

# Eletrônica Digital

**Sistema de Numeração e Conversão  
entre Sistemas.**

Prof. Rômulo Calado Pantaleão Camara

Carga Horária: 60h

# Representação da Informação

- ✓ Um dispositivo eletrônico, armazena e movimenta as informações internamente sob forma eletrônica; tudo o que faz é reconhecer dois estados físicos distintos, produzidos pela eletricidade, pela polaridade magnética ou pela luz refletida - em essência, eles sabem dizer se um "interruptor" está **ligado** ou **desligado**.
- ✓ O computador, por ser uma máquina eletrônica, só consegue processar duas informações: a **presença** ou **ausência** de energia.
- ✓ Para que a máquina pudesse representar eletricamente todos os símbolos utilizados na linguagem humana, seriam necessários mais de 100 diferentes valores de tensão (ou de corrente).

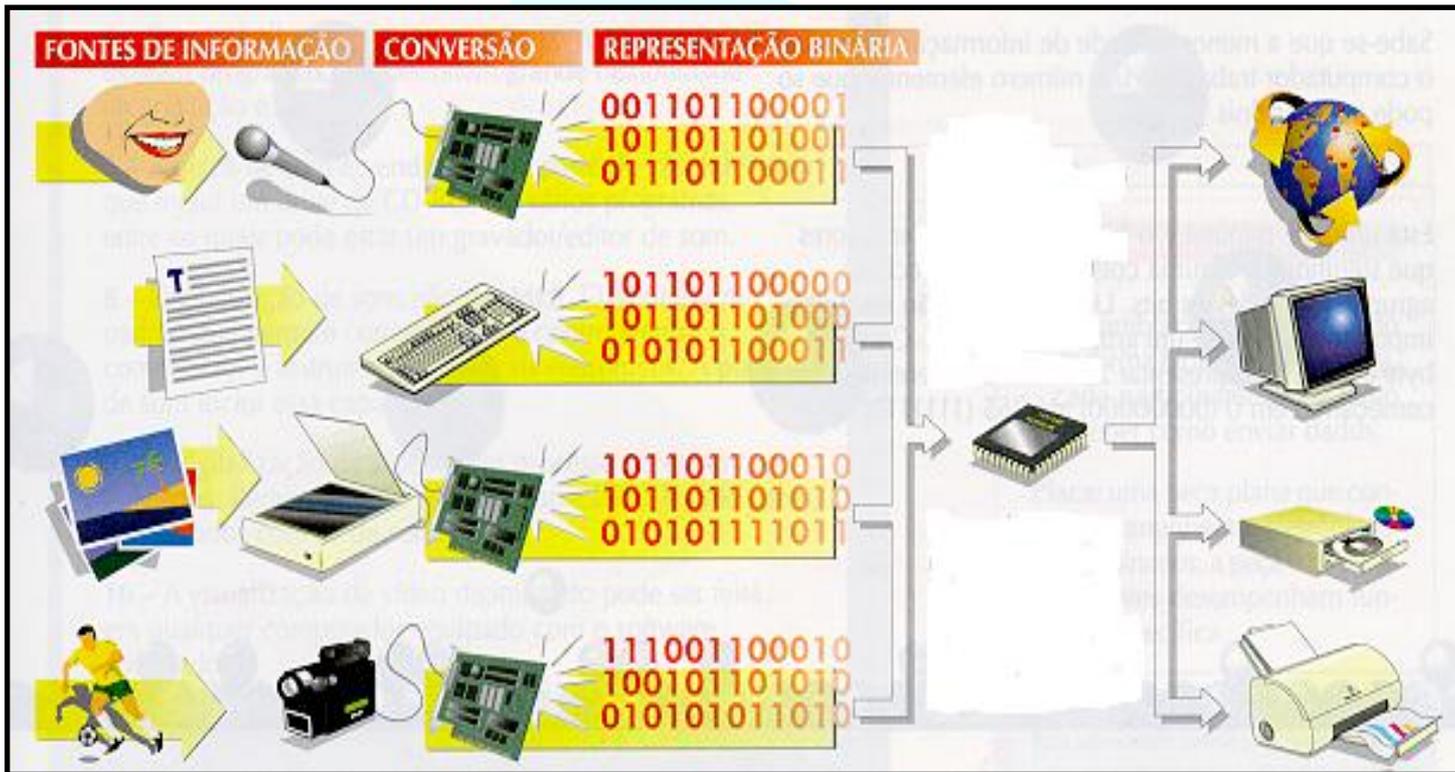
# Representação da Informação

## Tipos de grandezas

- ✓ **Analógica**  $\equiv$  contínua
- ✓ **Digital**  $\equiv$  discreta (passo a passo)
- ✓ **Mundo analógico** - Trabalha com sinais elétricos de infinitos valores de tensão e corrente (modelo continuamente variável, ou **analogia**, do que quer que estejam medindo).
- ✓ **Mundo digital** - Trabalha com dois níveis de sinais elétricos: **alto** e **baixo**. Representam dados por meio de um símbolo facilmente identificado (**dígito**).

# Representação da Informação

- ✓ Como os computadores modernos representam as informações?



# Representação da Informação

- ✓ Para sistema digital, tudo são números.
- ✓ **Sistema Digital** ⇒ Normalmente a informação a ser processada é de forma numérica ou texto ⇒ codificada internamente através de um **código numérico**.
- ✓ Código mais comum ⇒ **BINÁRIO**

Por que é utilizado o sistema binário ?

# Representação da Informação

- ✓ Para sistema digital, tudo são números.
- ✓ **Sistema Digital** ⇒ Normalmente a informação a ser processada é de forma numérica ou texto ⇒ codificada internamente através de um **código numérico**.
- ✓ Código mais comum ⇒ **BINÁRIO**

Por que é utilizado o sistema binário ?

# Representação da Informação

- ✓ Como os computadores representam as informações utilizando apenas dois estados possíveis - eles são totalmente adequados para números binários.

**0 – desligado**

**1 – ligado**

- ✓ Número binário no computador: **bit** [de “Binary dig**IT**”]
