + (0 \times 2^0)$. Giá trị này bằng 2 trong hệ thập phân. Nhị phân sang thập phân tương đồng:

$$1_2 = 1 \times 2^0 = 1 \times 1 = 1_{10}$$
$$10_2 = (1 \times 2^1) + (0 \times 2^0) = 2 + 0 = 2_{10}$$
$$101_2 = (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = 4 + 0 + 1 = 5_{10}$$

Để quan sát công thức biến chuyển cụ thể từ hệ này sang hệ kia, xin xem thêm phần Phương pháp chuyển hệ dưới đây.

Ngược lại, chúng ta có thể suy nghĩ theo một cách khác. Khi chúng ta đã dùng hết các ký tự trong hệ thống số, chẳng hạn dãy số "11111", chúng ta cộng thêm "1" vào phía bên trái và hoàn trả tất cả các con số ở vị trí bên phải về số "0", tạo thành 100000. Phương thức này cũng có thể dùng được cho các ký tự ở giữa dãy số. Chẳng hạn với dãy số 100111. Nếu chúng ta cộng thêm 1 vào số này, chúng ta phải chuyển vị trí về bên trái một vị trí bên cạnh các con số 1 trùng lặp (vị trí thứ tư), nâng cấp vị trí này từ số 0 lên số 1, rồi hoàn trả tất cả các con số 1 bên tay phải về vị trí số không, tạo thành 10**1000**.

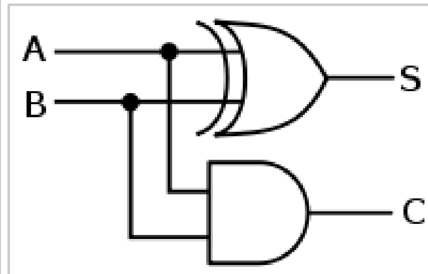
Các phép tính dùng hệ nhị phân

Phép tính dùng trong hệ nhị phân cũng tương tự như các phép tính được áp dụng trong các hệ khác. Tính cộng, tính trừ, tính nhân và tính chia cũng có thể được áp dụng với các giá trị số nhị phân.

Tính cộng

Phép tính đơn giản nhất trong hệ nhị phân là tính cộng. Cộng hai đơn vị trong hệ nhị phân được làm như sau:

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 0 (nhớ 1 lên hàng thứ 2)



Một sơ đồ mạch điện (*circuit diagram*) mạch bán cộng nhị phân, dùng để cộng hai bit với nhau, tạo nên một tổng và số nhớ mang sang hàng bên cạnh

Cộng hai số "1" với nhau tạo nên giá trị "10", tương đương với giá trị 2 trong hệ thập phân. Điều này xảy ra tương tự trong hệ thập phân khi hai số đơn vị được cộng vào với nhau. Nếu kết quả bằng hoặc cao hơn giá trị gốc (10), giá trị của con số ở hàng tiếp theo được nâng lên:

5 + 5 = 10
7 + 9 = 16

Hiện tượng này được gọi là "nhớ" hoặc "mang sang", trong hầu hết các hệ thống số dùng để tính, đếm. Khi tổng số vượt lên trên gốc của hệ số, phương thức làm là "nhớ" một sang vị trí bên trái, thêm một hàng. Phương thức "nhớ" cũng hoạt động tương tự trong hệ nhị phân:

```
  1 1 1 1 1      (nhớ)
    0 1 1 0 1
+   1 0 1 1 1
-------------
= 1 0 0 1 0 0
```

Trong ví dụ trên, hai số được cộng với nhau: 01101_2 (13 thập phân) và 10111_2 (23 thập phân). Hàng trên cùng biểu đạt những số nhớ, hoặc mang sang. Bắt đầu bằng cột cuối cùng bên phải, $1 + 1 = 10_2$. 1 được mang sang bên trái, và 0 được viết vào hàng tổng phía dưới, cột cuối cùng bên phải. Hàng thứ hai từ cột cuối cùng bên phải được cộng tiếp theo: $1 + 0 + 1 = 10_2$; Số 1 lại được nhớ lại và mang sang, và số 0 được viết xuống dưới cùng. Cột thứ ba: $1 + 1 + 1 = 11_2$. Lần này 1 được nhớ và mang sang hàng bên cạnh, và 1 được viết xuống hàng dưới cùng. Tiếp tục khai triển theo quy luật trên cho chúng ta đáp án cuối cùng là 100100_2.

