Tema 1: Sistemas de representación

Contenido

1. **Sistemas numéricos**
   - Sistemas de numeración y cambio de base
   - Aritmética binaria
   - Sistemas de codificación y representación de los números
2. **Codificación binaria**
   - Representación binaria de datos e instrucciones
   - Características de los espacios de representación

1. Sistemas numéricos

Sistemas de numeración y cambio de base

- Un sistema de numeración en *base* ***b*** utiliza para representar los números un alfabeto compuesto por **b** símbolos o cifras
- Ejemplos:

  b = 10 (*decimal*) {0,1,2,3,4,5,6,7,8,9}

  b = 16 (*hexadecimal*) {0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F}

  b = 2 (*binario*) {0,1}
- El número se expresa mediante una secuencia de cifras:

  $N \equiv \ldots\ n_4\ n_3\ n_2\ n_1\ n_0\ n_{-1}\ n_{-2}\ n_{-3}\ \ldots$
- El valor de cada cifra depende de la cifra en sí y de la posición que ocupa en la secuencia

1. Sistemas numéricos

Sistemas de numeración y cambio de base

- El valor del número se calcula mediante el polinomio:

$$N \equiv \ldots + n_3 \cdot b^3 + n_2 \cdot b^2 + n_1 \cdot b^1 + n_0 \cdot b^0 + n_{-1} \cdot b^{-1} + \ldots$$

$$N \equiv \sum_i n_i \cdot b^i$$

- Ejemplos:

$$3278{,}52_{10} = 3 \cdot 10^3 + 2 \cdot 10^2 + 7 \cdot 10^1 + 8 \cdot 10^0 + 5 \cdot 10^{-1} + 2 \cdot 10^{-2}$$

$$175{,}372_8 = 1 \cdot 8^2 + 7 \cdot 8^1 + 5 \cdot 8^0 + 3 \cdot 8^{-1} + 7 \cdot 8^{-2} + 2 \cdot 8^{-3} = 125{,}4882812_{10}$$

Sistemas de numeración y cambio de base

- Conversión decimal - base b
  - Método de divisiones sucesivas entre la base **b**
  - Para números fraccionarios se realizan multiplicaciones sucesivas por la base **b.**
  - Consideración de restos mayores que 9 y *Error de truncamiento*
- Ejemplos:

$26 \,|\, 2$

$0 \quad 13 \,|\, 2$

$1 \quad 6 \,|\, 2$

$0 \quad 3 \,|\, 2$

$1 \quad 1$

$26_{10} = 11010_2$

$0{,}1875_{10} = 0{,}0011_2$

$$0{,}1875 \times 2 = 0{,}3750 \qquad 0{,}3750 \times 2 = 0{,}7500 \qquad 0{,}7500 \times 2 = 1{,}5000 \qquad 0{,}5000 \times 2 = 1{,}0000$$

$26{,}1875_{10} = 11010{,}0011_2$

1. Sistemas numéricos

Sistemas de numeración y cambio de base

- **Rango de representación**: Conjunto de valores representable. Con **n** cifras en la base **b** podemos formar b^n combinaciones distintas. $[0..b^n-1]$

- **Sistema de numeración en base dos o binario**

b = 2 (*binario*)

{0,1}

Números binarios del 0 al 7

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

- Ejemplos:

$110100_2 = (1\cdot 2^5) + (1\cdot 2^4) + (1 \cdot 2^2) =$

$= 2^5 + 2^4 + 2^2 = 32 + 16 + 4 = 52_{10}$

$0{,}10100_2 = 2^{-1} + 2^{-3} = (1/2) + (1/8) = 0{,}625_{10}$

$10100{,}001_2 = 2^4 + 2^2 + 2^{-3} = 16 + 4 +(1/8)$

$= 20{,}125_{10}$

Sistemas de codificación y representación de números

- **Octal**

  b = 8 (*octal*) {0,1,2,3,4,5,6,7}

  - Correspondencia con el binario

    $\mathbf{8 = 2^3} \Rightarrow$ Una cifra en octal corresponde a 3 binarias

- Ejemplos

  $10\,001\,101\,100.110\,10_2 = 2154.64_8$

  $537.24_8 = 101\,011\,111.010\,100_2$

- Conversión Decimal - Octal

  $760.33_{10} \cong 1370.2507_8$

760 ÷ 8 = 95, resto 0
95 ÷ 8 = 11, resto 7
11 ÷ 8 = 1, resto 3
1

0.33 x 8 = 2.64

0.64 x 8 = 5.12

0.12 x 8 = 0.96

0.96 x 8 = 7.68

1. Sistemas numéricos

Sistemas de representación y codificación de números

- **Hexadecimal**

  b = 16 (*hexadecimal*)

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

  - Correspondencia con el binario

    $16 = 2^4$ $\Rightarrow$ Una cifra en hexadecimal corresponde a 4 binarias

Hexadecimal	Decimal	Binario
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

1. Sistemas numéricos

Sistemas de representación y codificación de números

- Ejemplos

$$\overbrace{10}\overbrace{0101}\overbrace{1101}\overbrace{1111}.\overbrace{1011}\overbrace{101}_2 = \text{25DF.BA}_H$$

- Conversión Decimal - Hexadecimal

$$4373.79_{10} \cong \text{1115.CA3D}_{16}$$

$$\begin{array}{r|l} 4373 & 16 \\ \hline 117 & 273 \\ 53 & \\ 5 & \end{array} \qquad \begin{array}{r|l} 273 & 16 \\ \hline 113 & 17 \\ 1 & \end{array} \qquad \begin{array}{r|l} 17 & 16 \\ \hline 1 & 1 \end{array}$$

$$\begin{array}{r} 0.79 \\ \times\ 16 \\ \hline 12.64 \end{array} \qquad \begin{array}{r} 0.64 \\ \times\ 16 \\ \hline 10.24 \end{array} \qquad \begin{array}{r} 0.24 \\ \times\ 16 \\ \hline 3.84 \end{array} \qquad \begin{array}{r} 0.84 \\ \times\ 16 \\ \hline 13.44 \end{array}$$

1. Sistemas numéricos

Aritmética binaria

- Operaciones básicas

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	0 (1)

A	B	A*B
0	0	0
0	1	0
1	0	0
1	1	1

A	B	A – B
0	0	0
0	1	1 (1)
1	0	1
1	1	0

A	B	A/B
0	0	--
0	1	0
1	0	--
1	1	1

1. Sistemas numéricos

Aritmética binaria

- Ejemplos
  - Sumas y restas

```
    1110101         1101010
+   1110110      -  1010111
-----------     -----------
   11101011         0010011
```

  - Multiplicaciones

```
    1101010           1010011
       x 11              x 10
-----------       -----------
    1101010           0000000
   1101010           1010011
-----------       -----------
  100111110          10100110
```

  - División

```
1101.011 | 101
 -101    |--------
-----      10.101
 00110
   -101
  -----
   00111
     -101
    -----
       10
```

1. Sistemas numéricos

Sistemas de representación y codificación de números

- **Código Gray**
  - Código no ponderado, continuo y cíclico
  - Basado en un sistema binario
  - Dos números sucesivos sólo varían en un bit

2 bits	3 bits	4 bits	Decimal
0 0	0 0 0	0 0 0 0	0
0 1	0 0 1	0 0 0 1	1
1 1	0 1 1	0 0 1 1	2
1 0	0 1 0	0 0 1 0	3
	1 1 0	0 1 1 0	4
	1 1 1	0 1 1 1	5
	1 0 1	0 1 0 1	6
	1 0 0	0 1 0 0	7
		1 1 0 0	8
		1 1 0 1	9
		1 1 1 1	10
		1 1 1 0	11
		1 0 1 0	12
		1 0 1 1	13
		1 0 0 1	14
		1 0 0 0	15

1. Sistemas numéricos

Sistemas de representación y codificación de números

- **Código BCD** - *Binary Coded Decimal*
- Dígitos decimales codificados en binario

