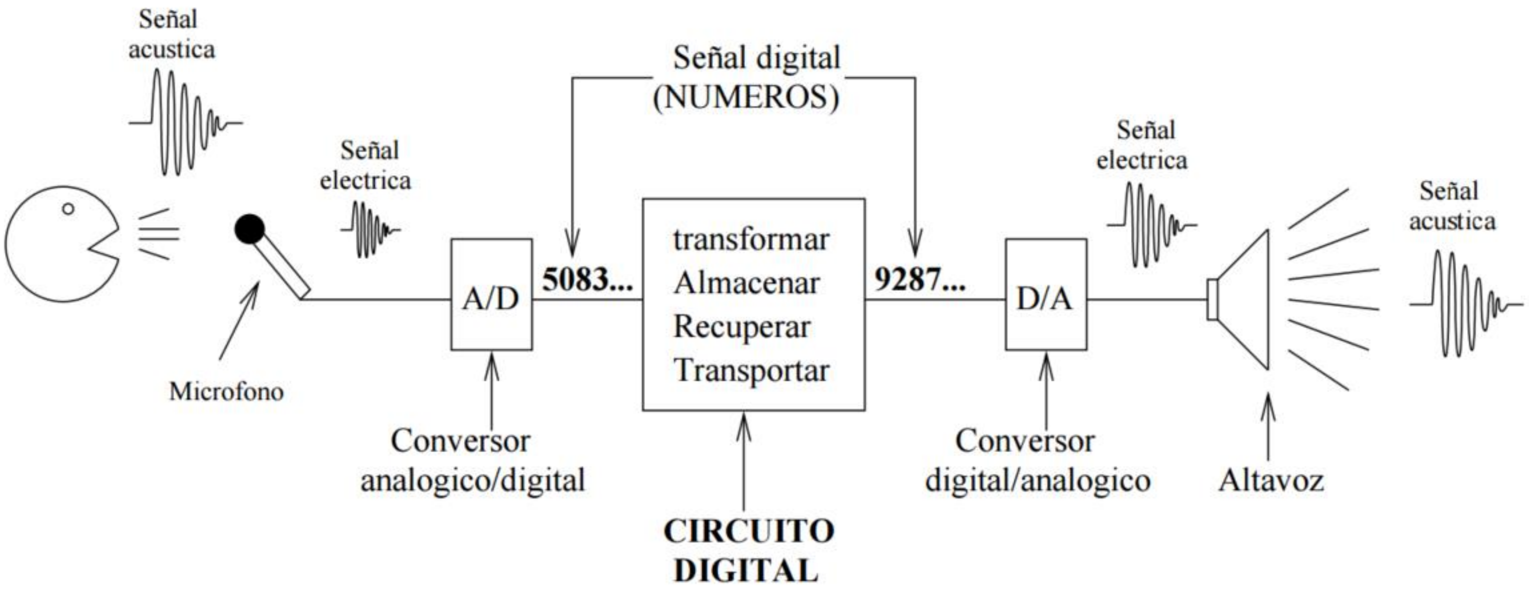
Sistemas Numéricos

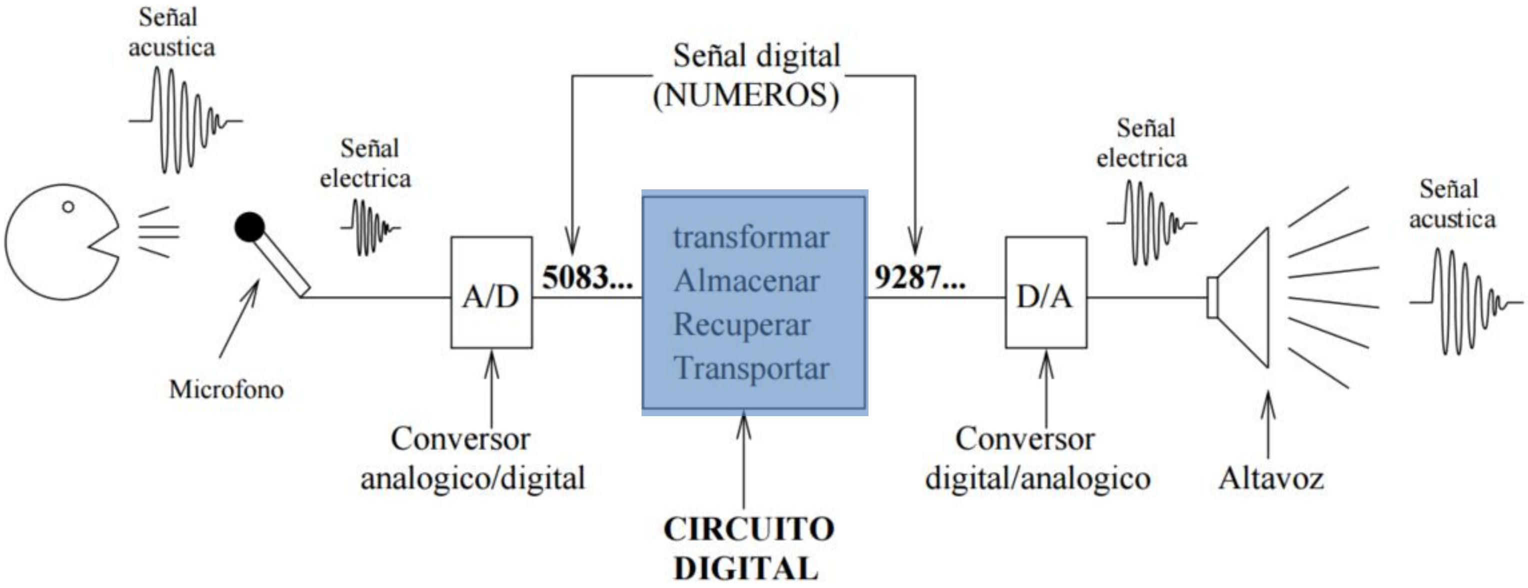
Por:
Carlos A. Fajardo
cafajar@uis.edu.co

¿Por qué estudiar Sistemas Numéricos?

