



Universidad Nacional  
de **Entre Ríos**

---

Facultad de Ciencias de la Administración

Licenciatura en Sistemas

Organización de computadoras

Mg. Marcelo Alberto Colombani  
[marcelo.colombani@uner.edu.ar](mailto:marcelo.colombani@uner.edu.ar)



Facultad de Ciencias de la Administración

Universidad Nacional  
de **Entre Ríos**

---

## UNIDAD TEMÁTICA 2

# SISTEMAS NUMÉRICOS Y REPRESENTACIÓN DE LA INFORMACIÓN

---

# Unidad N° 2

---

## UNIDAD TEMÁTICA 2

### Objetivos:

- Conocer los métodos de representación numérica de los sistemas: decimal, binario, octal y hexadecimal, para números enteros y fraccionarios.
- Discutir los métodos de conversión entre los sistemas numéricos.
- Comprender la necesidad de codificar la información.

# SISTEMAS NUMÉRICOS Y REPRESENTACIÓN DE LA INFORMACIÓN

---

- Sistemas de numeración. Posicional y absoluto.
- Números en punto fijo.
  - Rango y precisión.
  - Sistemas de numeración posicionales.
    - Binario, octal, hexadecimal, decimal, base n.
  - Conversión entre sistemas.
  - Operaciones aritméticas.
  - Números signados.
  - Decimal codificado en binario (BCD).
- Números en punto flotante.
  - Rango y precisión.
  - Errores en la representación en punto flotante.
  - Operaciones aritméticas.
- Codificación.
  - ASCII.
  - EBCDIC.
  - UNICODE.

# BIBLIOGRAFÍA:

---

**Organización de computadoras. Un enfoque estructurado.**

Andrew S.Tanenbaum . Editorial Pearson Educación. Séptima Edición, 2000

**Introducción a la ciencia de la computación.**

Behrouz A. Forouzan . Editorial Thomson 2003

**Organización y arquitectura de computadores**

William Stallings. Editorial Prentice Hall 2000

**Principios de arquitectura de computadoras**

Murdocca, Miles J. Heuring, Vincent P. Editorial Prentice Hall 2002

**Introducción a los computadores**

José Angulo. Editorial Paraninfo 1994

# Repasemos

---

✕ Multiplicación

✕ División

---

# REPRESENTACION DE CANTIDADES

---

- ✗ Los números son símbolos convencionales, que aisladamente considerados no tienen ningún significado, pero lo adquieren cuando se lo relaciona con una cantidad que queremos representar.-
- ✗ Si estuviéramos utilizando los símbolos del sistema decimal, y tenemos \* cantidades a representar, tendríamos:

$$*** = 3$$

$$***** = 8$$

$$***** = 5$$

$$**** = 4$$

---

# SISTEMAS DE NUMERACIÓN

---

- ✖ Un sistema de numeración es un conjunto de símbolos y reglas que permiten representar datos numéricos.



# SISTEMAS DE NUMERACIÓN

---

✗ - Valor absoluto:

ej.: Sistema romano

✗ - Valor posicional o relativo:

ej.: Sistema decimal

# **SISTEMAS DE NUMERACIÓN DE VALOR ABSOLUTO:**

Un sistema de numeración se denomina de valor absoluto cuando los símbolos utilizados para representar cantidades poseen un valor constante independientemente del lugar que ocupa en una cifra.

# SISTEMA DE NUMERACIÓN ROMANO

---

El sistema de numeración romano tiene las siguientes reglas:

- ✗ Los símbolos se agrupan de cinco en cinco y de dos en dos: IIIII=v ; vv= x
- ✗ Los números se leen de izquierda a derecha. Si el primero es mayor o igual que el segundo, los mismos se suman (VI).
- ✗ Si los primeros son menores se restan (IX).

# **SISTEMAS DE NUMERACIÓN DE VALOR RELATIVO O POSICIONAL**

---

- ✖ Los sistemas de numeración actuales son sistemas posicionales, que se caracterizan porque un símbolo tiene distinto valor según la posición que ocupa en la cifra.

# **CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACIÓN DE VALOR RELATIVO O POSICIONAL**

---

- 1. Base**
- 2. Forma de escribir la base**
- 3. Mayor dígito del sistema**
- 4. Principio básico de agrupación**
- 5. Forma de descomponer el número**

# **CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACION DE VALOR RELATIVO O POSICIONAL**

---

**1 - Base:** en un sistema de numeración de valor relativo se denomina base, a la cantidad de símbolos distintos que tiene el sistema.