  - A unidade de informação.
  - Uma quantidade computacional que pode tomar um de dois valores, tais como verdadeiro e falso ou 1 e 0, respectivamente (lógica positiva).

Um bit está ligado (*set*) quando vale 1, desligado ou limpo (*reset* ou *clear*) quando vale 0; comutar, ou inverter (*toggle* ou *invert*) é passar de 0 para 1 ou de 1 para 0. (lógica positiva)

# Representação da Informação

- ✓ Um bit pode representar apenas 2 símbolos (0 e 1)
- ✓ **Necessidade** - unidade maior, formada por um conjunto de bits, para representar números e outros símbolos, como os caracteres e os sinais de pontuação que usamos nas linguagens escritas.
- ✓ Unidade maior (**grupo de bits**) - precisa ter bits suficientes para representar todos os símbolos que possam ser usados:
  - dígitos numéricos,
  - letras maiúsculas e minúsculas do alfabeto,
  - sinais de pontuação,
  - símbolos matemáticos e assim por diante.

# Representação da Informação

✓ Necessidade:

|                                       |            |
|---------------------------------------|------------|
| Caracteres alfabéticos maiúsculos     | <b>26</b>  |
| Caracteres alfabéticos minúsculos     | <b>26</b>  |
| Algarismos                            | <b>10</b>  |
| Sinais de pontuação e outros símbolos | <b>32</b>  |
| Caracteres de controle                | <b>24</b>  |
| Total                                 | <b>118</b> |

# Representação da Informação

✓ Capacidade de Representação:

| Bits     | Símbolos   |
|----------|------------|
| 2        | 4          |
| 3        | 8          |
| 4        | 16         |
| 5        | 32         |
| 6        | 64         |
| 7        | 128        |
| <b>8</b> | <b>256</b> |
| 9        | 512        |
| 10       | 1024       |

# Representação da Informação

## ✓ BYTE (BInary TErm)

- Grupo ordenado de 8 bits, para efeito de manipulação interna mais eficiente
- Tratado de forma individual, como unidade de armazenamento e transferência.
- Unidade de memória usada para representar um caractere.

Com 8 bits, podemos arranjar 256 configurações diferentes: dá para 256 caracteres, ou para números de 0 a 255, ou de -128 a 127, por exemplo.

O termo *bit* apareceu em 1949, inventado por John Tukey, um pioneiro dos computadores. Segundo Tukey, era melhor que as alternativas *bigit* ou *binit*.