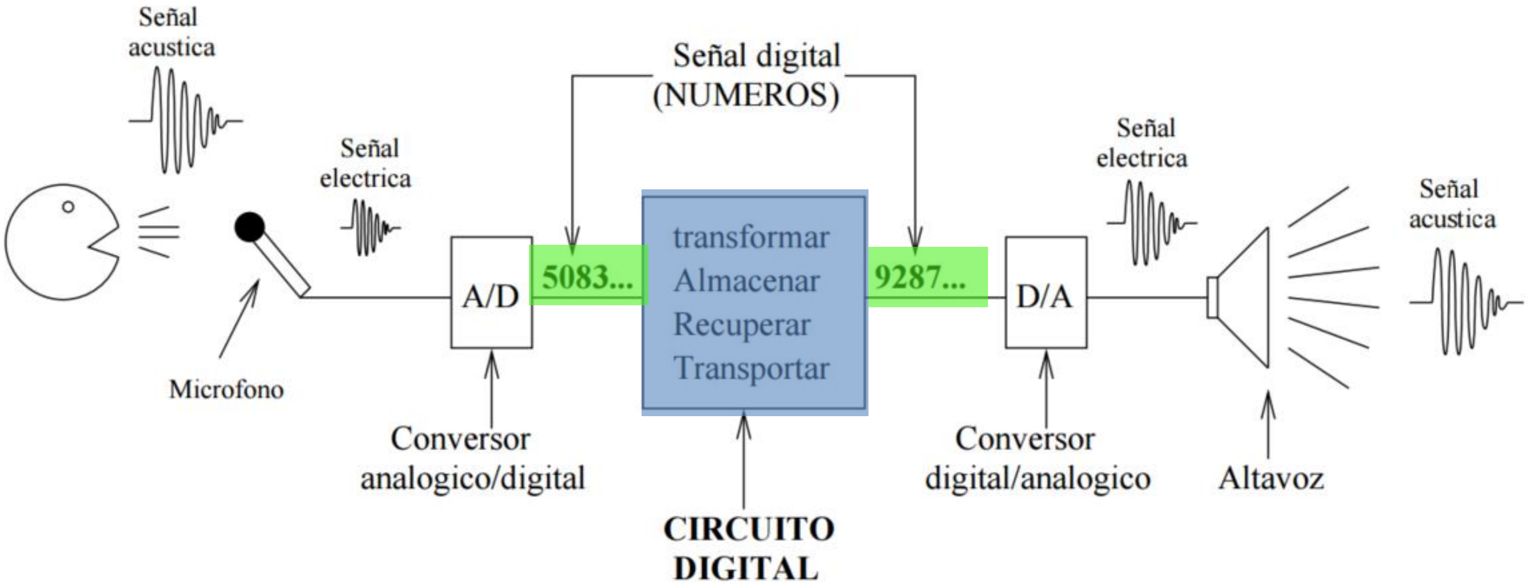
Un Sistema digital



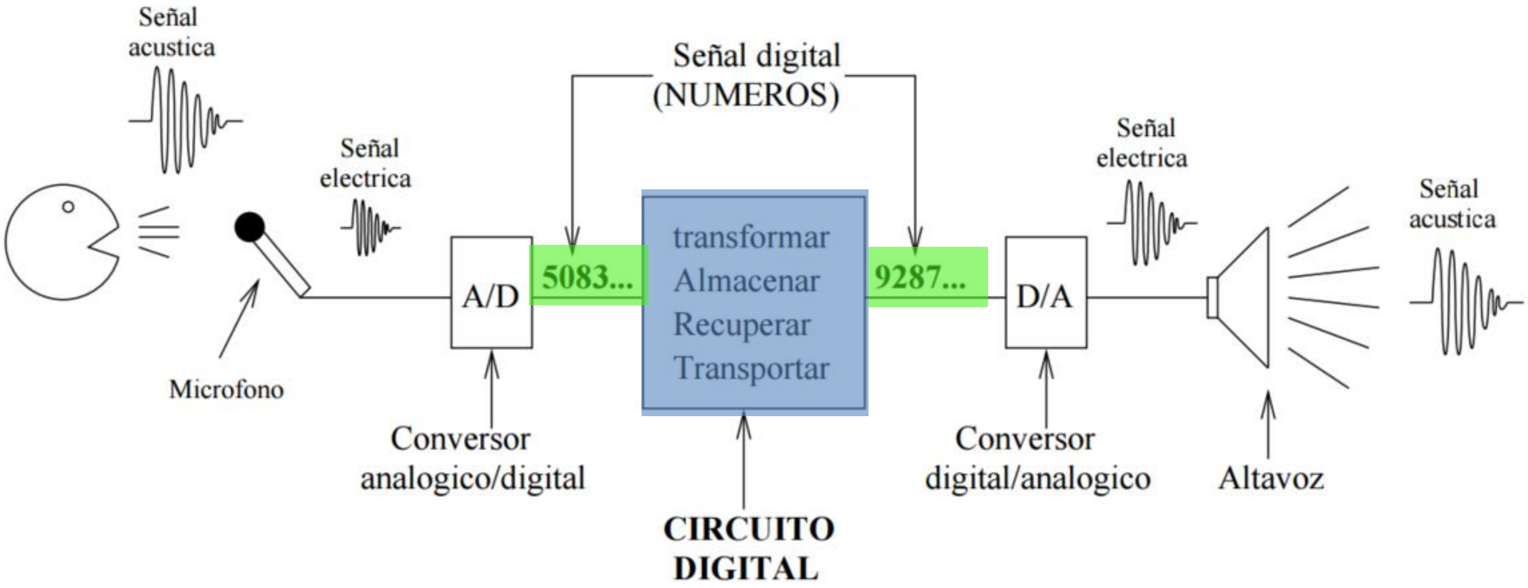
Un sistema digital



Un sistema digital



Un sistema digital



Un sistema digital trabaja sólo con números

Sistemas Numéricos Posicionales

Sistemas numéricos posicionales

4391

Sistemas numéricos posicionales
Base 10

$$4391$$

$$4 * 10^3 + 3 * 10^2 + 9 * 10^1 + 1 * 10^0$$

Sistemas numéricos posicionales
Base 10

$$4391 =$$

$$4 * 10^3 + 3 * 10^2 + 9 * 10^1 + 1 * 10^0$$

Dígitos

Sistemas numéricos posicionales
Base 10

$$4391 =$$

$$4 * 10^3 + 3 * 10^2 + 9 * 10^1 + 1 * 10^0$$

Base y pesos

Sistemas numéricos posicionales
Base 10

$$291,2 =$$

Sistema numéricos posicionales
Base 10

$$291{,}2 =$$

$$2 * 10^2 + 9 * 10^1 + 1 * 10^0 \; + \; 2 * 10^{-1}$$

Sistema numéricos posicionales
Base 10

$$291,2 =$$

$$\boxed{2 * 10^2 + 9 * 10^1 + 1 * 10^0} + \boxed{2 * 10^{-1}}$$

Parte entera

Parte decimal

Sistema numéricos posicionales
Base 7

Se tienen 7 símbolos:

$\{0,1,2,3,4,5,6\}$

$256{,}32_7 =$

Sistema numéricos posicionales
Base 7

Se tienen 7 símbolos:

$\{0,1,2,3,4,5,6\}$

$$256{,}32_7 =$$

$$2 * 7^3 + 5 * 7^2 + 6 * 7^0 + 3 * 7^{-1} + 2 * 7^{-2}$$

Sistema numéricos posicionales
Base 2

Se tienen 2 símbolos:

$$\{0,1\}$$

$$1101_2 =$$

Sistema numéricos posicionales
Base 2

Se tienen 2 símbolos:

$$\{0,1\}$$

$$1101_2 =$$

$$1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0$$

Sistema numéricos posicionales
Base 2 (Binario)

$$1001{,}11_2 =$$