Tính trừ

Phép tính trừ theo quy chế tương tự:

0 − 0 = 0
0 − 1 = −1 (mượn)
1 − 0 = 1
1 − 1 = 0

Một đơn vị nhị phân được trừ với một đơn vị nhị phân khác như sau:

```
   *   * * *   (hình sao đánh dấu các cột phải mượn)
  1 1 0 1 1 1 0
-     1 0 1 1 1
----------------
= 1 0 1 0 1 1 1
```

Trừ hai số dương cũng tương tự như "cộng" một số âm với giá trị tương đồng của một số tuyệt đối; máy tính thường dùng ký hiệu Bù 2 để diễn đạt số có giá trị âm. Ký hiệu này loại trừ được nhu cầu bức thiết phải có một phương pháp làm phép trừ biệt lập. Xin xem thêm chi tiết trong chương mục Bù 2.

Tính nhân

Phép tính nhân trong hệ nhị phân cũng tương tự như phương pháp làm trong hệ thập phân. Hai số A và B được nhân với nhau bởi những tích số cục bộ: với mỗi con số ở B, tích của nó với số một con số trong A được tính và viết xuống một hàng mới, mỗi hàng mới phải chuyển dịch vị trí sang bên trái, hầu cho con số cuối cùng ở bên phải đứng cùng cột với vị trí của con số ở trong B đang dùng. Tổng của các tích cục bộ này cho ta kết quả tích số cuối cùng.

Vì chỉ có 2 con số trong hệ nhị phân, nên chỉ có 2 kết quả khả quan trong tích cục bộ:

- Nếu con số trong *B* là 0, tích cục bộ sẽ là 0
- Nếu con số trong *B* là 1, tích cục bộ sẽ là số ở trong *A*

Ví dụ, hai số nhị phân 1011 và 1010 được nhân với nhau như sau:

```
          1 0 1 1   (A)
        × 1 0 1 0   (B)
        ---------
          0 0 0 0   ← tương đương với 0 trong B
   +    1 0 1 1     ← tương đương với 1 trong A
   +  0 0 0 0
   + 1 0 1 1
   ---------------
   = 1 1 0 1 1 1 0
```

Xem thêm Phương pháp làm tính nhân của Booth.

Tính chia

Tính chia nhị phân cũng tương tự như phép chia trong hệ thập phân.

```
         __________
1 0 1  | 1 1 0 1 1
```

Ở đây ta có số bị chia là 11011_2, hoặc 27 trong số thập phân, số chia là 101_2, hoặc 5 trong số thập phân. Cách làm tương tự với cách làm trong số thập phân. Ở đây ta lấy 3 số đầu của số bị chia 110_2 để chia với số chia, tức là 110_2, được 1, viết lên trên hàng kẻ. Kết quả này được nhân với số chia, và tích số được trừ với 3 số đầu của số bị chia. Số tiếp theo là một con số 1 được hạ xuống để tạo nên một dãy số có 3 con số, tương tự với số lượng các con số của số chia:

```
          1
       __________
        1 1 0 1 1 | 1 0 1
      - 1 0 1
        -----
          0 1 1
```

Quy luật trên được lặp lại với những hàng số mới, tiếp tục cho đến khi tất cả các con số trong số bị chia đã được dùng hết:

```
          1 0 1
       __________
        1 1 0 1 1 | 1 0 1
      - 1 0 1
        -----
          0 1 1
        - 0 0 0
          -----
            1 1 1
          - 1 0 1
            -----
              1 0
```

Phân số của 11011_2 chia cho 101_2 là 101_2, như liệt kê phía trên đường kẻ, trong khi số dư còn lại được viết ở hàng cuối là 10_2. Trong hệ thập phân, 27 chia cho 5 được 5, dư 2.