O termo *byte* foi criado por Werner Buchholz em 1956 durante o desenho do computador IBM Stretch. Inicialmente era um grupo de 1 a 6 *bits*, mas logo se transformou num de 8 *bits*. A palavra é uma mutação de *bite*, para não confundir com *bit*.

# Representação da Informação

- ✓ Todas as letras, números e outros caracteres são codificados e decodificados pelos equipamentos através dos bytes que os representam, permitindo, dessa forma, a comunicação entre o usuário e a máquina.
- ✓ Sistemas mais importantes desenvolvidos para representar símbolos com números binários (bits):
  - **EBCDIC** (*Extended Binary Coded Decimal Interchange Code* - Código Ampliado de Caracteres Decimais Codificados em Binário para o Intercâmbio de Dados).
  - **ASCII** (*American Standard Code for Information Interchange* - Código Padrão Americano para o Intercâmbio de Informações).
  - **UNICODE** (Unicódigo).

# Representação da Informação

✓ Parte do conjunto de caracteres ASCII:

| <b>Binário</b> | <b>Caractere</b> |
|----------------|------------------|
| 0100 0001      | A                |
| 0100 0010      | B                |
| 0110 0001      | a                |
| 0110 0010      | b                |
| 0011 1100      | <                |
| 0011 1101      | =                |
| 0001 1011      | ESC              |
| 0111 1111      | DEL              |

# Representação da Informação

- ✓ A conversão de dados em informações, e estas novamente em dados, é uma parte tão fundamental em relação ao que os computadores fazem que é preciso saber como a conversão ocorre para compreender como o computador funciona.
- ✓ Infelizmente os computadores não usam nosso sistema de numeração.

# Sistema de Numeração

- ✓ Conjunto de símbolos utilizados para representação de quantidades e de regras que definem a forma de representação.
- ✓ Cada sistema de numeração é apenas um método diferente de representar quantidades. As quantidades em si não mudam; mudam apenas os símbolos usados para representá-las.
- ✓ A quantidade de algarismos disponíveis em um dado sistema de numeração é chamada de **base**.
- ✓ Representação numérica mais empregada: *notação posicional*.

# Sistema de Numeração

| <b>Sistema</b>     | <b>Base</b> | <b>Algarismos</b>                      |
|--------------------|-------------|--|
| <b>Binário</b>     | <b>2</b>    | <b>0,1</b>                             |
| <b>Ternário</b>    | <b>3</b>    | <b>0,1,2</b>                           |
| <b>Octal</b>       | <b>8</b>    | <b>0,1,2,3,4,5,6,7</b>                 |
| <b>Decimal</b>     | <b>10</b>   | <b>0,1,2,3,4,5,6,7,8,9</b>             |
| <b>Duodecimal</b>  | <b>12</b>   | <b>0,1,2,3,4,5,6,7,8,9,A,B</b>         |
| <b>Hexadecimal</b> | <b>16</b>   | <b>0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F</b> |

Como os números representados em base 2 são muito extensos e, portanto, de difícil manipulação visual, costuma-se representar externamente os valores binários em outras bases de valor mais elevado (octal ou hexadecimal). Isso permite maior compactação de algarismos e melhor visualização dos valores.

# Sistema de Numeração

## Padrões de Representação

- ✓ Letra após o número para indicar a base;
- ✓ Número entre parênteses e a base como um índice do número.
- ✓ **Exemplo:**
  - Sistema Decimal - 1234D ou  $(1234)_{10}$  ou  $1234_{10}$

# Sistema de Numeração

## Decimal

- ✓ Sistema mais utilizado.
- ✓ Apareceu naturalmente no aprendizado de contagem (dez dedos).
- ✓ 10 símbolos para representar quantidades.

0 1 2 3 4 5 6 7 8 9

# Sistema de Numeração

## Decimal

- ✓ Também chamado de sistema de *base 10* é um sistema posicional, no qual o valor de cada dígito depende de sua posição no número: **unidade**, **dezena**, (dez unidades), **centena** (cem unidades), **milhar** (mil unidades), **dezena de milhar**, **centena de milhar**, etc.
- ✓ Exemplo: 1234 é composto por **4** unidades, **3** dezenas, **2** centenas e **1** milhar, ou  $1000+200+30+4 = 1234$ ;

# Sistema de Numeração

## Sistema Binário

- ✓ Também chamado de sistema de *base 2* é um sistema posicional, no qual o valor de cada dígito é nomeado de bit.