Sistema numéricos posicionales
Base 2 (Binario)

$$1001{,}11_2 =$$

$$1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0 + 1 * 2^{-1} + 1 * 2^{-2}$$

Sistema numéricos posicionales
Base 8(Octal)

Se tienen 8 símbolos:

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

$$2475{,}2_8 =$$

Sistema numéricos posicionales
Base 8(Octal)

Se tienen 8 símbolos:

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

$$2475{,}2_8 =$$

$$2 * 8^3 + 4 * 8^2 + 7 * 8^1 + 5 * 8^0 + 2 * 8^{-1}$$

Sistema numéricos posicionales
Base 16 (Hexa)

Se tienen 16 símbolos:

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

$$A9F7,2_H =$$

Sistema numéricos posicionales
Base 16 (Hexa)

Se tienen 16 símbolos:

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

$$A9F7{,}2_H =$$

$$10 * 16^3 + 9 * 16^2 + 15 * 16^1 + 7 * 16^0 + 2 * 16^{-1}$$

Conversión entre bases

Conversión entre bases

- Binario a decimal: directo
- Decimal a binario
  - Parte entera: divisiones sucesivas
  - Parte decimal: multiplicaciones sucesivas
- Decimal a Hexadecimal: directo
- Decimal a Octal: divisiones sucesivas
- Hexadecimal a Binario: directo
- Octal a Binario: directo

De base "n" a decimal

De base 4 a base 10:

$$231{,}2_4 =$$

De base "n" a decimal

De base 4 a base 10:

$$231{,}2_4 =$$

$$2 * 4^2 + 3 * 4^1 + 1 * 4^0 + 2 * 4^{-1}$$

De base decimal a base "2"

$$22,25_{10}$$

De base decimal a base "2"

Parte entera: divisiones sucesivas

22

De base decimal a base "2"

