Sistema Binário

Existem duas maneiras de representar uma informação eletrônica: analogicamente ou digitalmente. Uma música qualquer, por exemplo, gravada em uma fita K-7 é uma forma analógica de gravação. Certamente se os computadores trabalhassem com dados analógicos seriam passíveis de erros, pois qualquer interferência, por mínima que fosse, causaria alterações nos dados processados e consequentemente nos resultados.
O sistema digital armazena qualquer informação na forma de uma seqüência de valores positivos e negativos, ou seja, na forma de uns e zeros. O sistema binário torna o computador confiável, pois num sistema digital não há possibilidade de um valor diferente de 0 e 1, portanto devido à simplicidade dos cálculos, a velocidade de processamento também se torna maior.
Cada valor binário é chamado de "bit", simplificação de "**bi**nary digi**t**" ou "dígito binário". Um conjunto de 8 bits forma um byte.

Base de um Sistema de Numeração
Sistema binário de numeração
No sistema binário somente dois algarismos são utilizados:

O e 1

Binário	Decimal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Quando os símbolos 0 e 1 são usados para representar números binários, cada símbolo é chamado de dígito binário, ou simplesmente de bit.
O número binário 101_2 é chamado de número binário de três dígitos ou de número binário de três bits. As regras de formação de um número em binário são as mesmas que foram utilizadas para o sistema decimal, com exceção de que no sistema binário a base é igual a 2. O resultado da soma é igual ao equivalente no sistema decimal.

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

Sistema octal de numeração
O sistema de números octais tem a base igual a 8, o que indica o uso de oito algarismos, sendo todos equivalentes aos do sistema decimal. Os algarismos usados neste sistema são:

0, 1, 2, 3, 4, 5, 6, 7

Sistema hexadecimal de numeração

O sistema de números hexadecimal usa a base 16, o que indica o uso de dezesseis símbolos, sendo os mesmos símbolos usados no sistema decimal mais as seis primeiras letras do alfabeto. Os algarismos utilizados no sistema hexadecimal são:

0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

As seis primeiras letras do alfabeto, utilizadas no sistema hexadecimal como números representam as seguintes quantidades: A=10 B=11 C=12 D=13 E=14 F=15 Representação em Octal e Hexadecimal O sistema de números hexadecimais é muito usado em projetos de hardware e software, já que estes representam grupos de dígitos binários, facilitando a representação de códigos binários. É usual representar quantidades usando sistemas em potências do binário, para reduzir o número de algarismos da representação. No sistema octal (base 8), três bits são representados por apenas um algarismo octal (de 0 a 7). No sistema hexadecimal (base 16), quatro bits são representados por apenas um algarismo hexadecimal (de 0 a F).

Conversão entre bases

Conversão binário-octal

Como $2^3 = 8$, separando os bits de um número binário em grupos de três bits (começando sempre da direita para a esquerda para a parte inteira) e convertendo cada grupo de três bits para seu equivalente em octal, teremos a representação do número em octal.

Exemplo:

$10101001_2 = (x)_8$

Separando em grupos de 3 bits

10101001_2 =	010	101	001

Sabemos que $010_2 = 2_8$; $101_2 = 5_8$; $001_2 = 1_8$ e portanto, $10101001_2 = 251_8$

2	5	1

$10101001_2 = 251_8$

Conversão octal-binário

A conversão inversa se faz convertendo cada dígito octal por um grupo de três dígitos binários.

Exemplo:

$1765_8=(x)_2$

1	7	6	5
001	111	110	101

Portanto, $1765_8 = 001111110101_2$

Conversão binário-hexadecimal

Como $2^4=16$, basta separarmos em grupos de 4 bits (começando sempre da direita para a esquerda para a parte inteira) e converter cada grupo para seu equivalente em hexadecimal.

Exemplo:

$11010101101_2 = (x)_{16}$

Separando em grupos de 4 bits

$11010101101_2 = 0110\ 1010\ 1101$

Cada grupo de 4 bits é então substituído pelo equivalente hexadecimal, então:

$110_2 = 6_{16}$; $1010_2 = A_{16}$;$1101_2 = D_{16}$

Portanto, $11010101101_2 = 6AD_{16}$.

Conversão hexadecimal-binário

A conversão inversa se faz convertendo cada dígito hexadecimal por grupos de dígitos de quatro dígitos binários.

exemplo

$3F5_{16} = (x)\ 2$

3	F	5
0011	1111	0101

Portanto,

$3F5_{16} = 1111110101_2$

Conversão de Números da Base 10 (decimal) para binário

Parte inteira

O número decimal deverá ser dividido sucessivas vezes pela base, sendo que o resto de cada divisão será igual a um algarismo do novo número.