0 e 1

- ✓ Segue as regras do sistema decimal - válidos os conceitos de **peso** e **posição**. Posições não têm nome específico.
- ✓ Cada algarismo é chamado de **bit**. Exemplo:  $101_2$
- ✓ **Expressão oral** - diferente dos números decimais.
  - Caractere mais à esquerda - *Most-Significative-Bit* - "**MSB**".
  - Caractere mais à direita - *Least-Significative-Bit* - "**LSB**".

# Sistema de Numeração

## Contagem Binário

✓ Em operações binários, circuitos restringem a um número de bits específico, portanto, a contagem é restrita ao número de bits do sistema considerado;

● Exemplo: números de 4 bits

• O “1” muda a cada contagem

• O “2” muda a cada duas contagens

• O “4” muda a cada quatro contagens

• O “8” muda a cada oito contagens

• Com N bits, conta-se  $2^N$  números, com a última contagem em  $2^N - 1$

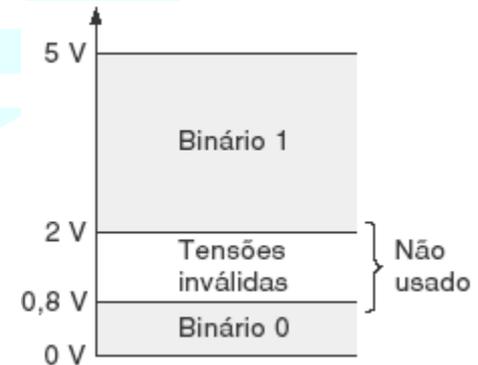
| Pesos → | $2^3 = 8$ | $2^2 = 4$ | $2^1 = 2$ | $2^0 = 1$ |   | Número decimal equivalente |
|---------|-----------|-----------|-----------|-----------|---|----------------------------|
|         | 0         | 0         | 0         | 0         | → | 0                          |
|         | 0         | 0         | 0         | 1         | → | 1                          |
|         | 0         | 0         | 1         | 0         |   | 2                          |
|         | 0         | 0         | 1         | 1         |   | 3                          |
|         | 0         | 1         | 0         | 0         |   | 4                          |
|         | 0         | 1         | 0         | 1         |   | 5                          |
|         | 0         | 1         | 1         | 0         |   | 6                          |
|         | 0         | 1         | 1         | 1         |   | 7                          |
|         | 1         | 0         | 0         | 0         |   | 8                          |
|         | 1         | 0         | 0         | 1         |   | 9                          |
|         | 1         | 0         | 1         | 0         |   | 10                         |
|         | 1         | 0         | 1         | 1         |   | 11                         |
|         | 1         | 1         | 0         | 0         |   | 12                         |
|         | 1         | 1         | 0         | 1         |   | 13                         |
|         | 1         | 1         | 1         | 0         | → | 14                         |
|         | 1         | 1         | 1         | 1         | → | 15                         |

↑  
LSB

# Sistema de Numeração

## Representação de Quantidades Binárias

- ✓ Quantidades binárias podem ser representadas por qualquer dispositivo que tenha dois estados;
- ✓ Exemplos: chave (liga-desliga), CD-ROM (furos ou "não-furos"), transistor (corte ou saturação);
- ✓ Em sistemas digitais, bits são tensões (ou correntes) presentes nas entradas e saídas - ex.: 0V ("0") ou 5V ("1");
- ✓ Bits são, na verdade, faixas de tensão, diferentes de sinais analógicos;



# Sistema de Numeração

## Sistema Octal

- ✓ Também chamado de sistema de *base 8* é um sistema posicional;

0 1 2 3 4 5 6 7

- ✓ Exemplo:  $563_8$
- ✓ **Expressão oral** - similar ao sistema binário.

# Sistema de Numeração

## Sistema Hexadecimal

- ✓ Também chamado de sistema de *base 16* é um sistema posicional.
- ✓ Possui 16 símbolos (algarismos) para representar qualquer quantidade.

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

- ✓ Uso das letras - **facilidade de manuseio.**
- ✓ Exemplo:  $FA3_{16}$
- ✓ **Expressão oral** - similar ao sistema binário.