Parte entera: divisiones sucesivas

$$\begin{array}{r|l} 22 & 2 \\ \hline 0 & 11 \end{array}$$

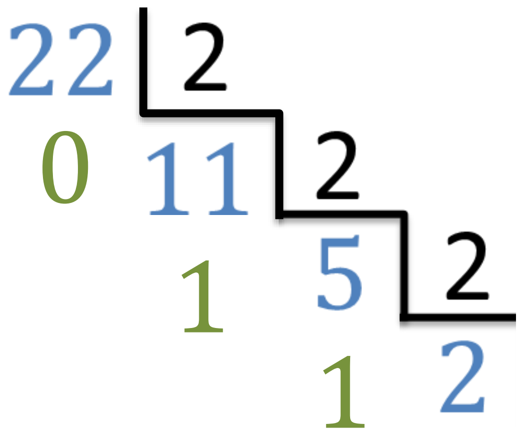
De base decimal a base “2”

Parte entera: divisiones sucesivas

22	2	
0	11	2
	1	5

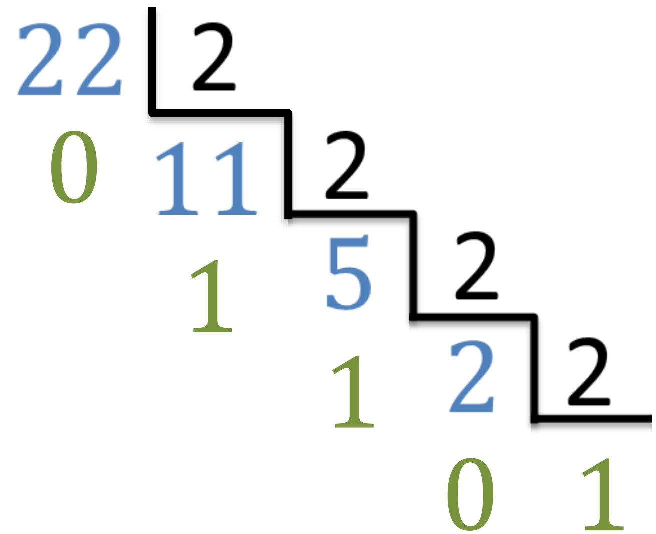
De base decimal a base “2”

Parte entera: divisiones sucesivas



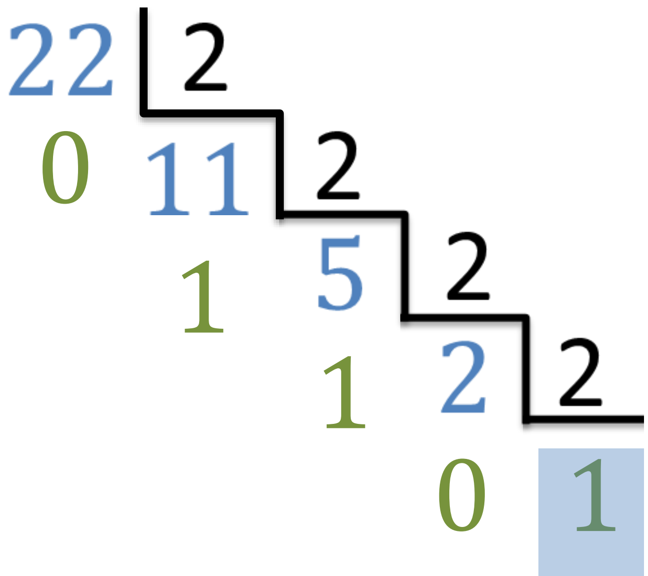
De base decimal a base “2”

Parte entera: divisiones sucesivas



De base decimal a base “2”