Phép toán thao tác bit trong hệ nhị phân

Mặc dù không liên quan trực tiếp đến sự nhận dạng của các ký tự trong hệ nhị phân, song các dãy số nhị phân có thể được thao tác dùng những toán tử trong lôgic Boole. Khi một dãy số trong hệ nhị phân được thao tác dùng các toán tử này, chúng ta gọi nó là Phép toán thao tác bit. Những thao tác dùng các toán tử AND (tương tự với tác động của chữ "và" trong lôgic, cả hai đơn vị so sánh phải là 1 thì mới cho kết quả 1), OR (tương tự với tác động của chữ "hoặc" trong lôgic, một trong hai đơn vị so sánh là 1 thì cho kết quả là 1), và XOR (nếu 2 bit được so sánh mà khác nhau thì kết quả bằng 1, giống nhau thì bằng 0) có thể được thi hành với từng cặp bit tương đồng trong một cặp số của hai số nhị phân. Thao tác của toán tử lôgic NOT (phép đổi ngược, 0 thành 1 và ngược lại) có thể được thi hành trên từng bit một trong một con số nhị phân. Đôi khi, những phép thao tác này được dùng làm những phương pháp cắt ngắn (làm nhanh) trong các thao tác số học, đồng thời chúng cũng cung cấp những lợi ích khác trong việc xử lý máy tính. Lấy ví dụ, loại bỏ bit cuối cùng (bên phải) trong một số nhị phân (còn được gọi là phép toán chuyển vị nhị phân - *binary shifting*) tương đương với phép chia 2 trong hệ thập phân, vì khi làm như vậy, giá trị của số giảm xuống một nửa. Xin xem thêm Phép toán thao tác bit.

Phương pháp chuyển hệ từ nhị phân sang các hệ khác và ngược lại

Hệ thập phân (cơ số 10)

Phương pháp này có thể áp dụng để chuyển số từ bất cứ gốc nào, song bên cạnh đó còn có những phương thức tốt hơn cho những số là tích số của một mũ, với số nguyên 2, chẳng hạn như hệ bát phân (2^3), và hệ thập lục phân (2^4) liệt kê dưới đây.

Trong các hệ thống số với giá trị của con số được định vị bởi vị trí của nó trong một dãy các ký hiệu con số, những con số ở vị trí thấp hơn, hoặc vị trí ít quan trọng hơn (ít quan trọng hơn là vì khi tính toán các số lớn và sai số xảy ra, mất những số này sẽ không quan trọng và không gây ảnh hưởng lớn đến kết quả tính toán, chẳng hạn số thập phân 10034 có thể được tính tròn số lại thành 10000 trong một thống kê dân số mà không gây ảnh hưởng lớn đến kết quả thống kê), thường có số mũ nhỏ hơn theo hệ số gốc ($2^0 < 2^3$). Số mũ đầu tiên, là một số kém hơn số lượng các chữ số, của một con số nào đó, bởi 1 giá trị. Một con số có 5 chữ số sẽ có số mũ đầu tiên bằng 4. Trong hệ thập phân, gốc của hệ là 10, vậy số cuối cùng ở bên trái của một số có 5 chữ số có số mũ là 4, được thể hiện là ở vị trí 10^4 (chục nghìn). Xem xét ví dụ sau:

$\mathbf{97352}_{10}$ tương đương với:

$$
\begin{array}{l}
\mathbf{9} \times 10^4\ (9 \times 10000 = \mathbf{90000})\ \text{cộng} \\
\mathbf{7} \times 10^3\ (7 \times 1000 = \mathbf{7000})\ \text{cộng} \\
\mathbf{3} \times 10^2\ (3 \times 100 = \mathbf{300})\ \text{cộng} \\
\mathbf{5} \times 10^1\ (5 \times 10 = \mathbf{50})\ \text{cộng} \\
\mathbf{2} \times 10^0\ (2 \times 1 = \mathbf{2})
\end{array}
$$