**Ejemplo:** decimal, 10 números. 0 al 9.

# **CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACIÓN DE VALOR RELATIVO**

---

## **2 - Forma de escribir la base:**

**La base se escribe combinando el segundo símbolo del sistema con el primer símbolo del sistema.**

**Ejemplo:**

**Sistema base 2**

**Símbolos del sistema: 0, 1**

**Se escribe 10**

# **CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACION DE VALOR RELATIVO**

---

## **3 - MAYOR DÍGITO DEL SISTEMA**

**El mayor dígito del sistema que se puede escribir con un solo símbolo, representa una cantidad menor en una unidad a la base:**

**Ejemplo: sistema decimal – base 10**

**Símbolos: 0,1,2,3,4,5,6,7,8,9**

**Mayor dígito del sistema: 9 (una unidad menor a la base)**



# CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACION DE VALOR POSICIONAL

---

## 4 - PRINCIPIO BÁSICO DE AGRUPACIÓN

- **1º) Determinar la base**
- **2º) Elegir los símbolos que representarán a las cantidades**
- **3º) Darles valores a los símbolos y ordenarlos de menor a mayor según su valor**
- **4º) Luego de utilizar todos los símbolos de un solo número combinar el segundo símbolo con cada uno de los símbolos y una vez agotadas las posibilidades combinar el segundo con cada uno y así sucesivamente**

# CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACION DE VALOR RELATIVO

---

## 5 - FORMA DE DESCOMPONER EL NÚMERO:

- En los sistemas de valor posicional o relativo el valor del símbolo es distinto según el lugar que ocupe en la cifra.
- Así en el sistema decimal el número 6 6 6, si bien los tres símbolos se escriben de idéntica forma el primero de la izquierda representa una cantidad de 600, el segundo una cantidad de 60 y el tercero una cantidad de 6.-

# CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACION DE VALOR RELATIVO

---

## 5 - FORMA DE DESCOMPONER EL NÚMERO:

- O sea que el número se puede descomponer así de izquierda a derecha:

$$(6 \times 10^2) + (6 \times 10^1) + (6 \times 10^0) = 666$$

# CARACTERÍSTICAS DE LOS SISTEMAS DE NUMERACIÓN DE VALOR RELATIVO

---

- FORMA DE DESCOMPONER EL NÚMERO:
- Los dígitos a la derecha de la coma de un número decimal fraccionario se representan con potencias negativas de la base 10:

$$(6 \times 10^0) + (6 \times 10^{-1}) + (6 \times 10^{-2})$$

$$(6 \times 1) + (6 \times \frac{1}{10}) + (6 \times \frac{1}{100}) = 6,66$$

# SISTEMA DECIMAL

---

- ✗ Valor relativo
- ✗ Base diez
- ✗ 0,1,2,3,4,5,6,7,8,9
- ✗ La base se representa 10
- ✗ Cada dígito a la izquierda vale diez veces más que el de la derecha.

# GENERACIÓN DE UN SISTEMA DE NUMERACIÓN DE VALOR RELATIVO O POSICIONAL

---

- ✗ Sistema de base cuatro
- ✗ Valor de la base: 4
- ✗ Dígitos: 0, 1, 2, 3
- ✗ Cómo se escribe la base: 10
- ✗ Cada dígito vale 4 veces más que el de la derecha
- ✗ Ej: 0,1,2,3,10,11,12,13,20,21,22,23,30.....

# SISTEMA BINARIO

---

- × Valor relativo
- × Base dos
- × 0,1
- × La base se representa 10
- × Cada dígito vale dos veces más que el de la derecha

# SISTEMA BINARIO

---

- Forma de descomponer un número binario:

Ejemplo: 1 1 0 1

$$(1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = 13$$



# SISTEMA BINARIO

---

- Forma de descomponer un número binario:

Ejemplo: 1 1 1 1 1 1 1 1

$$(1 \times 2^7) + (1 \times 2^6) + (1 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) \Rightarrow$$

$$(1 \times 128) + (1 \times 64) + (1 \times 32) + (1 \times 16) + (1 \times 8) + (1 \times 4) + (1 \times 2) + (1 \times 1) = 255$$

# SISTEMA BINARIO

---

- Forma de descomponer un número binario fraccionario:
- Ejemplo 11,01

$$(1 \times 2^1) + (1 \times 2^0) + (0 \times 2^{-1}) + (1 \times 2^{-2})$$

$$(1 \times 2) + (1 \times 1) + (0 \times \underline{1}) + (1 \times \underline{1})$$