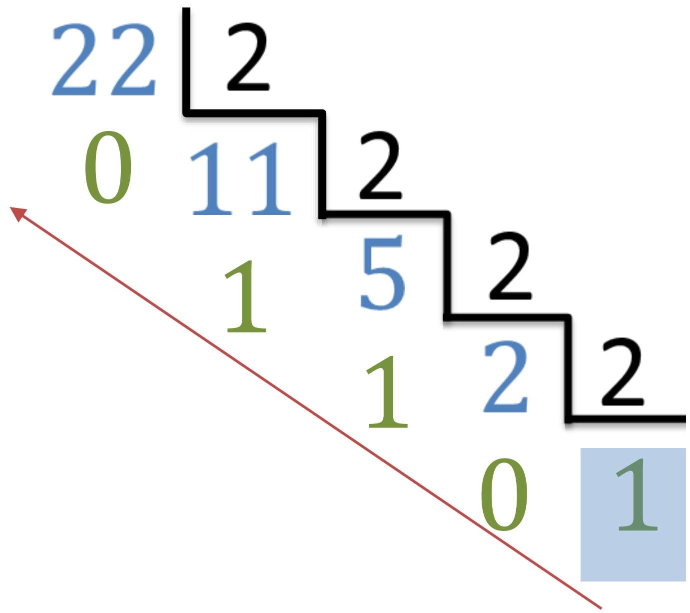
Parte entera: divisiones sucesivas



Menor que la base

De base decimal a base "2"

Parte entera: divisiones sucesivas



Menor que la base

Respuesta: 10110

De base decimal a base “2”

Parte decimal: multiplicaciones sucesivas

$$0{,}25 * 2 = 0\,,\,5$$

$$0{,}5 * 2 = 1\,,\,0$$

De base decimal a base "2"

Parte decimal: multiplicaciones sucesivas

$$0{,}25 * 2 = 0{,}5$$

$$0{,}5 * 2 = 1{,}\boxed{0}$$ **Sea cero**

De base decimal a base “2”

$$22{,}25_{10} =$$

$$\mathbf{1011{,}01}$$

Entre base binaria y bases *de potencia de 2*

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**

Representación de los números en base binaria

Entre base binaria y bases *de potencia de 2*

$$203{,}625_{10} = CB,A_H$$

Entre base binaria y bases *de potencia de 2*

$$203{,}625_{10} = CB,A_H$$

$$= [\mathbf{1100}][\mathbf{1011}],[\mathbf{1010}]$$

Entre base binaria y bases *de potencia de 2*

$$BF09{,}1F_H =$$

Entre base binaria y bases *de potencia de 2*

$$BF09{,}1F_H =$$

$$[\mathbf{1011}][\mathbf{1111}][\mathbf{0000}][\mathbf{1001}],[\mathbf{0001}][\mathbf{1111}]$$

Entre base binaria y bases *de potencia de 2*

$$BF09{,}1F_H =$$

$$[\mathbf{1011}][\mathbf{1111}][\mathbf{0000}][\mathbf{1001}],[\mathbf{0001}][\mathbf{1111}]$$

$$732{,}46_8 =$$

Entre base binaria y bases *de potencia de 2*

$$BF09{,}1F_H =$$

$$[\mathbf{1011}][\mathbf{1111}][\mathbf{0000}][\mathbf{1001}],[\mathbf{0001}][\mathbf{1111}]$$

$$732{,}46_8 =$$

$$[\mathbf{111}][\mathbf{011}][\mathbf{010}],[\mathbf{100}][\mathbf{110}]$$

Entre base binaria y bases *de potencia de 2*

$$BF09{,}1F_H =$$

$$[1011][1111][0000][1001],[0001][1111]$$

$$732{,}46_8 =$$

$$[111][011][010],[100][110]$$

$$321{,}12_4 =$$

Entre base binaria y bases *de potencia de 2*

$$BF09{,}1F_H =$$

$$[1011][1111][0000][1001],[0001][1111]$$

$$732{,}46_8 =$$

$$[111][011][010],[100][110]$$

$$321{,}12_4 = [11][10][01],[01][10]$$

¿Cómo se representan los números negativos en binario?

