Arquitetura de Computadores

SISTEMA DE NUMERAÇÃO

CURSO SUPERIOR DE ANALISE E DESENVOLVIMENTO DE SISTEMAS
FACULDADE DE TECNOLOGIA DE AMERICANA
Prof. Ms Diógenes de Oliveira

Revisão: julho/2021

1 SISTEMAS DE NUMERAÇÃO

TÓPICOS:

- ❑ Conceito de sistema de numeração;
- ❑ Sistema de numeração posicional e não posicional;
- ❑ Sistema decimal;
- ❑ Sistema Binário;
- ❑ Sistema Hexadecimal;
- ❑ Conversão de bases;
- ❑ Aritmética Binária;
- ❑ Aritmética Hexadecimal.

INTRODUÇÂO

Um sistema de numeração define como um número pode ser representado utilizando algarismos distintos.

Desde o início de sua existência, o homem sentiu a necessidade de contar objetos, fazer divisões, diminuir, somar, entre outras operações aritméticas.

Diversas formas de contagem e representação de valores foram criadas e podem ser classificadas em dois grupos: **sistema de numeração posicional** e **não posicional**.

A forma mais amplamente utilizada no mundo para a representação de números é a indo-arábico decimal (nosso sistema decimal)

No sistema de numeração decimal utiliza-se dez algarismos: d={0,1,2,3,4,5,6,7,8,9}

SISTEMA DE NUMERAÇÃO POSICIONAL

Em um sistema de numeração posicional, a posição que o algarismo ocupa no número determina o valor (grandeza) que ele representa.

Neste sistema um número é representado como:

$$\pm\,(d_{n-1}\ \ldots\ d_2\, d_1\, d_0\ ,\ d_{-1}\, d_{-2}\ \ldots\ d_{-k})_b$$

Separador de fração

d = algarismo (símbolo) na posição **i**

n = número de algarismos inteiros (n-i...n-2, n-1, n indicam a posição do algarismo na parte inteira)

k = número de algarismos negativos (-1...-2, -1, -k indicam a posição do algarismo na parte negativa)

b = base (ou raiz) do sistema de numeração.

SISTEMA DE NUMERAÇÃO POSICIONAL

$$\mathbf{N} = \pm \underbrace{d_{n-1} \times b^{n-1} + d_2 \times b^2 + d_1 \times b^1 + d_0 \times b^0}_{\text{Inteiro}} \mathbf{+} \underbrace{d_{-1} \times b^{-1} + \ldots d_{-2} \times b^{-2} + d_{-k} \times b^{-k}}_{\text{fração}}$$

Os termos com potência não negativas de **b** compõem a parte integral do número, enquanto aqueles com potencia negativa de **b** representam a parte fracionária do numero.

Exemplo: $+242,35_{10}$

$$N = 2 \times 10^2 + 4 \times 10^1 + 2 \times 10^0 \mathbf{,}\ 3 \times 10^{-1} + 5 \times 10^{-2}$$

$$N = 2 \times 100 + 4 \times 10 + 2 \times 1 \mathbf{,}\ 3 \times 0{,}1 + 5 \times 0{,}01$$

$$N = 200 + 40 + 2 \mathbf{,}\ 0{,}3 + 0{,}05$$

$$N = 242{,}35$$

Sistema Decimal

O sistema de numeração indo-arábico decimal (em latim *decem*) é composto por **dez algarismos** que ordenados de diferentes maneiras, formam números de qualquer grandeza.

O conjunto de algarismos é **d = 0 1 2 3 4 5 6 7 8 9**

A base do sistema: **b = 10**

Sistema de numeração posicional:
Dependendo da posição que o algarismo ocupa no número determina o valor (grandeza) que ele representa.

Exemplo:
O valor posicional do algarismo **5** nos números **54** e **45** é diferente.

No número **54**, o algarismo **5** representa **50** unidades;
No número **45** o algarismo **5** representa **5** unidades.