Decimal	BCD natural	BCD exceso 3	BCD Aiken	BCD 5421
0	0 0 0 0	0 0 1 1	0 0 0 0	0 0 0 0
1	0 0 0 1	0 1 0 0	0 0 0 1	0 0 0 1
2	0 0 1 0	0 1 0 1	0 0 1 0	0 0 1 0
3	0 0 1 1	0 1 1 0	0 0 1 1	0 0 1 1
4	0 1 0 0	0 1 1 1	0 1 0 0	0 1 0 0
5	0 1 0 1	1 0 0 0	1 0 1 1	1 0 0 0
6	0 1 1 0	1 0 0 1	1 1 0 0	1 0 0 1
7	0 1 1 1	1 0 1 0	1 1 0 1	1 0 1 0
8	1 0 0 0	1 0 1 1	1 1 1 0	1 0 1 1
9	1 0 0 1	1 1 0 0	1 1 1 1	1 1 0 0

- BCD natural tiene pesos 8421
- BCD Aiken tiene pesos 2421
- Ejemplo

$$9\ 8\ 3\ 2\ 5_{10} = 1001\ 1000\ 0011\ 0010\ 0101_{\text{BCD-natural}}$$

$$9\ 8\ 3\ 2\ 5_{10} = 1111\ 1110\ 0011\ 0010\ 1011_{\text{BCD-Aiken}}$$

1. Sistemas numéricos

- **Representación de números enteros**
  - Es necesario la representación del signo
  - Se utiliza una cantidad determinada de bits (n)
- **Signo y magnitud (SM)**
  - El signo se representa en el bit más a la izquierda del dato. Bit (n-1)
  - En el resto de los bits se representa el valor del número en binario natural. Bits (n-2)..0
  - Doble representación del 0.

n = 6

$10_{10} = 001010_{SM}$ $-4_{10} = 100100_{SM}$

$0_{10} = 000000_{SM}$ $0_{10} = 100000_{SM}$

n = 4

$-7_{10} = 1111_{SM}$ $-14_{10} =$ no representable

1. Sistemas numéricos

- **Complemento a la base menos uno**
  - Los valores positivos se representan en SM.
  - Los valores negativos se obtienen restando la magnitud del número a la base elevada al número de dígitos, menos uno.
  - Convierte las restas en sumas.
  - Doble representación del 0.
  - Ejemplos **Base 10**

n = 3 $-63_{10} = 936_{C9} \Rightarrow 936 = 999 - 63$

$-16_{10} = 983_{C9} \Rightarrow 983 = 999 - 16$

n = 4 $-16_{10} = 9983_{C9} \Rightarrow 9983 = 9999 - 16$

Operación: 77 - 63

$$\begin{array}{r} 77 \\ -63 \\ \hline 14 \end{array} \qquad \begin{array}{lr} & 077_{C9} \\ + & 936_{C9} \\ \hline & (1)013 \\ + & 1 \\ \hline & 014_{C9} \end{array}$$

Base 2

- Se intercambian ceros por unos y unos por ceros
- Rango : $[-2^{n-1} + 1, 2^{n-1} - 1]$
- Ejemplos:

n = 6 C_1 de $-10010_2 = 101101_{C1}$

$$\begin{array}{r} 111111 \\ -\ 010010 \\ \hline 101101 \end{array}$$

C_1 de -100111_2 = no representable

C_1 de $0 = \{000000_{C1}, 111111_{C1}\}$

Operación: $1000111_2 - 10010_2$

Restando en binario natural

$$\begin{array}{r} 1000111_2 \\ -\ 0010010_2 \\ \hline 0110101_2 \end{array}$$

Sumando en C1 (n=8)

$$\begin{array}{r} 01000111_{C1} \\ +\ 11101101_{C1} \\ \hline (1)00110100 \\ +\ 1 \\ \hline 00110101_{C1} \end{array}$$