Representación de Números Negativos

- No existe el signo –, sólo tenemos bits (dos valores de voltaje)
- Se debe buscar la manera de representar los números negativos usando bits (1 y 0)

Representación de Números Negativos

Dos formas principales:

- **Magnitud y signo**
- **Complemento a la base**

Magnitud y Signo

- Se utiliza el primer bit como signo:
  - Cero para los positivos
  - Uno para lo negativos

- Ejemplo con cuatro bits:
  - 0111 = +7
  - 1111 = -7
  - 0010 = +2
  - 1010 = -2

Magnitud y Signo

- Rango:

Para 3 bits: 111 (-3) hasta 011 (+3)

Para 4 bits: 1111 (-7) hasta 0111 (+7)

Para 5 bits: 11111 (-15) hasta 01111 (+15)

- En general para n bits:

$$-(2^{n-1}) \text{ hasta } (2^{n-1})$$

Magnitud y Signo

Desventajas

- Es más complejo operar aritméticamente.
  - Para sumar: Primero hay que determinar si los dos números tienen el mismo signo o si tienen signo diferentes.
  - El cero posee doble representación.

Complemento a la base

Ventajas

- Facilita las operaciones matemáticas
- El cero tiene una única representación.

Complemento a la base

- Los **números positivos** se representan de la **misma forma** que en Signo y Magnitud.

- Los **negativos** se representan de la siguiente manera:

$$\boldsymbol{N_c = b^n - N}$$

Donde=

N_c: El número en complemento a la base

b: la base

n: numero de dígitos de la representación

N: El número positivo

Complemento a la base

- Los **negativos** se representan de la siguiente manera:

$$N_c = b^n - N$$

Ejemplos:

Representar en complemento a 2, con 4 bits, el número -7:

$$N_c = 2^4 - 7 = 9$$

1001

Complemento a la base

- Otra forma de representarlo:

$$N_c = b^n - N$$

$$N_c = (b^n - 1) - N + 1$$

$$N_c = [(b^n - 1) - N] + 1$$

Representar en complemento a 2, con 4 bits, el número -7:

$$\begin{array}{r} 1111 \\ -0111 \\ \hline 1000 \\ +1 \\ \hline 1001 \end{array}$$

Se complementa

Complemento a la base

En resumen:

- Se niega el número y se le suma 1:

Ejemplos:

0011 $\rightarrow$ 1100 $\rightarrow$ **1101**

0101 $\rightarrow$ 1010 $\rightarrow$ **1011**

Complemento a la base

Nemotecnia :

- Se transcribe igual el número.
- De derecha a izquierda hasta que se encuentra el primer 1.
- Manteniendo el 1 intacto, se complementan los restantes dígitos que haya a su izquierda.

Ejemplo:

Complemento a 2 de $00000100_2 \Rightarrow 11111100_{2c}$

Representación de número reales en los sistemas de cómputo

Representación de números reales

- Punto Fijo
- Coma flotante

Punto fijo

- Se define el número de bits de la parte entera y el número de bits de la parte decimal.

Ejemplo:

[4:2] Cuatro bits parte entera y dos bits parte decimal.

110011 equivale a 1100,11

Coma flotante

- Muchas aplicaciones requieren trabajar con números muy grandes.

- También se requiere trabajar con números muy pequeños (decimales)

- Una alternativa para representarlos que es comúnmente usada en los computadores es la que se conoce como representación en **punto flotante** .

Coma flotante Precisión simple

El formato para los números de precisión simple en estándar IEEE 754 es de 32 bits:

Nombre	***Signo***	***Exponente***	***Mantisa***
Posición [bits]	31	30 23	22 0

– **Signo**(0:positivo y 1:negativo). Se encuentra en el bit más significativo

– **Exponente en exceso.** Está conformado por los siguientes 8 (n) bits.

$$\mathbf{Exp + \ 127}$$

– **Mantisa.** Está formada por 23 bits .

Coma flotante Precisión doble

El formato para los números de precisión doble en estándar IEEE 754 es de 64 bits:

Nombre	***Signo***	***Exponente***	***Mantisa***
Posición [bits]	63	62 52	51 0

- **Signo**(0:positivo y 1:negativo). Se encuentra en el bit más significativo
- **Exponente en exceso.** Está conformado por los siguientes 11 bits.

$$\mathbf{Exp + 1023}$$

- **Mantisa.** Está formada por 52 bits .