Sistema Decimal

Um número inteiro é representado como:

$$N = d_{n-1} \times b^{n-1} + \dots \quad d_4 \times b^4 + d_3 \times b^3 + d_2 \times b^2 + d_1 \times b^1 + d_0 \times b^0$$

n-1	**...**	**4**	**3**	**2**	**1**	**0**	**Posição p**
$\mathbf{b^{k-1}}$	**...**	$\mathbf{b^4}$	$\mathbf{b^3}$	$\mathbf{b^2}$	$\mathbf{b^1}$	$\mathbf{b^0}$	**Valor posicional**
$\mathbf{d_{k-1}}$	**...**	$\mathbf{d_4}$	$\mathbf{d_3}$	$\mathbf{d_2}$	$\mathbf{d_1}$	$\mathbf{d_0}$	**Algarismo d**

Exemplo:

Valores posicionais para o número 224, no sistema decimal

	10^2		10^1		10^0	Valor Posicional
	2		2		4	Algarismos
N =	2×10^2	+	2×10^1	+	4×10^0	Valor

Sistema Binário

O sistema de numeração binário utiliza apenas dois algarismos, 0 e 1 para representar qualquer valor numérico.

d = 0 ou 1
b = 2

$$\pm\,(d_{n-1}\ \ldots\ d_2\, d_1\, d_0\ ,\ d_{-1}\, d_{-2}\ \ldots\ d_{-k})_b$$

Separador de fração

d = algarismo (símbolo) na posição **i**

n = número de algarismos inteiros (n-i...n-2, n-1, n indicam a posição do algarismo na parte inteira)

k = número de algarismos negativos (-1...-2, -1, -k indicam a posição do algarismo na parte negativa)

b = base (ou raiz) do sistema de numeração.

Conversão de Bases Numéricas

Binário para Decimal

$$\sum (b^s * d)$$

$$\sum (2^s * d)$$

5 4 3 2 1 0 ← Posição S

101110_2

b^s d

$2^0 * 0 = 0$

$2^1 * 1 = 2$

$2^2 * 1 = 4$

$2^3 * 1 = 8$

$2^4 * 0 = 0$

$2^5 * 1 = 32$ +

46_{10}

Conversão de Bases Numéricas

Tabela de valores decimais para cada posição do número binário

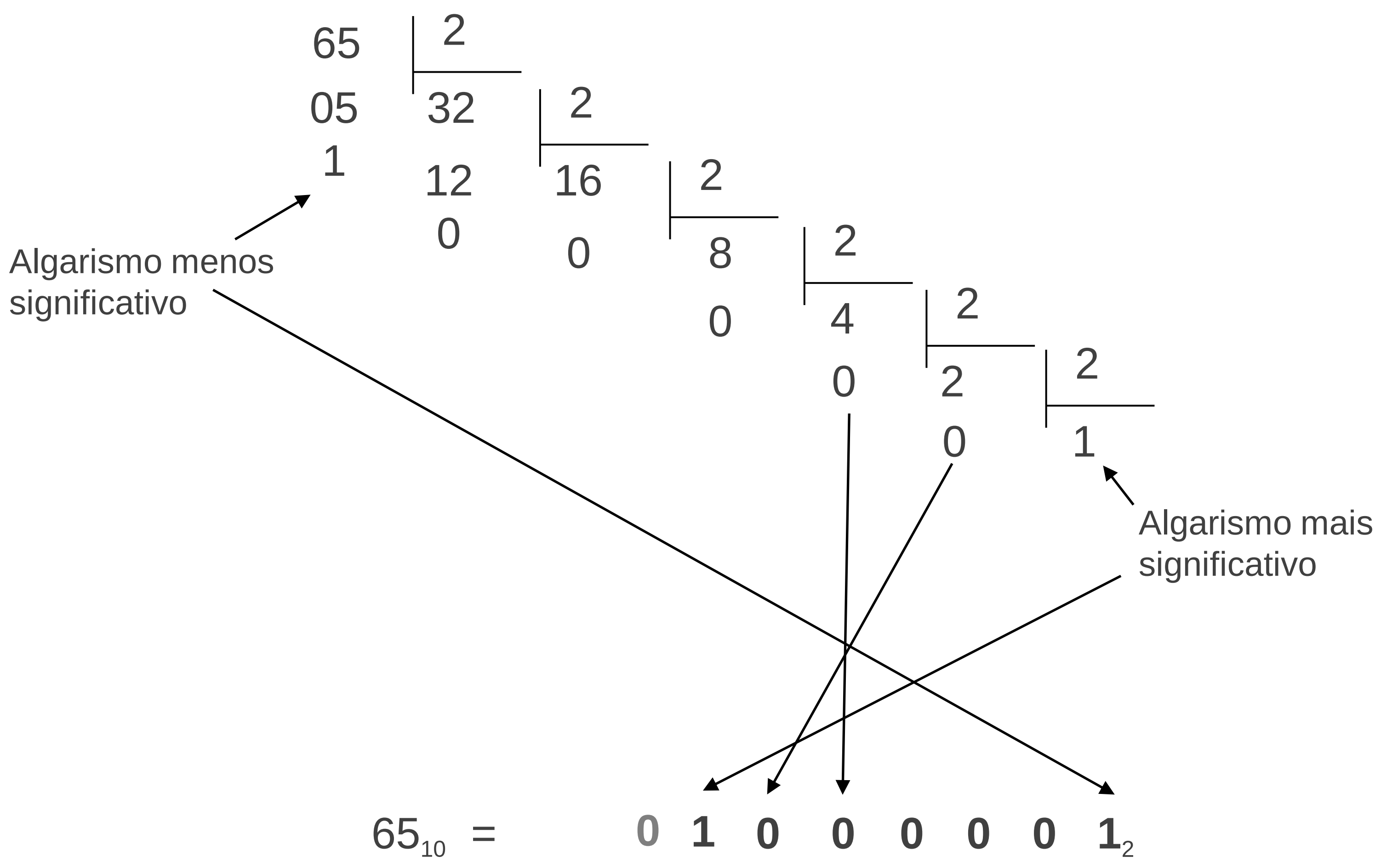
s	**Valor = 2^s**
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1.024
11	2.048
12	4.096
13	8.192
14	16.384
15	32.768
16	65.536