1. Sistemas numéricos

Sistemas de representación y codificación de números

- **Complemento a la base**
  - Los valores positivos se representan en SM.
  - Los valores negativos se obtienen sumado uno a la representación en complemento a la base menos uno
  - Convierte las restas en sumas.
  - Ejemplos **Base 10**

n = 3 $-63_{10} = 937_{C10} \Rightarrow 937 = (999 - 63) + 1$

$-16_{10} = 984_{C10} \Rightarrow 984 = (999 - 16) + 1$

n = 4 $-16_{10} = 9984_{C10} \Rightarrow 9984 = (9999 - 16) + 1$

Operación: 77 - 63

$$\begin{array}{rr} & 077 \\ + & 937 \\ \hline & (1)014 \end{array}$$

El acarreo, si existe, no se considera

Sistemas de representación y codificación de números

Base 2

- Se intercambian los ceros y los unos y se suma ‘1’
- Rango : $[-2^{n-1}, 2^{n-1} - 1]$
- Ejemplos:

$n = 6 \quad C_2$ de $-10010_2 = 101110_{C2}$

$$\begin{array}{r} 111111 \\ -\ 010010 \\ \hline 101101_{C1} \end{array} \qquad \begin{array}{r} 101101 \\ +\qquad 1 \\ \hline 101110_{C2} \end{array}$$

C_2 de -1110010_2 = no representable

Operación: $11001_2 - 10010_2 = 111_2$

Operando en C2

(n=6)

$$\begin{array}{r} 011001_{C2} \\ +\ 101110_{C2} \\ \hline (1)000111_{C2} \end{array}$$

El acarreo no se considera

1. Sistemas numéricos

Sistemas de representación y codificación de números

- **Representación sesgada**
  - La representación se obtiene sumando un sesgo o cantidad al valor del número
  - El sesgo suele ser: 2^{n-1}
  - Rango : $[-2^{n-1}, 2^{n-1} - 1]$
  - Ejemplos **Base 2**

$n = 8 \Rightarrow \text{Sesgo} = 2^{8-1} = 128_{10} = 1000\ 0000_2$

$11010_2 = 10011010_S$

$-11010_2 = 01100110_S$

$0_2 = 1000\ 0000_S$

$n = 4 \Rightarrow \text{Sesgo} = 2^{4-1} = 8_{10} = 1000_2$

$1_2 = 1001_S$

$-1_2 = 0111$

1. Sistemas numéricos

Sistemas de representación y codificación de números

- Representación de los números reales
  - Representación en coma fija
  - Representación en coma flotante

$$N = (s)\, M \cdot B^{E}$$

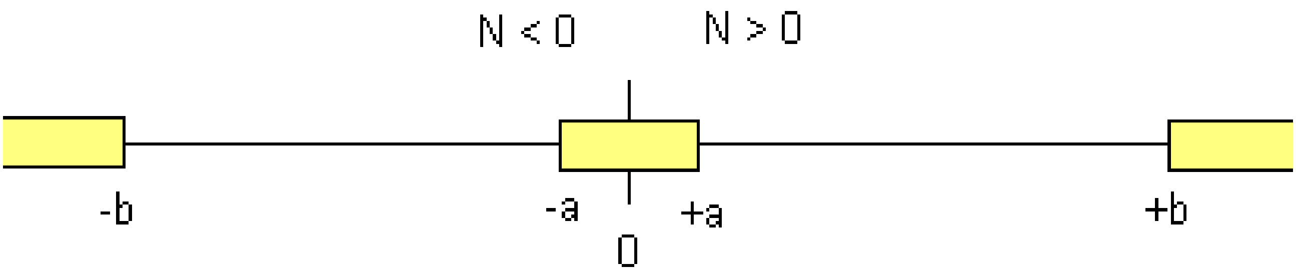
N ≡ Valor numérico **M** ≡ Mantisa **s** ≡ signo

B ≡ Base **E** ≡ Exponente

- Ejemplo en base 10:

$1.234535 \cdot 10^{3} = 1234.535 \cdot 10^{0} = 0.1234535 \cdot 10^{4}$
$= 123453.5 \cdot 10^{-2} = 0.0001234535 \cdot 10^{7}$

- Valores límite



- **Estándar IEEE754**

$$N = (-1)^s M \cdot 2^E$$

Representación **S E M**

$$N = nS + nE + Nm$$

Siendo *nS* la cantidad de bits para el signo, *nE* la cantidad de bits para el exponente y *nM* la cantidad de bits para la mantisa.