2

4

$$2 + 1 + 0 \times 0,50 + 0,25 = 3,25$$

# SISTEMA BINARIO

---

- ✗ Cada cifra o dígito de un número representado en este sistema se denomina BIT (contracción de binary digit).
- ✗ Para la medida de cantidades de información representadas en binario se utilizan una serie de múltiplos del bit que poseen nombre propio; estos son:
  - ✗ 1 bit = unidad mínima de información.
  - ✗ 8 bits = 1 Byte
  - ✗ 1 byte = 1 letra, número, símbolo de puntuación.
  - ✗ Unidades de medida de almacenamiento
    - ✗ 1,024 bytes = 1 Kilobyte, Kbyte o KB
    - ✗ 1,024 KB = 1 Megabyte, Mbyte o MB (1,048,576 bytes)
    - ✗ 1,024 MB = 1 Gigabyte, Gbyte o GB (1,073,741,824 bytes)
    - ✗ 1,024 GB = 1 Terabyte, Tbyte o TB (1,099,511,627,776 bytes)
    - ✗ 1,024 TB = 1 Pentabyte, Pbyte o PB (1,125,899,906,842,624 bytes)

# SISTEMA OCTAL

---

- ✗ Valor relativo
- ✗ Base 8
- ✗ 0,1,2,3,4,5,6,7
- ✗ La base se representa 10
- ✗ Cada dígito vale 8 veces más que el de la derecha

# SISTEMA HEXADECIMAL

---

- ✗ Valor relativo
- ✗ Base 16
- ✗ 0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f
- ✗ La base se representa 10
- ✗ Cada dígito vale 16 veces más que el de la derecha

# SISTEMAS DE VALOR RELATIVO

DECIMAL	BINARIO	HEXADECIMAL
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

# COMPARACIÓN ENTRE DISTINTOS SISTEMAS

SISTEMAS DE NUMERACIÓN	BASE	CÓMO SE ESCRIBE	CANTIDAD DE DÍGITOS	DÍGITOS	VALOR DE CADA DÍGITO A LA IZQUIERDA
DECIMAL	10	10	10	0,1,2,3,4,5,6,7,8,9	10 VECES
BINARIO	2	10	2	0,1	2 VECES
OCTAL	8	10	8	0,1,2,3,4,5,6,7	8 VECES
HEXADECIMAL	16	10	16	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F	16 VECES

# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

- ✗ Binario a decimal
- ✗ Decimal a binario
- ✗ Hexadecimal a decimal
- ✗ Decimal a hexadecimal
- ✗ Binario a hexadecimal
- ✗ Hexadecimal a binario



# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

---

## ✗ BINARIO A DECIMAL

- ✗ Se descompone el número binario 1 0 1 en las sucesivas potencias de la base:

1                      0                      1

$$(1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = 5$$

# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

---

## ✗ BINARIO A DECIMAL

### ✗ Forma directa:

✗ Se escriben las sucesivas potencias de la base y debajo el número binario 1 0 1:

4	2	1
1	0	1

$$(4 \times 1) + (2 \times 0) + (1 \times 1) = 5$$

# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

## ✗ DECIMAL A BINARIO :

- ✗ Se realizan divisiones sucesivas del número decimal por la base binaria 2 hasta obtener el último cociente entero. El número binario se forma por el último cociente y por todos los restos de las divisiones, del último al primero.

✗  $18 / 2$

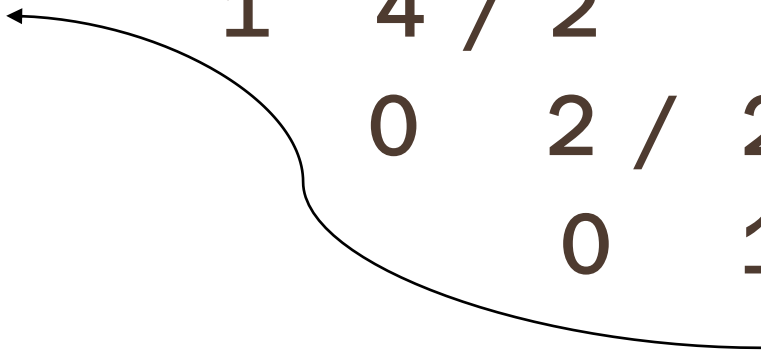
$$18 = 10010$$

$$0 \quad 9 / 2$$

$$1 \quad 4 / 2$$

$$0 \quad 2 / 2$$

$$0 \quad 1$$



# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

---

## ✖ HEXADECIMAL A DECIMAL

✖ Se descompone el número hexadecimal en las sucesivas potencias de la base:

✖           A           4           B  
          10         4         11

$$\begin{array}{ccccccc} (10 \times 16^2) & + & (4 \times 16^1) & + & (11 \times 16^0) & & \\ 2560 & + & 64 & + & 11 & & = 2635 \end{array}$$

# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

---

## × DECIMAL A HEXADECIMAL :

- × Se realizan divisiones sucesivas del número decimal por la base hexadecimal 16 hasta obtener el último cociente entero. El número Hexadecimal se forma por el último cociente y por todos los restos de las divisiones, del último al primero.