9	8	7	6	5	4	3	2	1	0	← Posição S
0	1	1	0	0	1	1	0	1	0	← Valor binário

Conversão de Bases Numéricas

Decimal para Binário

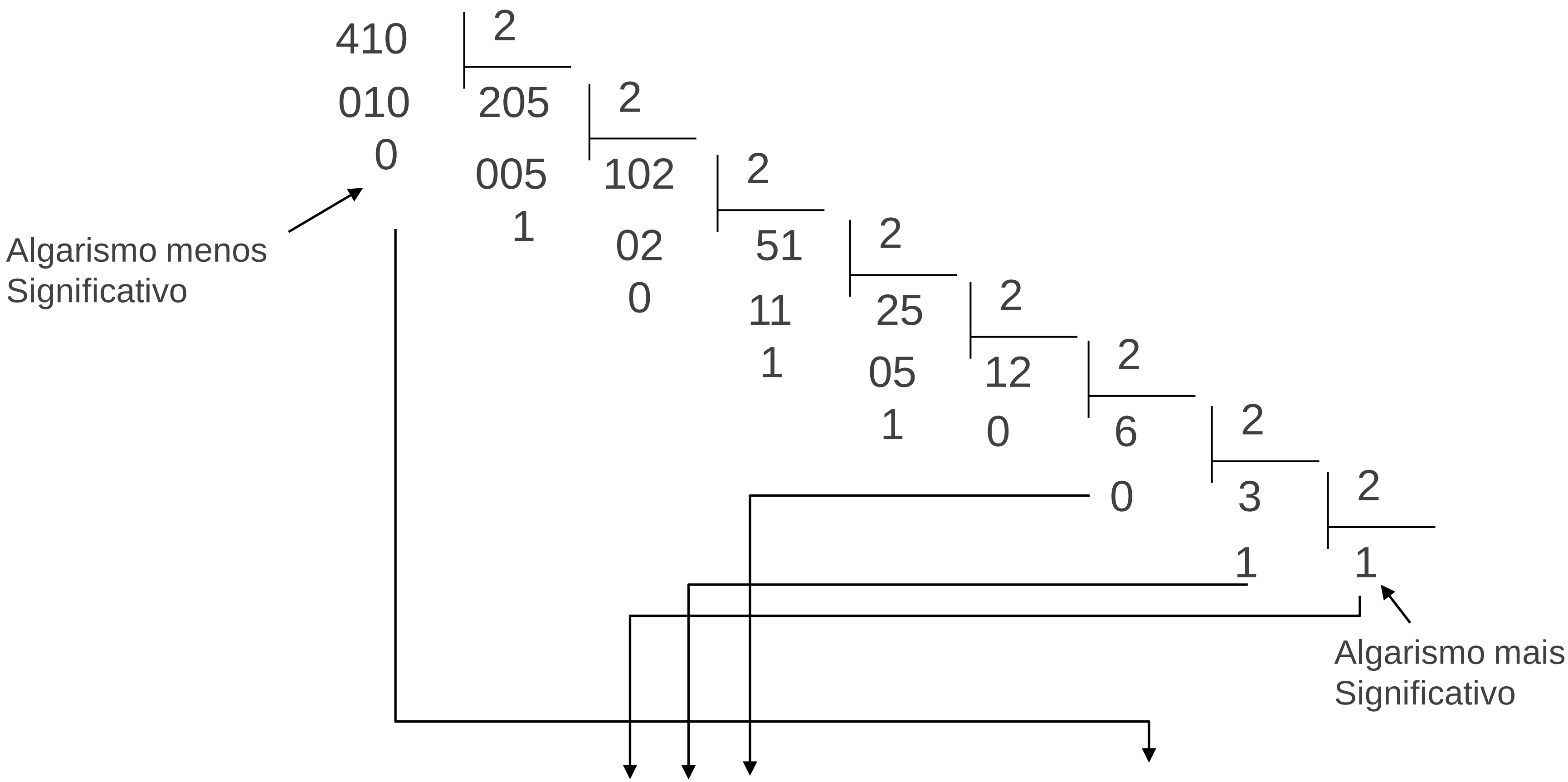
"Divisões consecutivas do valor decimal por dois (2) até não ser mais possível obter um resultado inteiro.



Conversão de Bases Numéricas

Decimal para Binário

Divisões consecutivas do valor decimal por dois (2)



410_{10} = **1 1 0 0 1 1 0 1 0**$_2$

Inverta todos os bits

Exercícios

Faça a conversão dos números escritos na base decimal para números na base binária:

12 = x 1100

45 = x 101101

123 = x 111 1011

Exercícios

Faça a conversão dos números binários para números decimais:

1000001 = x 65

101110 = x 46

11111111 = x 255

Conversão de Bases Numéricas

Binário p/ Decimal

$$\sum (b^s * d)$$

b = base = 2

9	8	7	6	5	4	3	2	1	0	← s = posição