# Sistema de Numeração

Ao trabalhar com sistemas de numeração, em qualquer base, deve-se observar o seguinte:

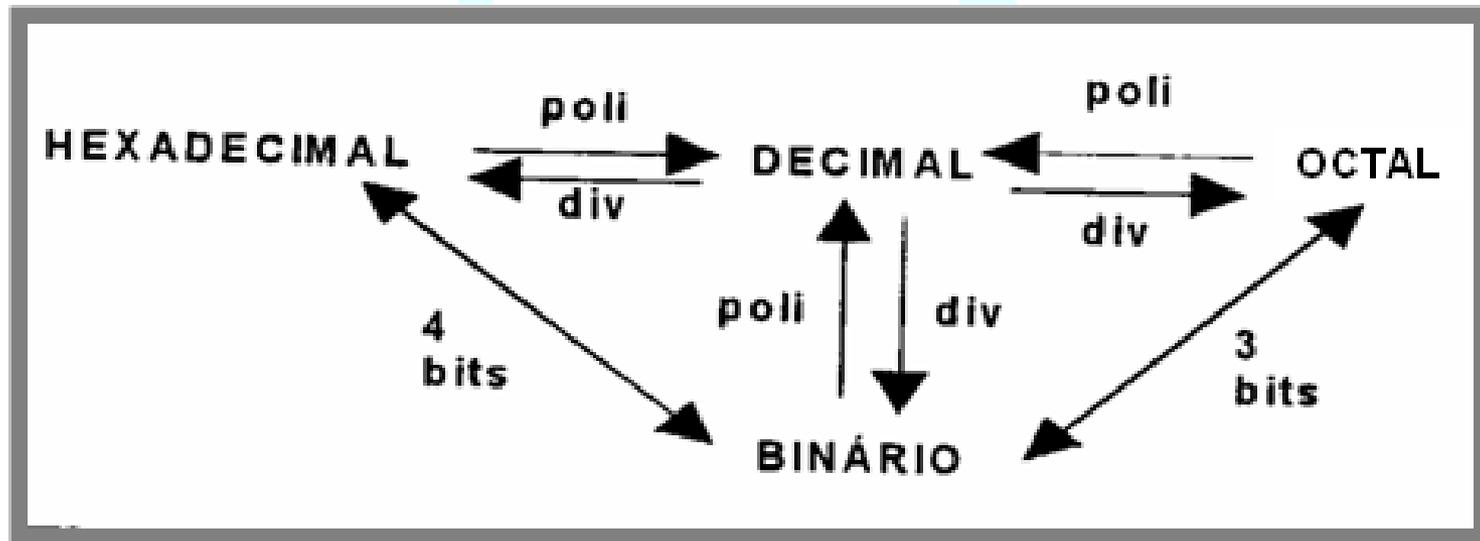
- ✓ O número de dígitos usado no sistema é igual à base.
- ✓ O maior dígito é sempre menor que a base.
- ✓ O dígito mais significativo está à esquerda, e o menos significativo à direita
- ✓ Um “vai-um” de uma posição para outra tem um peso igual a uma potência da base.
- ✓ Em geral se toma a base decimal como referência.

# Sistema de Numeração

| Decimal | Binário | Octal | Hexadecimal |
|---------|---------|-------|-------------|
| 0       | 0       | 0     | 0           |
| 1       | 1       | 1     | 1           |
| 2       | 10      | 2     | 2           |
| 3       | 11      | 3     | 3           |
| 4       | 100     | 4     | 4           |
| 5       | 101     | 5     | 5           |
| 6       | 110     | 6     | 6           |
| 7       | 111     | 7     | 7           |
| 8       | 1000    | 10    | 8           |
| 9       | 1001    | 11    | 9           |
| 10      | 1010    | 12    | A           |
| 11      | 1011    | 13    | B           |
| 12      | 1100    | 14    | C           |
| 13      | 1101    | 15    | D           |
| 14      | 1110    | 16    | E           |
| 15      | 1111    | 17    | F           |
| .       | .       | .     | .           |
| .       | .       | .     | .           |
| .       | .       | .     | .           |

# Conversão entre Sistemas de Numeração

- ✓ Procedimentos básicos: - divisão  
(números inteiros)
- polinômio
- agrupamento de bits