Phép nhân với gốc của hệ số trở thành một phép tính đơn giản. Vị trí của các chữ số được dịch sang bên trái một vị trí, và số 0 được thêm vào ở phía bên phải của dãy các con số. Ví dụ **9735** nhân 10 bằng **97350**. Một cách định giá trị của một dãy các con số, khi một con số được cộng vào sau con số cuối cùng, bằng cách nhân tất cả các chữ số trước con số cuối cùng ấy với gốc của hệ, trừ số cuối cùng ra, rồi cộng với con số ấy sau cùng. **97352** = **9735** x 10 + **2**. Một ví dụ trong hệ nhị phân là $\mathbf{1101100111}_2 = \mathbf{110110011}_2 \times 2 + \mathbf{1}$. Đây chính là mấu chốt của phép biến đổi hệ số. Trong mỗi bước làm, chúng ta viết xuống con số sẽ phải đổi hệ theo công thức 2 × k + 0 hoặc 2 × k + 1 với một số nguyên k nào đó, và nó sẽ trở thành một số mới mà chúng ta muốn đổi.

$\mathbf{118}_{10}$ tương đương:

$$
\begin{array}{r}
\mathbf{59} \times 2 + \mathbf{0} \\
(\mathbf{29} \times 2 + \mathbf{1}) \times 2 + \mathbf{0} \\
((\mathbf{14} \times 2 + \mathbf{1}) \times 2 + \mathbf{1}) \times 2 + \mathbf{0} \\
(((\mathbf{7} \times 2 + \mathbf{0}) \times 2 + \mathbf{1}) \times 2 + \mathbf{1}) \times 2 + \mathbf{0} \\
((((\mathbf{3} \times 2 + \mathbf{1}) \times 2 + \mathbf{0}) \times 2 + \mathbf{1}) \times 2 + \mathbf{1}) \times 2 + \mathbf{0} \\
(((((\mathbf{1} \times 2 + \mathbf{1}) \times 2 + \mathbf{1}) \times 2 + \mathbf{0}) \times 2 + \mathbf{1}) \times 2 + \mathbf{1}) \times 2 + \mathbf{0}
\end{array}
$$

$$\mathbf{1} \times 2^6 + \mathbf{1} \times 2^5 + \mathbf{1} \times 2^4 + \mathbf{0} \times 2^3 + \mathbf{1} \times 2^2 + \mathbf{1} \times 2^1 + \mathbf{0} \times 2^0$$

1110110_2

Do vậy phương pháp biến đổi một số nguyên, ở hệ thập phân sang hệ nhị phân tương đương, có thể được tiến hành bằng cách chia số này với 2, và những số dư được viết xuống vào hàng (đơn vị) của nó. Kết quả lại tiếp tục được chia với 2, và số dư lại được viết xuống vào hàng (chục) của nó. Phương thức này được tiếp tục nhắc lại cho đến khi thương số của phép chia là 0.

Ví dụ, 118_{10}, trong hệ thập phân là:

Phép tính	Số dư
118 ÷ 2 = 59	0
59 ÷ 2 = 29	1
29 ÷ 2 = 14	1
14 ÷ 2 = 7	0
7 ÷ 2 = 3	1
3 ÷ 2 = 1	1
1 ÷ 2 = 0	1

Lược trình các con số dư theo thứ tự từ dưới lên trên, cho chúng ta một số nhị phân 1110110_2.

Để biến đổi một số nhị phân sang hệ thập phân, chúng làm ngược lại. Bắt đầu từ bên trái, nhân đôi kết quả, rồi cộng con số bên cạnh cho đến khi không còn con số nào nữa. Lấy ví dụ để đổi 110010101101_2 sang hệ thập phân:

Kết quả	Số còn lại
0	110010101101
0 × 2 + 1 = **1**	10010101101
1 × 2 + 1 = **3**	0010101101
3 × 2 + 0 = **6**	010101101
6 × 2 + 0 = **12**	10101101
12 × 2 + 1 = **25**	0101101
25 × 2 + 0 = **50**	101101
50 × 2 + 1 = **101**	01101
101 × 2 + 0 = **202**	1101
202 × 2 + 1 = **405**	101
405 × 2 + 1 = **811**	01
811 × 2 + 0 = **1622**	1
1622 × 2 + 1 = **3245**	

Kết quả là 3245_{10}.