0	1	1	0	0	1	1	0	1	0	← d = valor

$2^0 * 0 = 0$
$2^1 * 1 = 2$
$2^2 * 0 = 0$
$2^3 * 1 = 8$
$2^4 * 1 = 16$
$2^5 * 0 = 0$
$2^6 * 0 = 0$
$2^7 * 1 = 128$
$2^8 * 1 = 256$
$2^9 * 0 = 0$

Valor decimal $= 410_{10}$

Conversão de Bases Numéricas

Binário para Decimal

$$\boxed{\sum (b^s * d)}$$

$$\sum (2^s * d)$$

5 4 3 2 1 0 ← Posição S

$$101110_2$$

b^s d

$$2^0 * 0 = 0$$
$$2^1 * 1 = 2$$
$$2^2 * 1 = 4$$
$$2^3 * 1 = 8$$
$$2^4 * 0 = 0$$
$$2^5 * 1 = 32 \quad +$$

$$46_{10}$$

Aritmética binária

Aritmética binária

Soma

```
   11
  1011
  1001
 ------
 10100
```

$1+1=2 = 10_2$

```
  111
  1111
   111
 ------
 10110
```

$1+1=2 = 10_2$

$1+1+1=3 = 11_2$

Aritmética binária

Soma

```
   1110111        1100110        110111
+    10111      + 1110101      +  10111
  --------       --------       -------
  10001110       11011011       1001110
```