En ***simple precisión c***onsta de 32 bits, distribuidos de la siguiente forma:

- Campo de signo : 1 bit

$$0 \Rightarrow +$$

$$1 \Rightarrow -$$

- Campo del exponente: 8 bits con representación sesgada, siendo el sesgo:

$$S = 2^{ne-1}-1$$

1. Sistemas numéricos

Sistemas de representación y codificación de números

Ejemplos:

ne = 8 $\Rightarrow$ S = 2^{ne-1}-1 = 127 = 0111 1111

(E)	E+S	(e)
0	127+0=127	0111 1111
+2	127+2=129	1000 0001
+127	127+127=254	1111 1110
-1	127-1=126	0111 1110
-126	127-126=1	0000 0001

- Campo de mantisa : 23 bits, con formato normalizado:
  - $1 \leq M < 2$
  - La mantisa siempre tendrá la forma M = [1.*m*] donde *m* es el valor que se almacena en el formato.
  - Ejemplos de normalización:

N1 = $1001.1100110 \cdot 2^{-5}$ = $1.\mathbf{0011100110} \cdot 2^{-2}$

N2 = $0.000001101101 \cdot 2^{34}$ = $1.\mathbf{101101} \cdot 2^{28}$

Ejercicio:

Supongamos un formato de las mismas características que el IEEE754, pero dotado solo de 16 bits, de los cuales 1 es para el signo y 8 para el exponente. Identifiquemos el número:

N = 1001 1111 0001 1101

$$s = 1 \Rightarrow N<0 \quad e = 0011\ 1110 \Rightarrow E = -65$$

$$m = 001\ 1101 \Rightarrow M = 1.0011101_2 = -1.2265625_{10}$$

$$N = -1.2265625 \cdot 2^{-65} = -3{,}32440346980633 \cdot 10^{-20}$$

1. Sistemas numéricos

Sistemas de representación y codificación de números

➢ Situaciones especiales:

Si el en el campo del exponente encontramos e = 0, la mantisa está **desnormalizada**.

El sesgo pasa a ser $2^{ne-1}-2$. Y el exponente del número será: $E = e - S = -2^{ne-1} + 2$

- Si $(e = 0) \wedge (m = 0) \rightarrow \quad N = 0$
- Si $(e = 11...1) \wedge (m = 0) \rightarrow N = \infty$
- Si $(e = 11...1) \wedge (m \neq 0) \rightarrow N = NaN$

➢ Redondeo:

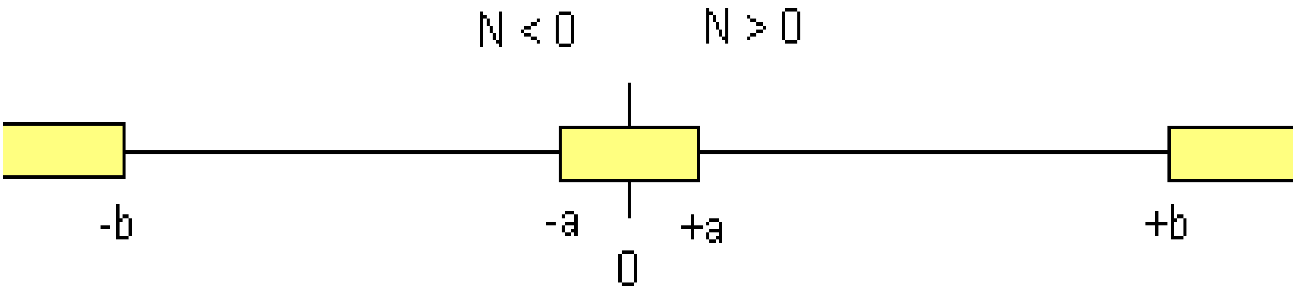
El redondeo de un número lo podemos realizar por exceso, por defecto, al más cercano o al par. El formato IEEE754 emplea el redondeo al par.