×  $66 / 16$

2 4  
↖

$$66_{10} = 42_{16}$$

# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

---

## ✗ BINARIO A HEXADECIMAL :

- ✗ Se separa el número binario en grupos de cuatro dígitos comenzando desde la derecha y se busca el número hexadecimal que le corresponde:

✗      1 1 0 1 1 0 1  
         1 1 0'1 1 0 1  
         0 1 1 0'1 1 0 1  
         6        D

# CONVERSIÓN ENTRE DISTINTOS SISTEMAS

---

## × HEXADECIMAL A BINARIO:

- × Se busca el número binario de 4 dígitos correspondiente a cada dígito hexadecimal:

×	B	E	3
	1 0 1 1	1 1 1 0	0 0 1 1

# OPERACIONES ARITMÉTICAS

---

## SUMA BINARIA

×  $0 + 0 = 0$

×  $0 + 1 = 1$

×  $1 + 0 = 1$

×  $1 + 1 = 0$  llevando 1



# OPERACIONES ARITMÉTICAS

---

## SUMA BINARIA

$$\begin{array}{r} 1\ 0\ 0\ 1\ 0\ 1 \\ +\quad 1\ 0\ 0\ 1\ 0 \\ \hline 1\ 1\ 0\ 1\ 1\ 1 \end{array}$$

$$\begin{array}{r} 1\ 1\ 1\ 1 \\ 1\ 0\ 1\ 1\ 0\ 1 \\ +\quad 1\ 1\ 1\ 0\ 1 \\ \hline 1\ 0\ 0\ 1\ 0\ 1\ 0 \end{array}$$

# OPERACIONES ARITMÉTICAS

---

## RESTA BINARIA

$$\times 1 - 1 = 0$$

$$\times 1 - 0 = 1$$

$$\times 0 - 0 = 0$$

$$\times 0 - 1 = 1 \quad \text{pidiendo 1 al anterior}$$

# OPERACIONES ARITMÉTICAS

## RESTA BINARIA

$$\begin{array}{r} \phantom{-} \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{1} \\ \phantom{-} \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{1} \\ - \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{1} \phantom{0} \\ \hline 1 \phantom{0} 1 \phantom{0} 1 \phantom{0} 1 \phantom{1} \end{array}$$

$$\begin{array}{r} \phantom{-} \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{0} \phantom{0} \phantom{0} \\ \phantom{-} \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{0} \phantom{0} \phantom{0} \\ - \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{0} \phantom{0} \phantom{0} \\ \hline 0 \phantom{1} 1 \phantom{1} 1 \phantom{1} \end{array} \quad \Rightarrow \quad \begin{array}{r} \phantom{-} \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{0} \phantom{0} \phantom{0} \\ \phantom{-} \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{0} \phantom{0} \phantom{0} \\ - \phantom{1} \phantom{1} \phantom{1} \phantom{1} \phantom{0} \phantom{0} \phantom{0} \phantom{0} \\ \hline 0 \phantom{1} 1 \phantom{1} 1 \phantom{1} \end{array}$$

# SUMA Y RESTA HEXADECIMAL

+	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

# OPERACIONES ARITMÉTICAS

---

## SUMA HEXADECIMAL

Intersección entre fila y columna:

$$\begin{array}{r} 213A \\ + 5172 \\ \hline 72AC \\ 1111 \\ 2BCDB \\ + 2572B \\ \hline 51406 \end{array}$$

# SUMA Y RESTA HEXADECIMAL

+	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

# RESTA HEXADECIMAL

---

- ✗ Buscar el sustraendo en el costado de la tabla
- ✗ Seguir por esa fila hasta encontrar el minuendo
- ✗ Ver a qué columna corresponde

$$\begin{array}{r} 9 \ 3 \ B \ D \\ - \ 2 \ 2 \ A \ 4 \\ \hline 7 \ 1 \ 1 \ 9 \end{array}$$

-1   -1   -1

$$\begin{array}{r} 1 \ 2 \ C \ 4 \ A \ 6 \\ - \ \ \ A \ 2 \ 5 \ 7 \ B \\ \hline 0 \ 8 \ 9 \ F \ 2 \ B \end{array}$$

# RESTA HEXADECIMAL

- ✗ Buscar el sustraendo en el costado de la tabla
- ✗ Seguir por esa fila hasta encontrar el minuendo
- ✗ Ver a qué columna corresponde

$$\begin{array}{r} \text{16} \quad \text{16} \quad \text{16} \\ \text{1 2 C}_B \text{ 4 A}_9 \text{ 6} \\ - \text{A 2 5 7 B} \\ \hline \text{0 8 9 F 2 B} \end{array}$$

Cálculos auxiliares

16	16	16
+6	+4	+2
-11	-5	-10
=11=B	=15=F	=8



# SUMA Y RESTA HEXADECIMAL

+	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

# CÓDIGO BCD

---

- ✗ El sistema decimal codificado en binario (bcd) se usa solamente para codificar datos numéricos, representando cada dígito decimal por medio de cuatro dígitos binarios.
- ✗ Mediante este código, la conversión de decimal a bcd y su inversa son simples, dado que cada dígito es codificado individualmente.