Aritmética binária

Subtração

$$
\begin{array}{cccccccl}
 & & 1 & 2 & & & & \\
 & 0 & \cancel{2} & 0 & 2 & & & \\
 & \cancel{1} & 0 & \cancel{1} & 0 & 1 & \longleftarrow & \text{minuendo} \\
- & & 1 & 1 & 1 & 1 & \longleftarrow & \text{subtraendo} \\
\hline
 & & 0 & 1 & 1 & 0 & &
\end{array}
$$

$$
\begin{array}{rr}
 & 101100 \\
- & 1111 \\
\hline
 & 11101
\end{array}
$$

Aritmética binária

Exercícios: Subtração

$$\begin{array}{r} 1110111 \\ -\quad 10111 \\ \hline 1100000 \end{array} \qquad \begin{array}{r} 1100110 \\ -\quad 110101 \\ \hline 110001 \end{array} \qquad \begin{array}{r} 110111 \\ -\quad 10111 \\ \hline 100000 \end{array}$$

Resolução dos exercícios

TAREFA

110101 + 11111 = x2

1101101 + FABC = x2

FA3 + CBF = x16

11101-111 = x2

FEA - 11001 = x16

EF - 111101 = x10

Meu computador tem 24 bits de Barramento de Endereço, pergunto:

N?

Quantos bits tem a MP?

Exercício: soma binário

$110101 + 11111 = x_2$

$$
\begin{array}{r}
110101 \\
+\ 11111 \\
\hline
1010100_2
\end{array}
$$

Exercício: soma binário e hexadecimal

$1101101 + FABC = x_2$

1. Igualar as bases numéricas antes da operação.

$FABC_{16} = 1111\ 1010\ 1011\ 1100_2$

```
       1 1111 1
          110 1101
1111 1010 1011 1100  +
-------------------
1111 1011 0010 1001
```

Exercício: soma Hexadecimal

$FA3 + CBF = x_{16}$

$$
\begin{array}{r}
111 \\
FA3 \\
+\quad CBF \\
\hline
1C62
\end{array}
$$

15+3=18

18 é maior que a base 16 do sistema. Assim subtrai-se a base do valor, o resultado fica e vai um.

18-16 = 2

1+15+12=28

28 é maior que a base 16 do sistema. Assim subtrai-se a base do valor, o resultado fica e vai um.

28-16 = 12 = C

Subtração: hexadecimal e binário

FEA – 11001 = x_{16}

1. Igualar as bases numéricas antes da operação.

$11001_2 = 19_{16}$

$$\begin{array}{rl} FEA & \\ 19 & - \\ \hline FD1_{16} & \end{array}$$

A = 10
10-9 = 1

Subtração: hexadecimal e binário

$EF - 111101 = x_{10}$

1. Igualar as bases numéricas antes da operação.

$EF_{16} = 1110\ 1111$

$$
\begin{array}{r}
2 \\
002 \\
1\cancel{1}\cancel{1}01111 \\
111101 \\
\hline
10110010
\end{array}
\quad -
$$

$$
\begin{array}{cccccccc}
128 & 64 & 32 & 16 & 8 & 4 & 2 & 1 \\
1 & 0 & 1 & 1 & 0 & 0 & 1 & 0
\end{array}
$$

$$
\begin{array}{r}
128 \\
32 \\
16 \\
2 \\
\hline
178_{10}
\end{array}
\quad +
$$

Subtração em Complemento a Dois

Aritmética binária

Subtração em complemento a dois

Passos:

a. Normalizar os dois valores. preencher o subtraendo com zeros a esquerda até atingir o mesmo número de algarismos do minuendo.

b. Inverter todos os bits do subtraendo.

c. Somar um bit ao valor do subtraendo.

d. Somar os dois valores, minuendo e subtraendo .

e. Desconsiderar o ultimo “vai um”

f. O produto da soma será a subtração entre eles.

$$
\begin{array}{rl}
10101 & \leftarrow \text{minuendo} \\
-\quad 1111 & \leftarrow \text{subtraendo} \\
\hline
\end{array}
$$

a. 01111

b. 10000

c. 10001

$$
\begin{array}{r}
10101 \\
+\ 10001 \\
\hline
110
\end{array}
$$

Aritmética binária

Subtração em complemento a dois

Exercício:

```
  1110111        1100110
-   10111      -  110101
  -------        -------

  1100000         110001
```