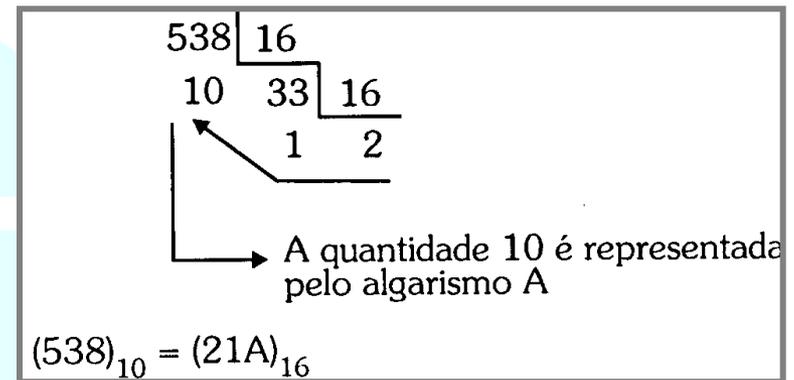
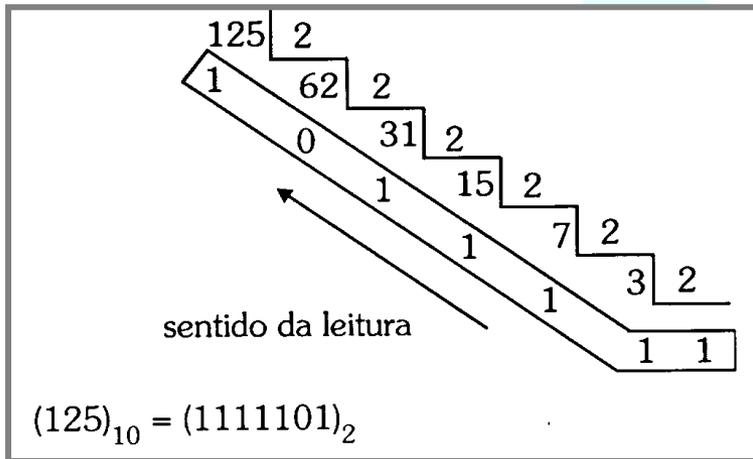
# Conversão entre Sistemas de Numeração

- ✓ **Divisão** (Decimal → outro sistema)
  - Divisão inteira (do quociente) sucessiva pela base, até que quociente seja menor do que a base.
  - Valor na base = composição do **último quociente** (MSB) com **restos** (primeiro resto é bit menos significativo - LSB)
  - Dividir o número por  $b$  (base do sistema) e os resultados consecutivas vezes.

# Conversão entre Sistemas de Numeração

✓ Ex.:  $(125)_{10} = (?)_2$

$(538)_{10} = (?)_{16}$



# Conversão entre Sistemas de Numeração

## Notação Polinomial ou Posicional

- ✓ Válida para qualquer base numérica.
- ✓ LEI DE FORMAÇÃO  
(Notação ou Representação Polinomial):

$$\text{Número} = a_n b^n + a_{n-1} b^{n-1} + a_{n-2} b^{n-2} + \dots + a_0 b^0$$

$a_n$  = algarismo,  $b$  = base do número

$n$  = quantidade de algarismo - 1

# Conversão entre Sistemas de Numeração

## Notação Polinomial ou Posicional

Ex.:

$$a) (1111101)_2 = (?)_{10}$$

$$(1111101)_2 =$$

$$1x2^6 + 1x2^5 + 1x2^4 + 1x2^3 + 1x2^2 + 0x2^1 + 1x2^0 = 125_{10}$$

$$b) (21A)_{16} = (?)_{10}$$

$$(21A)_{16} = 2x16^2 + 1x16^1 + 10x16^0 = 538_{10}$$

# Conversão entre Sistemas de Numeração

## Agrupamento de Bits

- ✓ Sistemas octal e hexa → binário (e vice versa)
- ✓ associando 3 bits ou 4 bits (quando octal ou hexadecimal, respectivamente) e vice-versa.
- ✓ Ex.:  $(1011110010100111)_2 = ( ? )_{16}$        $(A79E)_{16} = ( ? )_2$

|      |      |      |      |
|------|------|------|------|
| 1011 | 1100 | 1010 | 0111 |
| ↓    | ↓    | ↓    | ↓    |
| B    | C    | A    | 7    |