Ejemplo 1

Convertir el siguiente numero a binario en formato de coma flotante precisión simple en estándar IEEE 754.

$$\mathbf{45,25_{10}}$$

Ejemplo 1

- *Se pasa el número a binario*

$$\mathbf{45,25_{10} = 101101,01_2}$$

- *Se normaliza: un solo bit en la parte entera*

$$\mathbf{101101,01_2 = 1,0110101_2 * 2^5}$$

Ejemplo 1

- *Se halla el exponente y la mantiza*

Exponente

$$101101,01_2 = 1,\boxed{0110101}_2 * 2^5$$

Mantisa

- *Se halla el exponente en exceso EE:*

$$\mathrm{EE} = \mathbf{127} + \boldsymbol{exponente}$$

$$\boldsymbol{E} = \mathbf{127} + \mathbf{5} = \mathbf{132} = \mathbf{10000100}_{\ \mathbf{2}}$$

Ejemplo 1

- *Se halla el exponente*

$$E = exponente\ en\ exceso + exponente\ externo$$

$$E = 127 + 5 = 132 = 10000100\ _2$$

- *Se normaliza: un solo bit en la parte entera*

$$101101,01_2 = 1,0110101_2 * 2^5$$

Mantisa

Exponente

Ejemplo 1

- *Se halla la Mantisa*: se omite el bit que antecede al mantisa(luego de normalizado el numero binario: $\mathbf{1,0110101_2 * 2^5}$) y se agregan 0 asta alcanzar la cantidad necesaria de bits.

$$\mathbf{01101010000000000000000}$$

Ejemplo 1

- *Se halla la Mantisa*: se omite el bit que antecede al mantisa(luego de normalizado el numero binario: $\mathbf{1,0110101_2 * 2^5}$) y se agregan 0 asta alcanzar la cantidad necesaria de bits.

$$\mathbf{01101010000000000000000}$$

- signo:

$$\mathbf{0}$$

Ejemplo 1: Solución

$$45,25_{10} =$$

Signo	exponente	mantissa
0	10000100	$01101010000000000000000_2$

Ejemplo 2

Convertir el siguiente numero a binario en formato de coma flotante precisión simple en estándar IEEE 754.

$$-\mathbf{0,005_{10}}$$

Ejemplo 2

Convertir el siguiente numero a binario en formato de coma flotante precisión simple en estándar IEEE 754.

$$-0,005_{10}$$

- *En binario:*

$$-0,005_{10} = -0,0000000101_2$$

Ejemplo 2

Convertir el siguiente numero a binario en formato de coma flotante precisión simple en estándar IEEE 754.

$$-0,005_{10}$$

- *En binario:*

$$-0,005_{10} = -0,0000000101_2$$

- *Normalizando*:

$$-0,0000000101_2 = -1,01_2 * 2^{-8}$$

Ejemplo 2

- *Exponente* :

$$\boldsymbol{E = 127 - 8 = 119 = 01110111_2}$$

Ejemplo 2

- *Exponente* :

$$E = 127 - 8 = 119 = 01110111_2$$

- *Mantisa*:

$$01000000000000000000000$$

Ejemplo 2

- *Exponente* :

$$\boldsymbol{E = 127 - 8 = 119 = 01110111_2}$$

- *Mantisa*:

$$\mathbf{01000000000000000000000}$$

- *signo*:

$$\mathbf{1}$$

Ejemplo 2: Solución

$$-0{,}005_{10} =$$

Ejemplo 2: Solución

$$-0,005_{10} =$$

$$10111011101000000000000000000000_{2}$$

Infinito en coma flotante

- El exponente máximo es +127 que en exceso es:

11111110

- Para más o menos infinito el exponente es:

11111111

Nombre	*Signo*	*Exponente*	*Mantisa*
+∞	**0**	**11111111**	**0**
-∞	**1**	**11111111**	**0**

¿Dónde puedo aprender más?

- Floyd, T. L. (2006). *Fundamentos de sistemas digitales* (Ed.9). Prentice Hall.
  - Secciones 2,1 - 2,8.
- González, Juan (2002). Circuitos y Sistema Digitales. Universidad Pontificia de Salamanca.
  - Secciones 2,1 – 2,7.
- Google.