# CÓDIGO BCD

---

- ✗ Si se compara el sistema BCD de 4 bits con el sistema binario absoluto se verá que en el BCD se requieren muchos mas símbolos para representar el mismo número.
- ✗ La principal ventaja del sistema bcd es que las cantidades fraccionarias son convertidas en forma precisa, dígito por dígito sin error de conversión.

# CÓDIGO BCD

---

✖ ¿Es posible representar caracteres no numéricos con este código?

**No**, ya que al utilizar solamente cuatro símbolos en el código, habrá solamente dieciséis posibilidades de representación, y diez de ellas son empleadas para los dígitos 0 al 9, y restan solo 6 posibilidades de combinación que no alcanzan para la representación de caracteres alfabéticos y símbolos especiales.

# CÓDIGOS UTILIZADOS

## BCD (BINARY CODED DECIMAL)

---

DECIMAL	8	4	2	1
0	0	0	0	
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Número decimal:      **9          2          8**  
En BCD :            **1 0 0 1   0 0 1 0   1 0 0 0**

# NÚMEROS SIGNADOS

---

- ✗ **Números signados son aquellos que a su izquierda tienen el signo que denota si es positivo o negativo: +12 -12 +7 -9**
- ✗ **Es la representación en MAGNITUD y SIGNO (también conocida como de VALOR ABSOLUTO y SIGNO)**

# NÚMEROS SIGNADOS

---

- ✗ En la representación binaria de magnitud y signo se utiliza el bit ubicado más a la izquierda (el de mayor valor absoluto) para representar el signo, el que tendrá valor 0 si el signo es POSITIVO y valor 1 si el signo es NEGATIVO. Los bits restantes contienen el valor absoluto del número.

Representación de números enteros signados de ocho bits:

Decimal +18 = 00010010

Decimal -18 = 10010010

# NÚMEROS SIGNADOS

---

- ✗ Esta representación tiene dos limitaciones:
  - + Las operaciones aritméticas requieren tener en cuenta tanto el signo como sus magnitudes.
  - + El 0 (cero) tiene dos representaciones:
    - ✗ + 0 = 00000000
    - ✗ - 0 = 10000000



# NÚMEROS DE PUNTO FIJO

---

- ✗ En la representación de números de punto fijo todos los números a representar tienen exactamente la misma cantidad de dígitos.
- ✗ La coma decimal está ubicada en el mismo lugar.
- ✗ Ej: 0,23    5,12    9,11
- ✗ cada número tiene 3 dígitos y la coma decimal se ubica a continuación del primer dígito

# NÚMEROS DE PUNTO FIJO

---

- ✗ En la representación de números de punto fijo en la computadora no se almacena coma decimal alguna, sino que se da por supuesto que ocupa un lugar determinado.

# NÚMEROS DE PUNTO FIJO

---

- ✗ RANGO Y PRECISIÓN :
- ✗ La representación de números de punto fijo se caracteriza por el RANGO DE LOS NÚMEROS QUE EXPRESA (dado por la diferencia entre el número mayor y el menor que expresa).
- ✗ Y por su PRECISIÓN (la distancia entre dos números consecutivos en una serie numérica).

# NÚMEROS DE PUNTO FIJO

---

- × RANGO Y PRECISIÓN :

- × En el ej: 0,23      5,12      9,11

- × (cada número tiene 3 dígitos y la coma decimal se ubica a continuación del primer dígito)

- × el RANGO de representación varía entre 0,00 y 9,99, incluyendo los extremos y se expresa [0,00. . 9,99]

- × Y la PRECISIÓN es 0,01

# NÚMEROS DE PUNTO FIJO

---

- ✗ RANGO Y PRECISIÓN :
- ✗ El ERROR se considera la mitad de la diferencia entre dos números consecutivos.
- ✗ En el ej:  $0,01 / 2 = 0,005$
- ✗ Cualquier número del rango 0,00 a 9,99 se puede representar en este formato con una aproximación de hasta 0,005 de su valor real o preciso.

# NÚMEROS DE PUNTO FIJO

---

- ✗ RANGO Y PRECISIÓN :
- ✗ En la arquitectura de computadoras RANGO y PRECISIÓN son elementos finitos.
- ✗ Y en el mundo real son infinitos.
- ✗ Se requiere una gran cantidad de dígitos para representar un rango amplio de números.