Exercício1

Faça a subtração utilizando o método de "complemento a dois" dos valores abaixo:

```
  110111 = 55
-  10111 = 23
-----------
         = 32
```

a) 010111
b) 101000
c) 101001

d)
```
   110111
  +101001
---------------
   100000 = 32
```

e) Não considerar o último vai um;

Exercício2

Faça a subtração utilizando o método de “complemento a dois” dos valores abaixo:

```
  11110111 = 247
 - 1110101 = 117
---------------
           = 130
```

a) 01110101
b) 10001010
c) 10001011

```
d)   11110111
   + 10001011
   ---------------
     10000010 = 130
```

e) Não considerar o último vai um;

Sistema Hexadecimal

SISTEMA HEXADECIMAL

O **sistema hexadecimal** é um sistema de numeração posicional, cuja base numérica é **16** e é representado pelos algarismos: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

É um sistema de numeração muito utilizado para representar números binários de uma forma mais compacta, pois é muito fácil converter binários pra **hexadecimal** e vice-versa.

Ex:

$F_{16} = 1111_2 = 15_{10}$

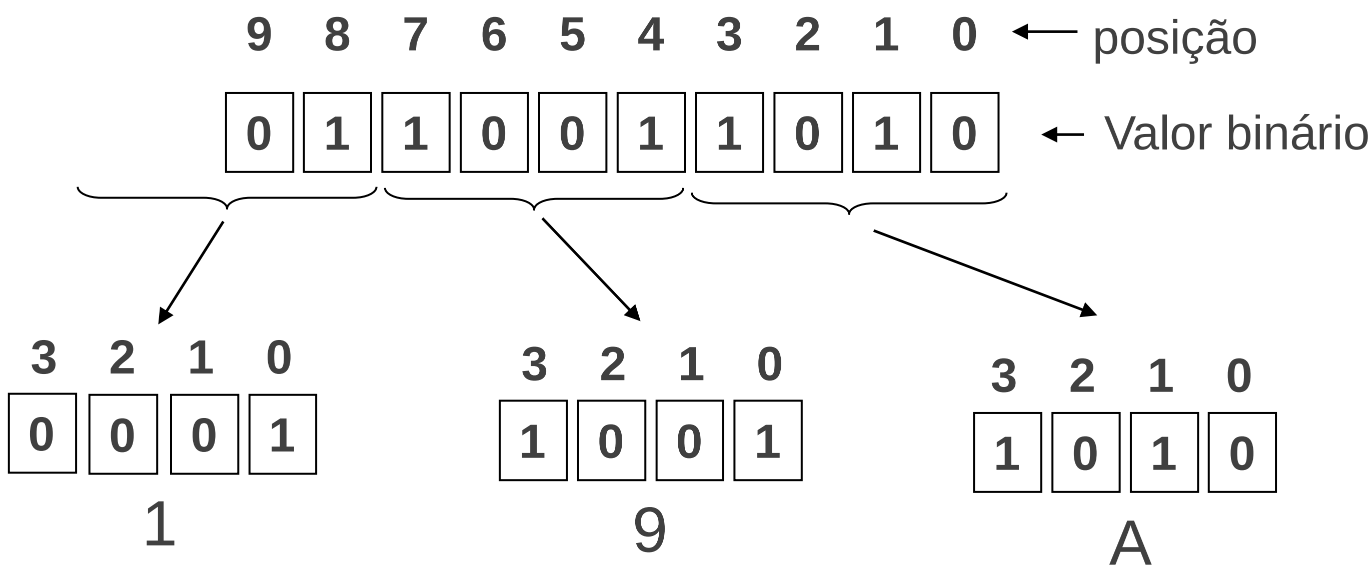
$1_{16} = 0001_2 = 1_{10}$

$1F3_{16} = 0001\ 1111\ 0011_2 = 499_{10}$

Conversão de Bases Numéricas

Binário p/ Hexadecimal

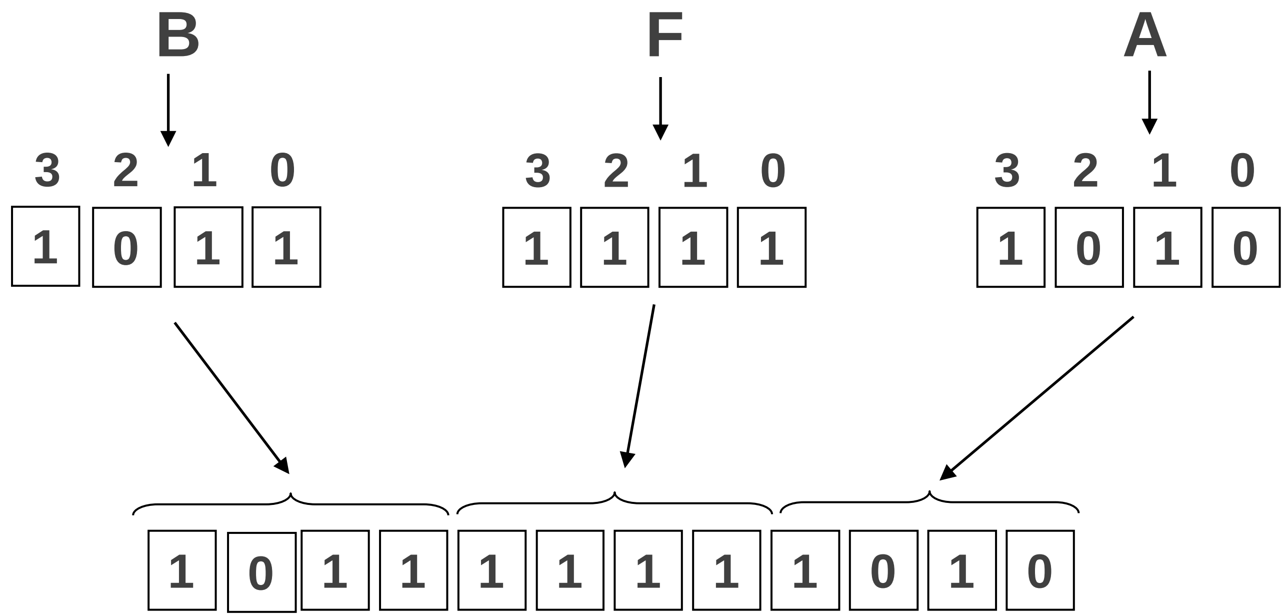
Hexadecimal		Decimal
0	=	0
1	=	1
2	=	2
3	=	3
4	=	4
5	=	5
6	=	6
7	=	7
8	=	8
9	=	9
A	=	10
B	=	11
C	=	12
D	=	13
E	=	14
F	=	15



Conversão de Bases Numéricas

Hexadecimal p/ Binário

Hexadecimal		Decimal
0	=	0
1	=	1
2	=	2
3	=	3
4	=	4
5	=	5
6	=	6
7	=	7
8	=	8
9	=	9
A	=	10
B	=	11
C	=	12
D	=	13
E	=	14
F	=	15