Phần phân số trong một số tự nhiên được biến đổi với cùng một phương pháp, dựa vào phép toán chuyển vị nhị phân để tăng gấp đôi hoặc giảm xuống một nửa giá trị của con số.

Với phân số nhị phân có giá trị "$0{,}11010110101_2$", giá trị của con số đầu tiên của phần thập phân là $\frac{1}{2}$, của con số thứ hai là $\left(\frac{1}{2}\right)^2 = \frac{1}{4}$, vân vân. Vậy nếu chúng ta có giá trị 1 ngay sau dấu phẩy thì giá trị của số thập phân ít nhất phải là $\frac{1}{2}$, và tương tự ngược lại. Nếu chúng ta gấp đôi giá trị của con số đó lên thì giá trị của số phải ít nhất là 1. Điều này khiến chúng ta liên tưởng đến một thuật toán: liên tục nhân đôi con số chúng ta cần biến đổi, ghi lại kết quả nếu kết quả ít nhất là 1, nhưng vứt đi phần số nguyên.

Ví dụ: $\left(\frac{1}{3}\right)_{10}$, trong nhị phân là:

Biến đổi	Kết quả
$\frac{1}{3}$	0,
$\frac{1}{3} \times 2 = \frac{2}{3} < 1$	0,0
$\frac{2}{3} \times 2 = 1\frac{1}{3} \geq 1$	0,01
$\frac{1}{3} \times 2 = \frac{2}{3} < 1$	0,010
$\frac{2}{3} \times 2 = 1\frac{1}{3} \geq 1$	0,0101

Vì vậy phần phân số nhắc đi nhắc lại 0,33<u>3</u> ... tương đương với phần phân số nhắc đi nhắc lại trong hệ nhị phân 0,01<u>01</u>... .

hoặc lấy ví dụ số $0{,}1_{10}$, trong hệ nhị phân là:

Biến đổi	Kết quả
0,1	0,
0.1 × 2 = **0,2** < 1	0,0
0.2 × 2 = **0,4** < 1	0,00
0.4 × 2 = **0,8** < 1	0,000
0.8 × 2 = **1,6** ≥ 1	0,0001
0.6 × 2 = **1,2** ≥ 1	0,00011
0.2 × 2 = **0,4** < 1	0,000110
0.4 × 2 = **0,8** < 1	0,0001100
0.8 × 2 = **1,6** ≥ 1	0,00011001
0.6 × 2 = **1,2** ≥ 1	0,000110011
0.2 × 2 = **0,4** < 1	0,0001100110

Đây cũng là một phân số vô hạn tuần hoàn 0,0001<u>0011</u>... . Có một điều đáng ngạc nhiên là có những phân số thập phân không tuần hoàn nhưng khi chuyển sang nhị phân, nó lại trở thành một phân số tuần hoàn. Chính vì lý do này mà nhiều người thấy ngạc nhiên khi họ kiểm nghiệm thấy phép cộng 0,1 + ... + 0,1 (gồm 10 số hạng) khác với giá trị 1 trong khi giải toán dùng phép toán phân số (*floating point arithmetic*). Thực tế cho thấy, phân số nhị phân chỉ không tuần hoàn khi dạng thập phân của nó là thương của phép chia giữa một số nguyên và lũy thừa cơ số $2(\frac{1}{2}, \frac{1}{4}, \frac{3}{8} ...)$ chứ không phải giữa một số nguyên và bội của $10(\frac{1}{10}, \frac{3}{100} ...)$. Phương pháp biến đổi sau cùng là cách đổi phân số nhị phân sang thập phân. Khó khăn duy nhất là trường hợp của những phân số tuần hoàn, ngoài ra, phương pháp này có thể được thực hiện bằng cách dịch vị trí của dấu thập phân, làm tròn thành số nguyên, biến đổi như cách ở trên, sau đó chia với số mũ của 2 tương ứng trong hệ thập phân. Lấy ví dụ:

$$
\begin{aligned}
x &= \mathbf{1100,101110\underline{011100}...} \\
x \times 2^6 &= \mathbf{1100101110,0111\underline{001110}...} \\
x \times 2 &= \mathbf{11001,0111\underline{001110}...} \\
x \times (2^6 - 2) &= \mathbf{1100010101} \\
x &= (789/62)_{10}
\end{aligned}
$$