Resultado de la ALU (normalizado)	Acción	Mantisa redondeada
1.01101|10	Sumar 1	1.01110
1.01100|10	Truncar	1.01100
1.01100|11	Sumar 1	1.01101
1.01100|01	Truncar	1.01100
1.01100|00	Truncar	1.01100

1. Sistemas numéricos

Sistemas de representación y codificación de números

- Rango de Representación



- Números normalizados

$$|b| = M_{max}2^{Emax} \qquad M_{max} = 2 - 2^{-nm}$$

$$E_{max} = 2^{ne-1}-1$$

$$|a| = M_{min}2^{Emin} \qquad M_{min} = 1$$

$$E_{min} = -(2^{ne-1}-2)$$

- Números desnormalizados

$$|a'| = M'_{min}2^{E'min} \qquad M'_{min} = 2^{-nm}$$

$$E'_{min} = -(2^{ne-1}-2)$$

1. Sistemas numéricos

Sistemas de representación y codificación de números

- Valores límite
  - Si $|N| > |b| \Rightarrow$ desbordamiento a infinito
    OVERFLOW
  - Si $|N| < |a'| \Rightarrow$ desbordamiento a cero
    UNDERFLOW

- Precisión:
  - Simple precisión

    n = 32 bits, ne = 8 bits, nm = 23 bits

  - Doble precisión

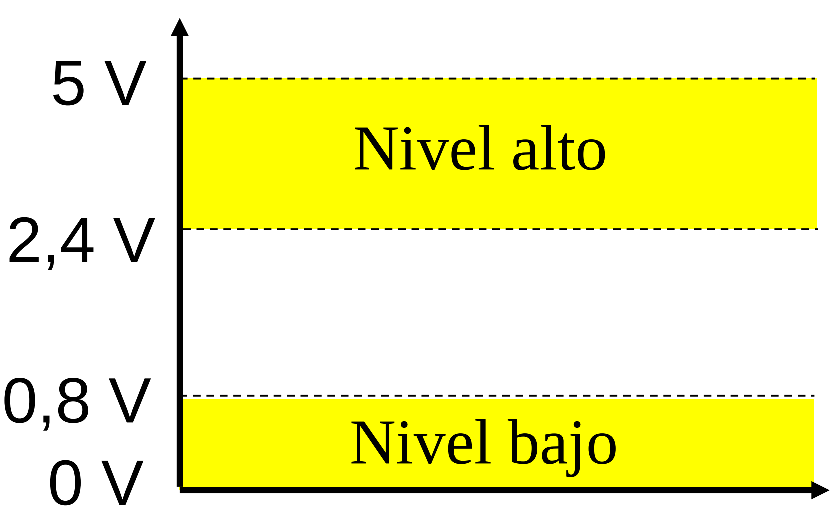
    n = 64 bits, ne = 11 bits, nm = 52 bits

  - Cuádruple precisión:

    n = 128 bits, ne = 15 bits, nm = 112 bits

Representación binaria de datos e instrucciones

- Magnitudes
  - **Analógicas**: toma valores continuos
  - **Digitales**: toma un conjunto de valores discreto
  - La mayoría de las magnitudes físicas son de tipo analógico
- Sistema digital binario
  - Representación de las magnitudes en base 2
  - Estados de un interruptor [ENCENDIDO, APAGADO]
  - Los dígitos {0, 1} corresponden con niveles de tensión eléctrica.



Niveles lógicos de la familia tecnológica TTL

2. Codificación binaria

Características de los espacios de representación

- Condicionantes
  - Cantidad de estados representables (digital, binario)
  - Cantidad de elementos representables (espacio material finito)
  - Tamaños predefinidos en las unidades del computador
  - Tamaños predefinidos en la comunicación entre unidades del computador

- Unidades de codificación

BIT	**Byte = 8 bits**	**Palabra**
1 KiloByte	**(KB) = 2^{10} Bytes =**	**1024 Bytes**
1 MegaByte	**(MB) = 2^{20} Bytes =**	**1024 KB**
1 GigaByte	**(GB) = 2^{30} Bytes =**	**1024 MB**
1 TeraByte	**(TB) = 2^{40} Bytes =**	**1024 GB**
1 PetaByte	**(PB) = 2^{50} Bytes =**	**1024 TB**