$(1011110010100111)_2 = (BCA7)_{16}$

|      |      |      |      |
|------|------|------|------|
| A    | 7    | 9    | E    |
| ↓    | ↓    | ↓    | ↓    |
| 1010 | 0111 | 1001 | 1110 |

$(A79E)_{16} = (1010011110011110)_2$

# Conversão entre Sistemas de Numeração

Conversão octal  $\longrightarrow$  hexadecimal

- ✓ Não é realizada diretamente - não há relação de potências entre as bases oito e dezesseis.
- ✓ Semelhante à conversão entre duas bases quaisquer - **base intermediária** (base binária)
- ✓ Conversão em duas etapas:
  - 1 - número: base octal (hexadecimal)  $\longrightarrow$  binária.
  - 2 - resultado intermediário: binária  $\longrightarrow$  hexadecimal (octal).

# Conversão entre Sistemas de Numeração

Conversão octal  $\rightarrow$  hexadecimal

Ex.:

a)  $(175)_8 = ( ? )_{16}$

$$(175)_8 = (1111101)_2 = (7D)_{16}$$

b)  $(21A)_{16} = ( ? )_8$

$$(21A)_{16} = (001000011010)_2 = (1032)_8$$

# Conversão entre Sistemas de Numeração

## Conversão de Números Fracionários

Lei de Formação ampliada (polinômio):

$$\text{Número} = \underbrace{a_n \cdot b^n + a_{n-1} \cdot b^{n-1} + a_{n-2} \cdot b^{n-2} + \dots + a_0 \cdot b^0}_{\text{parte inteira}} + \underbrace{a_{-1} \cdot b^{-1} + a_{-2} \cdot b^{-2} + \dots + a_{-m} \cdot b^{-m}}_{\text{parte fracionária}}$$

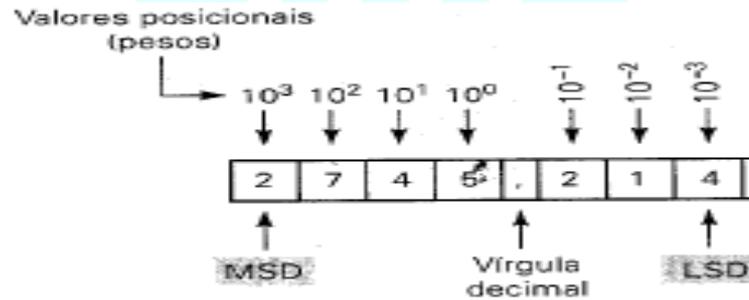
**Exemplo:  $(101,110)_2 = ( ? )_{10}$**

$$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} = (5,75)_{10}$$

# Conversão entre Sistemas de Numeração

## Conversão de Números Fracionários

### Lei de Formação Decimal



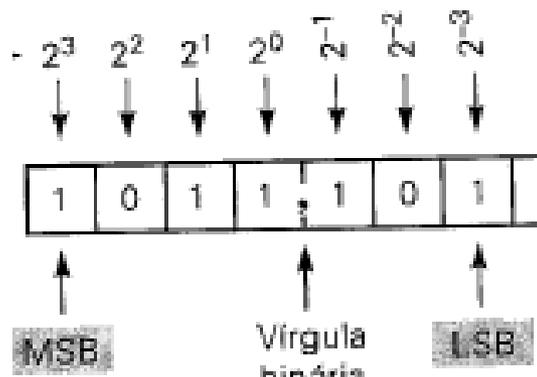
Exemplo:  $(10,214)_{10}$

$$1 \times 10^1 + 0 \times 10^0 + 2 \times 10^{-1} + 1 \times 10^{-2} + 4 \times 10^{-3} = (10,214)_{10}$$

# Conversão entre Sistemas de Numeração

## Conversão de Números Fracionários

### Lei de Formação Binário



Exemplo:  $(1011,101)_2$

$$1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} =$$

$(?)_{10}$

# Conversão entre Sistemas de Numeração

Conversão decimal  $\longrightarrow$  outro sistema

- ✓ Operação inversa: multiplicar a parte fracionária pela base até que a parte fracionária do resultado seja zero.