Một cách khác để biến đổi hệ nhị phân sang thập phân nhanh hơn, đối với những người đã quen thuộc với hệ thập lục phân, là làm bằng cách gián tiếp, đầu tiên đổi (x trong hệ nhị phân) sang (x trong hệ thập lục phân), rồi đổi (x trong hệ thập lục phân) sang (x hệ thập phân).

Hệ thập lục phân (cơ số 16 hay hệ hexa)

Số nhị phân có thể đổi được sang hệ thập lục phân đôi chút dễ dàng hơn. Sự dễ dàng này là do gốc của hệ thập lục phân (16) là số mũ của gốc hệ nhị phân (2). Cụ thể hơn $16 = 2^4$. Vậy chúng ta phải cần 4 ký tự số trong hệ nhị phân để có thể biểu đạt được một ký tự số trong hệ thập lục phân.

Bảng liệt kê sau đây chỉ ra cho chúng ta từng ký tự số của hệ thập lục phân, cùng với giá trị tương ứng của nó trong hệ thập phân, và một dãy bốn ký tự số tương đương trong hệ nhị phân.

Thập luc phân	Thập phân	Nhị phân
0	0	0000
1	1	0001
2	2	0010
3	3	0011

Thập luc phân	Thập phân	Nhị phân
4	4	0100
5	5	0101
6	6	0110
7	7	0111

Thập luc phân	Thập phân	Nhị phân
8	8	1000
9	9	1001
A	10	1010
B	11	1011

Thập luc phân	Thập phân	Nhị phân
C	12	1100
D	13	1101
E	14	1110
F	15	1111

Để biến đổi từ hệ thập lục phân sang số nhị phân tương đương, chúng ta chỉ đơn giản thay thế những dãy ký tự số tương đương trong hệ nhị phân:

$3A_{16} = 0011\ 1010_2$
$E7_{16} = 1110\ 0111_2$

Để biến đổi một số nhị phân sang hệ thập lục phân tương đương, chúng ta phải phân nhóm các ký tự thành nhóm của bốn ký tự số (nhóm 4 con số). Nếu số lượng của các con số không phải là bội số của 4 (4, 8, 16 ...), thì chúng ta chỉ cần thêm các số **0** vào phía bên trái của con số, còn gọi là phép độn thêm số (*padding*). Chẳng hạn:

1010010_2 = **0**101 0010 nhóm lại cùng với số độn thêm = 52_{16}
11011101_2 = 1101 1101 nhóm lại = DD_{16}

Để biến đổi một số thập lục phân sang số thập phân tương đương, chúng ta nhân mỗi giá trị thập phân của từng con số trong số thập lục phân với số mũ của 16, rồi tìm tổng của các giá trị:

$C0E7_{16} = (12 \times 16^3) + (0 \times 16^2) + (14 \times 16^1) + (7 \times 16^0) = (12 \times 4096) + (0 \times 256) + (14 \times 16) + (7 \times 1) = 49.383_{10}$

Hệ bát phân (cơ số 8)

Số nhị phân cũng có thể được biến đổi sang hệ bát phân một cách dễ dàng, vì bát phân dùng gốc 8, và cũng là số mũ của 2 (chẳng hạn 2^3, vậy số bát phân cần 3 ký tự số nhị phân để biểu đạt trọn vẹn một số bát phân). Sự tương ứng giữa các số bát phân và các số nhị phân cũng giống như sự tương đương với 8 con số đầu tiên của hệ thập lục phân, như đã liệt kê trên bảng trước đây. Số nhị phân 000 tương đương với số bát phân 0, số nhị phân 111 tương đương với số bát phân 7, và tương tự.