# NÚMEROS DE PUNTO FLOTANTE

---

- ✖ Permite representar un amplio rango de números con poca cantidad de dígitos binarios.
- ✖ RANGO Y PRECISIÓN:
- ✖ Se separan los dígitos para determinar la precisión y los dígitos para representar el rango.
- ✖ El rango se expresa a través de una potencia de la base (EXPONENTE) y la precisión a través del número de punto fijo (MANTISA).

# NÚMEROS DE PUNTO FLOTANTE

## ✖ RANGO Y PRECISIÓN :

Ej: Número en punto fijo: 15.000

Mantisa: 1,5      Exponente: 4

## Notación científica: $1,5 \times 10^4$

[illegible]



# NÚMEROS DE PUNTO FLOTANTE

---

- ✗ ERRORES EN LA REPRESENTACIÓN DE PUNTO FLOTANTE :
- ✗ La representación finita introduce errores.
- ✗ Se debe considerar cuán grande es el error (distancia entre dos números consecutivos) y si es significativo para la aplicación en uso.

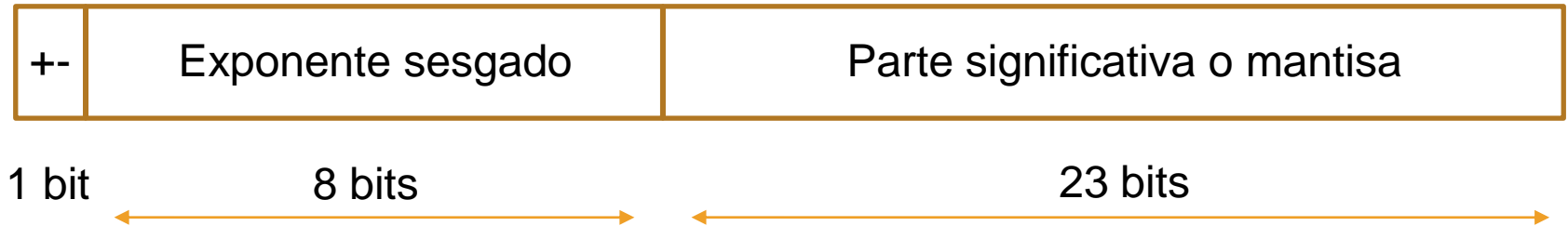
# NÚMEROS DE PUNTO FLOTANTE

---

## ✗ RANGO Y PRECISIÓN :

Casi todo el hardware y lenguajes de programación utilizan números de punto flotante en los mismos formatos binarios, que están definidos en el estándar **IEEE 754** (Instituto de Ingenieros Eléctricos y Electrónicos). Los formatos más comunes son de 32 o 64 bits de longitud total (simple o doble precisión)

# NÚMEROS DE PUNTO FLOTANTE



1 bit de signo

8 bits de exponente sesgada o en exceso 127. Base 2.

23 bits de mantisa

¿Cómo hago la representación sesgada?

Los valores van del 00000000 al 11111111 que representan 0 al 255, pero al estar sesgada va del -127 al 128.

Al valor del exponente se le suma 127 para almacenar el valor.

Para obtener el valor se le resta 127.

# NÚMEROS DE PUNTO FLOTANTE

Los valores van del 00000000 al 11111111 que representan 0 al 255, pero al estar sesgada va del -127 al 128.

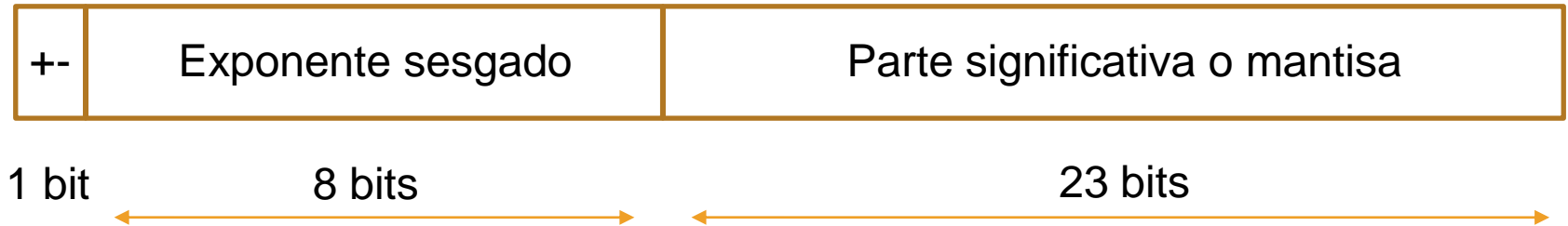
Al valor del exponente se le suma 127 para almacenar el valor.