Exemplo:  $(8,375)_{10} = ( ? )_2$

- parte inteira:  $(8)_{10} = (1000)_2$
- parte fracionária:

$$\begin{array}{r} 0,375 \\ \times 2 \\ \hline 0,750 \\ \downarrow \\ 0 \end{array} \quad \begin{array}{r} 0,750 \\ \times 2 \\ \hline 1,500 \\ \downarrow \\ 1 \end{array} \quad \begin{array}{r} 0,500 \\ \times 2 \\ \hline 1,000 \\ \downarrow \\ 1 \end{array} \quad 0,000 \rightarrow \text{Final}$$

$$(8,375)_{10} = (1000,011)_2$$

# Conversão entre Sistemas de Numeração

✓ Mostre que:

-  $5,8_{10} = 101,11001100\dots_2$  (uma dízima).

-  $11,6_{10} = 1011,10011001100\dots_2$

- a vírgula foi deslocada uma casa para a direita, pois  $11,6 = 2 \times 5,8$ .

# Exercício

- ✓ Uma caixa alienígena com o número 25 gravado na tampa foi entregue a um grupo de cientistas. Ao abrirem a caixa, encontraram 17 objetos. Considerando que o alienígena tem um formato humanóide, quantos dedos ele tem nas duas mãos?

# Exercício

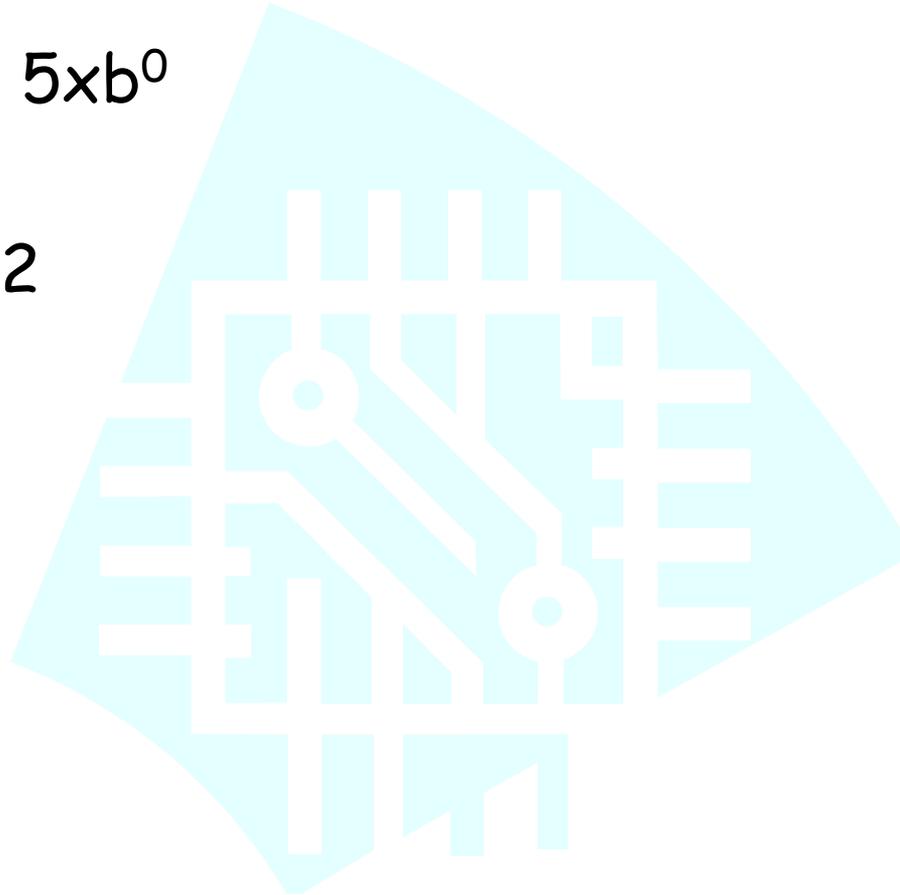
✓  $17_{10} = 25_b$

✓  $17 = 2 \times b^1 + 5 \times b^0$

✓  $17 = 2b + 5$

✓  $b = (17 - 5) / 2$

✓  **$b = 6$**



# Exercício

- ✓ Desenvolva um algoritmo que receba três entradas: um número qualquer, a base do número e a base que será convertido o número. A saída do programa é o número convertido na base escolhida.
  - Pode ser desenvolvido em qualquer linguagem.