Bát phân	Nhị phân
0	000
1	001
2	010
3	011

Bát phân	Nhị phân
4	100
5	101
6	110
7	111

Phương pháp đổi bát phân sang nhị phân cũng tương tự như cách làm đối với hệ thập lục phân:

$65_8 = 110\ 101_2$
$17_8 = 001\ 111_2$

và từ nhị phân sang bát phân:

$101100_2 = 101\ 100_2$ nhóm lại = 54_8
10011_2 = **0**10 011_2 nhóm lại với số độn thêm = 23_8

từ bát phân sang thập phân:

$65_8 = (6 \times 8^1) + (5 \times 8^0) = (6 \times 8) + (5 \times 1) = 53_{10}$
$127_8 = (1 \times 8^2) + (2 \times 8^1) + (7 \times 8^0) = (1 \times 64) + (2 \times 8) + (7 \times 1) = 87_{10}$

Biểu thị số thực

Những số không phải là số nguyên có thể được biểu thị bằng số mũ âm, và dùng dấu tách biệt phân số (dấu "phẩy") làm cho chúng biệt lập khỏi các con số khác. Lấy ví dụ, số nhị phân $11{,}01_2$ có nghĩa là:

$\mathbf{1} \times 2^1$ $(1 \times 2 = \mathbf{2})$ cộng
$\mathbf{1} \times 2^0$ $(1 \times 1 = \mathbf{1})$ cộng
$\mathbf{0} \times 2^{-1}$ $(0 \times ½ = \mathbf{0})$ cộng
$\mathbf{1} \times 2^{-2}$ $(1 \times ¼ = \mathbf{0{,}25})$

Tổng số là 3,25 trong hệ thập phân.

Tất cả các nhị thức số hữu tỷ $\frac{p}{2^a}$ đều có một số nhị phân hữu hạn— Biểu thức nhị phân có một dãy số giới hạn sau điểm chia phân số (*radix point*). Các số hữu tỷ khác cũng có biểu thị nhị phân (*binary representation*), song thay vì là một dãy số hữu hạn, một loạt dãy các con số hữu hạn được lặp đi lặp lại, theo một tiến trình vô hạn. Chẳng hạn:

$$\frac{1_{10}}{3_{10}} = \frac{1_2}{11_2} = 0.0101010\underline{01}\ldots_2$$

$$\frac{12_{10}}{17_{10}} = \frac{1100_2}{10001_2} = 0.10110100\ 10110100\ \underline{10110100}\ldots_2$$

Hiện tượng biểu thị nhị phân cho một phân thức có thể là một dãy số hữu hạn (*terminating*) hoặc là một dãy số vô hạn cũng được thấy trong các hệ số dựa trên cơ số khác (*radix-based numeral systems*). Xem thêm phần giải thích như trong bản phân tích về hệ thập phân. Một biểu hiện tương tự các cách biểu thị phân số hữu hạn, dựa vào thực tế 0,111111... là tổng của cấp số nhân (*geometric series*) $2^{-1} + 2^{-2} + 2^{-3} + \ldots$ tức là 1.

Số nhị phân vừa không phải là số hữu hạn, cũng không phải là số vô hạn thì được gọi là số vô tỷ (*irrational number*). Chẳng hạn:

- 0.10100100010000100000100.... dãy số có mô hình nhắc lại, nhưng dãy số mô hình nhắc lại này không có giới hạn về số lượng, cho nên được gọi là số vô tỷ
- 1.0110101000001001111001100110011111110... là một biểu thức nhị phân của $\sqrt{2}$ (căn bậc hai của 2), một số vô tỷ khác. Số vô tỷ này không có mô hình nhắc lại có thể nhận dạng, song để chứng minh rằng $\sqrt{2}$ là một số vô tỷ thì chúng ta phải đòi hỏi bằng chứng hơn thế này nữa. Xin xem trong bài về số vô tỷ để được rõ thêm.