Para obtener el valor se le resta 127.

Tabla de sesgo:

Número	Sesgado	Binario
-127	0	00000000
-126	1	00000001
-125	2	00000010
-124	3	00000011
0	127	01111111
1	128	10000000
2	129	10000001
3	130	10000010
126	253	11111101
127	254	11111110
128	255	11111111

# NÚMEROS DE PUNTO FLOTANTE



1 bit de signo

8 bits de exponente sesgada o en exceso 127. Base 2.

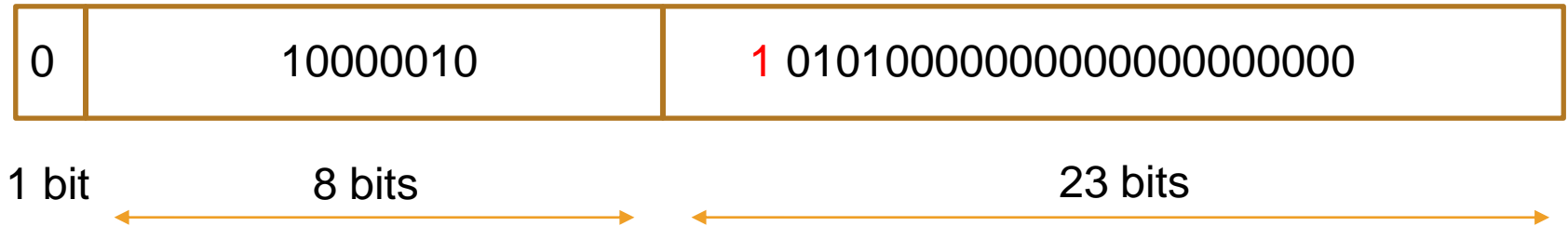
23 bits de mantisa

Vamos a representar el numero +10.5 decimal, en binario 1010.1000

Valor de la mantisa = 10101 suponiendo se escribe 1.0101

Exponente al escribir  $127 + 3 = 130$  entonces 10000010

# NÚMEROS DE PUNTO FLOTANTE



1 bit de signo

8 bits de exponente sesgada o en exceso 127. Base 2.

23 bits de mantisa

Vamos a representar el numero +10.5 decimal, en binario 1010.1000

Valor de la mantisa = 10101 suponiendo se escribe 1.0101

Exponente al escribir  $127 + 3 = 130$  entonces 10000010

# NÚMEROS DE PUNTO FIJO Y FLOTANTE

---

La notación en coma flotante puede ser más lenta de procesar y es menos precisa que la notación en coma fija ya que además de almacenar el número (parte significativa), también debe almacenarse el exponente, pero permite un mayor rango en los números que se pueden representar.

# CÓDIGOS ALFANUMÉRICOS:

---

Tres representaciones habituales en la codificación de caracteres son los códigos alfanuméricos:

- ✗ EBCDIC
- ✗ ASCII
- ✗ Unicode



# CODIGO EBCDIC

---

- ✖ Para solucionar el problema de la falta de posibilidades se expandió el código bcd hasta formar un sistema de 8 símbolos denominado EBCDIC (extended binary coded decimal interchange).
- ✖ Este sistema tiene 4 bits numéricos y 4 bits de zona y de esta manera ahora hay 256 posibilidades de combinación, las que son mas que suficientes para representar todo tipo de datos.

# EBCDIC (EXTENDED BINARY CODED DECIMAL INTERCHANGE CODE)

DECIMAL	C (1100)	D (1101)	E (1110)	F (1111)
0 (0000)				0
1 (0001)	A	J		1
2 (0010)	B	K	S	2
3 (0011)	C	L	T	3
4 (0100)	D	M	U	4
5 (0101)	E	N	V	5
6 (0110)	F	O	W	6
7 (0111)	G	P	X	7
8 (1000)	H	Q	Y	8
9 (1001)	I	R	Z	9

# EBCDIC (EXTENDED BINARY CODED DECIMAL INTERCHANGE CODE)

---

Ej.      3                      4  
         F      3      F      4  
     1111 0011 1111 0100

Ej.      H                      O                      L                      A  
         C      8      D      6                      D      3                      C      1  
     1100 1000 1101 0110 1101 0011 1100 0001

# CODIGO ASCII 8

---

- ✖ Este sistema es conocido también con las siglas uscii, y es utilizado para intercambiar información entre sistemas de procesamiento de datos y sistemas de comunicación.
- ✖ En este sistema cada dígito es representado por una secuencia de 8 bits y puede dársele forma hexadecimal.