Conversão de Bases Numéricas

Hexadecimal p/ Decimal

$$\sum (b^s * d)$$

$$\sum (16^s * d)$$

3	2	1	0	← posição
1	**5**	**F**	**A**	← valor

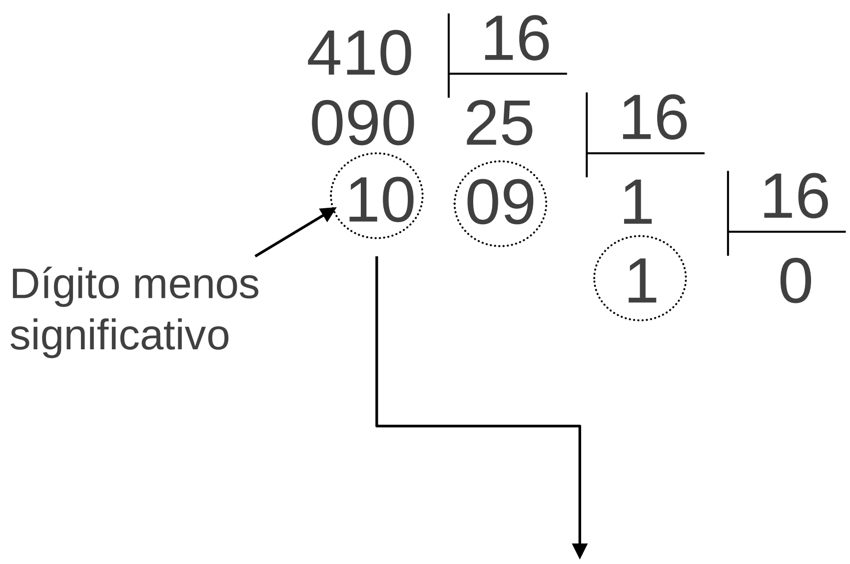
$$16^0 * 10 = 10$$
$$16^1 * 15 = 240$$
$$16^2 * 5 = 1.280$$
$$16^3 * 1 = 4.096$$

Valor Dec. $= 5626_{10}$

Conversão de Bases Numéricas

Decimal p/ Hexadecimal

Divisões consecutivas do valor decimal por dezesseis (16)



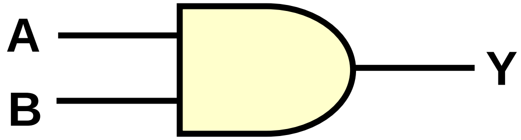
1 9 A$_{16}$

Circuito Lógico
Somador de um bit

PORTAS LÓGICAS

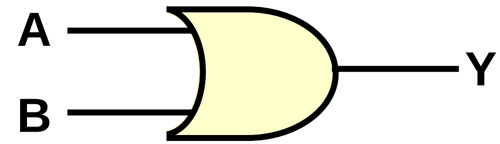
Portas lógicas são dispositivos eletrônicos que operam um ou mais sinais **lógicos** de entrada para produzir um único sinal de saída.

AND



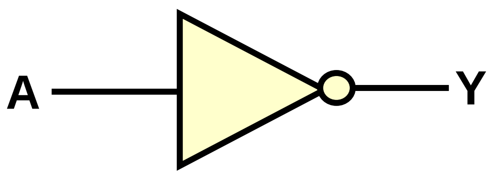
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

OR



A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

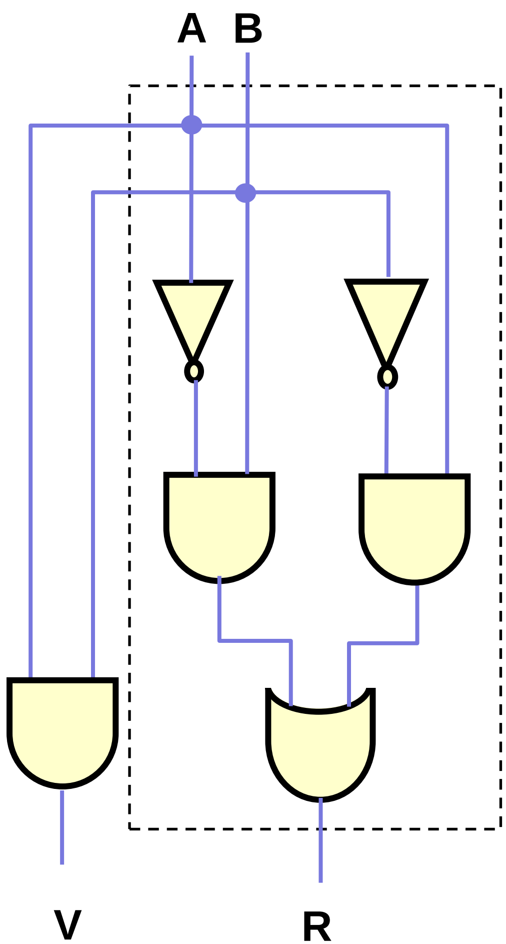
NOT



A	Y
0	1
1	0

CIRCUITOS LÓGICOS

Somador de um bit



A	B	R	V
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1