Tếu nhị phân

- *Binary is as easy as 1, 10, 11.* (Nhị phân dễ như là 1, 10, 11 vậy.)
- *I'm just 10 people short of a threesome!* (Tôi chỉ thiếu mỗi 10 người để được một nhóm 3 hú hí.)
- *There are 10 kinds of people in the world—those who understand binary, and those who don't."* (Chỉ có 10 loại người trên thế gian này mà thôi, loại hiểu nhị phân và loại không hiểu nhị phân..)
- *11 is the magic number.* (11 là một con số kỳ diệu.)

Nhị phân sang chữ cái

1. a: 01100001
2. b: 01100010
3. c: 01100011
4. d: 01100100
5. e: 01100101
6. f: 01100110
7. g: 01100111
8. h: 01101000
9. i: 01101001
10. j: 01101010
11. k: 01101011
12. l: 01101100
13. m: 01101101
14. n: 01101110
15. o: 01101111
16. p: 01110000
17. q: 01110001
18. r: 01110010
19. s: 01110011
20. t: 01110100
21. u: 01110101
22. v: 01110110
23. w: 01110111
24. x: 01111000
25. y: 01111001
26. z: 01111010

Xem thêm

- Hệ thập lục phân
- Hệ thập phân

Chú thích

1. ^ Aiton, Eric J. (1985), *Leibniz: A Biography*, Taylor & Francis, tr. 245–8, ISBN 0-85274-470-6
2. ^ Boole, George (2009) [1854]. *An Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probabilities* (http://www.gutenberg.org/files/15114/15114-pdf.pdf) . New York: Cambridge University Press. ISBN 9781108001533.
3. ^ Shannon, Claude Elwood (1940). *A symbolic analysis of relay and switching circuits* (http://dspace.mit.edu/handle/1721.1/11173). Cambridge: Massachusetts Institute of Technology.
4. ^ "National Inventors Hall of Fame – George R. Stibitz" (http://www.invent.org/hall_of_fame/140.html). 20 tháng 8 năm 2008. Truy cập ngày 5 tháng 7 năm 2010.
5. ^ "George Stibitz : Bio" (http://stibitz.denison.edu/bio.html). Math & Computer Science Department, Denison University. 30 tháng 4 năm 2004. Truy cập ngày 5 tháng 7 năm 2010.

6. ^ "Pioneers – The people and ideas that made a difference – George Stibitz (1904–1995)" (http://www.kerryr.net/pioneers/stibitz.htm). Kerry Redshaw. 20 tháng 2 năm 2006. Truy cập ngày 5 tháng 7 năm 2010.
7. ^ "George Robert Stibitz – Obituary" (http://ei.cs.vt.edu/~history/Stibitz.html). Computer History Association of California. 6 tháng 2 năm 1995. Truy cập ngày 5 tháng 7 năm 2010.

Tài liệu tham khảo

- Sanchez, Julio; Canton, Maria P. (2007), Microcontroller programming: the microchip PIC, Boca Raton, FL: CRC Press, p. 37, ISBN 0849371899

Liên kết ngoài

- Converting Decimal, Hexadecimal, text, numbers, and ascii to binary and back (http://www.binaryguide.com/)
- Binary numeral system (http://www.ee.up.ac.za/main/_media/en/undergrad/subjects/emk310/wiki_binary.pdf)
- Indian mathematics (http://www-history.mcs.st-andrews.ac.uk/history/Projects/Pearce/index.html)
- Binary System (http://www.cut-the-knot.org/do_you_know/BinaryHistory.shtml) at cut-the-knot
- Conversion of Fractions (http://www.cut-the-knot.org/blue/frac_conv.shtml) at cut-the-knot
- Converting Hexadecimal to Decimal (http://www.permadi.com/tutorial/numHexToDec/)
- Converting Decimal to Hexadecimal (http://www.permadi.com/tutorial/numDecToHex/)
- Weisstein, Eric W., "Binary (http://mathworld.wolfram.com/Binary.html)" từ MathWorld.

Lấy từ "http://vi.wikipedia.org/w/index.php?title=Hệ_nhị_phân&oldid=13744551"
Thể loại: Hệ đếm | Số học máy tính | Số học sơ cấp