Exemplos:

19 |2
 1   9|2
  1 4|2
   0 2|2
    0 1|2
     1 0 $19_{10} = 10011_2$

$10011_2 = (x)_{10} \rightarrow 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$

$16 + 0 + 0 + 2 + 1 = 19$

30|2
 0 15|2
  1 7|2
   1 3|2
    1 1|2
     1 0 $30_{10} = 11110_2$

$11110_2 = (x)_{10} \rightarrow 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

$16 + 8 + 4 + 2 + 0 = 30$

Conversão de Números da Base 8 (octal) para decimal
Converter 345_8 em decimal.
$345_8 = 3x8^2 + 4x8^1 + 5x8^0$
$345_8 = 192 + 32 + 5 = 229_{10}$

Converter 477_8 em decimal.
$477_8 = 4x8^2 + 7x8^1 + 7x8^0$
$477_8 = 256 + 56 + 7 = 319_{10}$

Conversão de Números da Base 10 (decimal) para octal
Converter 90_{10} para octal.
90|8
  2 11|8
      3 1|8
          1 0 $90_{10} = 132_8$

Converter 128_{10} para octal.
128|8
    0 16|8
        0 2|8
            2 0 $128_{10} = 200_8$

Conversão de Números da Base 16 (hexadecimal) para decimal
Converter $2D_{16}$ em decimal.
$2D_{16} = 2x16^1 + 13x16^0 = 32 + 13 = 45$

Converter $1C3_{16}$ em decimal.
$1C3_{16} = 1x16^2 + 12x16^1 + 3x16^0 =$
$256 + 192 + 3 = 451_{10}$.

Conversão de Números da base 10 (decimal) para hexadecimal
Converter 1000_{10} em hexadecimal.
1000|16
      8 62|16
          14 3|16
                3 0 $1000_{10} = 3E8_{16}$

Converter 120_{10} em hexadecimal
120|16
    8 7|16
        7 0 $120_{10} = 78_{16}$

Aritmética em Binário
A tabuada da soma aritmética em binário é muito simples. São poucas regras:
0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 0 (e "vai 1" para o dígito de ordem superior)
1 + 1 + 1 = 1 (e "vai 1" para o dígito de ordem superior)

Exemplo:
Efetuar 011100 + 011010
Observações:
1) Lembre-se: somam-se as colunas da direita para a esquerda, tal como uma soma em decimal.
2) No exemplo, são usadas, em seqüência, da direita para a esquerda, todas as regrinhas acima.
3) Na primeira linha, em azul, é indicado o "vai um".
4) Por simplicidade, no exemplo estamos considerando os dois números positivos.

Solução:
11-----> "vai um"
011100
011010+

110110

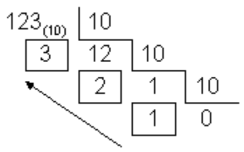
Subtração em binário
0 - 0 = 0
0 - 1 = 1 ("vem um do próximo")
1 - 0 = 1
1 - 1 = 0

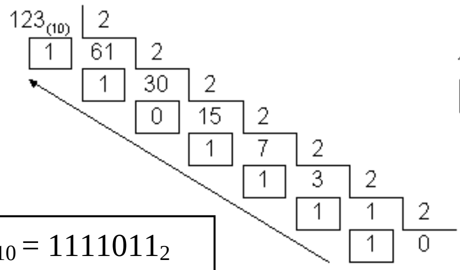
Como é impossível tirar 1 de zero, o artifício é "pedir emprestado" 1 da casa de ordem superior. Ou seja, na realidade o que se faz é subtrair 1 de 10 e encontramos 1 como resultado, devendo então subtrair 1 do dígito de ordem superior (aquele 1 que se "pediu emprestado"). Vamos lembrar que esse algoritmo é exatamente o mesmo da subtração em decimal a que já estamos acostumados desde o curso primário.
Exemplo:
Efetuar 111100 + 011010
1) Lembre-se: subtraem-se as colunas da direita para a esquerda, tal como uma subtração em decimal.
2) No exemplo, são usadas, em seqüência, da direita para a esquerda, todas as regrinhas acima.
3) Na primeira linha, em vermelho, é indicado o "vem um".
4) Por simplicidade, no exemplo estamos considerando os dois números positivos.

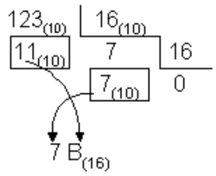
Solução:
---1--- "vem um"
11100
01010-

10010

Da base decimal para outras Bases







123_{10}

$123_{10} = 1111011_2$

$123_{10} = 7B_{16}$

Exercícios

1) Faça as conversões de base como estabelecido em cada um:

$943_{10} = (x)_2$

943 | 2
1 471 | 2
1 235 | 2
1 117 | 2
1 58 | 2
0 29 | 2
1 14 | 2
0 7 | 2
1 3 | 2
1 1 | 2
1 0

$(943)_{10} = (1110101111)_2$

$101001111_2 = (x)_8$ → 101 001 111
5 1 7 = 517_8

$100100111_2 = (x)_8$ → 100 100 111
4 4 7 = 447_8

$1010011_2 = (x)_8$ → 1 010 011 ou 001 010 011
1 2 3 = 123_8

$327_8 = (x)_2$ → 3 2 7
011 010 111 = 011010111_2

$(3964)_{10} = (x)_8$

3964 | 8
4 495 | 8
7 61 | 8

5 7

$(3964)_{10} = (7574)_8$

$(943)_{10} = (x)_8$

943 8

7 117 8

5 14 8

6 1

$(943)_{10} = (1657)_8$

2) Efetue:

a) 1010 + 111 =

binário decimal

$$\begin{array}{r} 1010 \\ +\,0111 \\ \hline 10001 \end{array} \qquad \begin{array}{r} 10 \\ +\ 7 \\ \hline 17 \end{array}$$

b) 1010 + 101 =

binário decimal

$$\begin{array}{r} 1010 \\ +\,0101 \\ \hline 1111 \end{array} \qquad \begin{array}{r} 10 \\ +\ 5 \\ \hline 15 \end{array}$$