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	Ç	160	A0	Á	192	C0	Ł	224	E0	α
129	81	ù	161	A1	í	193	C1	ł	225	E1	β
130	82	é	162	A2	ó	194	C2	ŧ	226	E2	Γ
131	83	â	163	A3	ú	195	C3	ł̈	227	E3	π
132	84	ä	164	A4	ñ	196	C4	—	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	†	229	E5	σ
134	86	ă	166	A6	ª	198	C6	‡	230	E6	μ
135	87	ç	167	A7	º	199	C7	‡	231	E7	τ
136	88	ê	168	A8	¿	200	C8	Ł	232	E8	Φ
137	89	ë	169	A9	ƒ	201	C9	ŕ	233	E9	Θ
138	8A	è	170	AA	¬	202	CA	Ł	234	EA	Ω
139	8B	ÿ	171	AB	½	203	CB	ŕ	235	EB	Θ
140	8C	î	172	AC	¼	204	CC	‡	236	EC	∞
141	8D	ì	173	AD	¡	205	CD	=	237	ED	∞
142	8E	Ä	174	AE	«	206	CE	‡	238	EE	ε
143	8F	Å	175	AF	»	207	CF	±	239	EF	Π
144	90	É	176	B0	⋮	208	DO	Ł	240	FO	≡
145	91	æ	177	B1	⋮	209	D1	ŕ	241	F1	±
146	92	Æ	178	B2	⋮	210	D2	π	242	F2	≥
147	93	ô	179	B3		211	D3	Ł	243	F3	≤
148	94	ö	180	B4	†	212	D4	Ł	244	F4	[
149	95	ò	181	B5	‡	213	D5	ŕ	245	F5	]
150	96	û	182	B6	‡	214	D6	π	246	F6	÷
151	97	ù	183	B7	π	215	D7	‡	247	F7	≈
152	98	ÿ	184	B8	ŕ	216	D8	‡	248	F8	°
153	99	Ö	185	B9	‡	217	D9	ſ	249	F9	•
154	9A	Û	186	BA		218	DA	ŕ	250	FA	·
155	9B	◊	187	BB	ŕ	219	DB	■	251	FB	√
156	9C	£	188	BC	ŕ	220	DC	■	252	FC	∞
157	9D	¥	189	BD	ŕ	221	DD	■	253	FD	≈
158	9E	₠	190	BE	ſ	222	DE	■	254	FE	■
159	9F	f	191	BF	ſ	223	DF	■	255	FF	□

# ASCII (AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE)

	4 (0100)	5 (0101)	3 (0011)
0	-	P	0
1	A	Q	1
2	B	R	2
3	C	S	3
4	D	T	4
5	E	U	5
6	F	V	6
7	G	W	7
8	H	X	8
9	I	Y	9
A	J	Z	
B	K		
C	L		
D	M		
E	N		
F	O		

# ASCII (AMERICAN STANDARD CODE FOR INFORMATION INTERCHANGE)

---

✕ Ej.

A

4

1

0100

0001



4 bits numéricos

4 bits de zona



# UNICODE

---

Unicode es un estándar de codificación de caracteres diseñado para facilitar el tratamiento informático, transmisión y visualización de textos de múltiples lenguajes y disciplinas técnicas, además de textos clásicos de lenguas muertas.

El término Unicode proviene de los tres objetivos perseguidos: **universalidad**, **uniformidad** y **unicidad**.

# UNICODE

---

- A diferencia de los códigos ASCII y EBCDIC que sirven para representar el conjunto de caracteres latinos, el **unicode** es un conjunto de caracteres universal y normalizado que sirve para soportar una buena cantidad de alfabetos que se usan en el mundo.
- Está en desarrollo. Se modifica a medida que se le agregan símbolos de alfabetos nuevos.

# UNICODE

ASCII/8859-1 Text

A	0100 0001
S	0101 0011
C	0100 0011
I	0100 1001
I	0100 1001
/	0010 1111
8	0011 1000
8	0011 1000
5	0011 0101
9	0011 1001
-	0010 1101
l	0011 0002
	0010 0000
t	0111 0100
e	0110 0101
x	0111 1000
t	0111 0100

Unicode Text

A	0000 0000 0100 0001
S	0000 0000 0101 0011
C	0000 0000 0100 0011
I	0000 0000 0100 1001
I	0000 0000 0100 1001
	0000 0000 0010 0000
天	0101 1001 0010 1001
地	0101 0111 0011 0000
	0000 0000 0010 0000
س	0000 0110 0011 0011
س	0000 0110 0100 0100
ا	0000 0110 0011 0111
ا	0000 0110 0100 0101
	0000 0000 0010 0000
α	0000 0011 1011 0001
α	0010 0010 0111 0000
γ	0000 0011 1011 0011