


Representación de la Información (1)

Alberto Ruiz Cristina

Introducción

- Para comunicarse, usamos...
 - Máquinas: información binaria (0/1)
 - Personas: cifras y texto
- Necesitamos un sistema de codificación:
Datos  Bits
- El BIT: la unidad mínima de información

Introducción

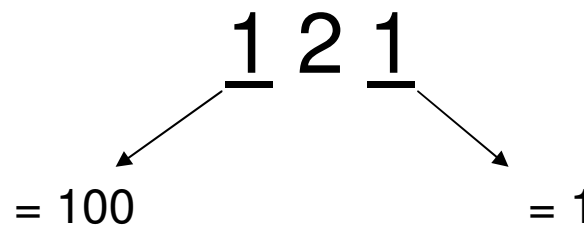
- Criterios para escoger el sistema de codificación:
 - Coste de Traducción
 - ¿Cuánto cuesta pasar de datos a bits o viceversa?
 - Coste de Almacenamiento
 - ¿Cuántos recursos son necesarios para almacenar la representación de un dato?
 - Coste de Tratamiento
 - ¿Cuánto cuesta operar con los valores representados según el código elegido?

Sistema de Numeración

- Conjunto de símbolos y reglas de generación que permiten construir todos los números válidos en el sistema.
- Es decir...
 - Proporciona:
 - Un conjunto de símbolos permitidos
 - Un conjunto de reglas que nos indican cómo formar números válidos.

Sistema de Numeración

- Dos grandes tipos:
 - Posicionales (los más utilizados)
 - El valor numérico de una combinación de dígitos depende
 - del valor de los dígitos
 - de la posición de cada uno de ellos respecto a un punto de referencia.
 - Ejemplo:



Sistemas Posicionales

- Número: cadena de dígitos en los que cada dígito está afectado por un factor; este factor depende del lugar que el dígito ocupa en la cadena.
- Base: número de símbolos permitidos en un sistema de numeración posicional.
 - Base 10:
 - Disponemos de 10 símbolos diferentes para escribir los números
 - 10 unidades forman una unidad de orden superior

Sistemas Posicionales

- Punto de referencia: cada dígito depende de su posición en una secuencia. Se toma un origen para esta secuencia:
 - La coma (,) en países latinos
 - El punto (.) en países de habla inglesa.
- Cuando se quiere explicitar la base al escribir un número:

135,64₍₁₀₎ 1011,1₍₂₎ 1437,3₍₈₎

Sistema Decimal

- número **N** con **p** dígitos enteros y **q** fraccionarios, expresado en base **b** y siendo **a_i** el valor decimal del dígito situado en la posición *i*.
- La siguiente expresión polinomial describe el valor del número y se denomina **ecuación general decimal de los sistemas de numeración.**

$$N_{(10)} = a_{p-1}b^{p-1} + a_{p-2}b^{p-2} + \dots + a_i b^i + \dots + a_1 b^1 + a_0 b^0 + a_{-1} b^{-1} + \dots + a_{-q} b^{-q}$$

$$125,2_{(10)} = 1 \cdot 10^2 + 2 \cdot 10^1 + 5 \cdot 10^0 + 2 \cdot 10^{-1} = 100 + 20 + 5 + 0,2 = 125,2_{(10)}$$

Sirve para obtener el valor decimal de números expresados en cualquier base:

$$1011,01_{(2)} = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} = 8 + 2 + 1 + 0,25 = 11,25_{(10)}$$

Sistema Decimal

- En general, para convertir números de una base a otra lo mejor suele ser:

Base X  Base 10  Base Y

- Hay métodos abreviados para algunas bases que se verán después

Conversión Decimal – Otras bases

- Otra Base \rightarrow Decimal: ecuación general
- Decimal \rightarrow Otra Base:
 - Parte entera: “divisiones sucesivas”
 - Parte decimal: “multiplicaciones sucesivas”

Divisiones sucesivas

- Dividimos el número por la base hasta que el cociente sea 0
- Los restos serán el número, desde el último hasta el primero

Ejemplo: $10_{(10)} = \text{¿¿??}_{(2)}$

Divisiones sucesivas

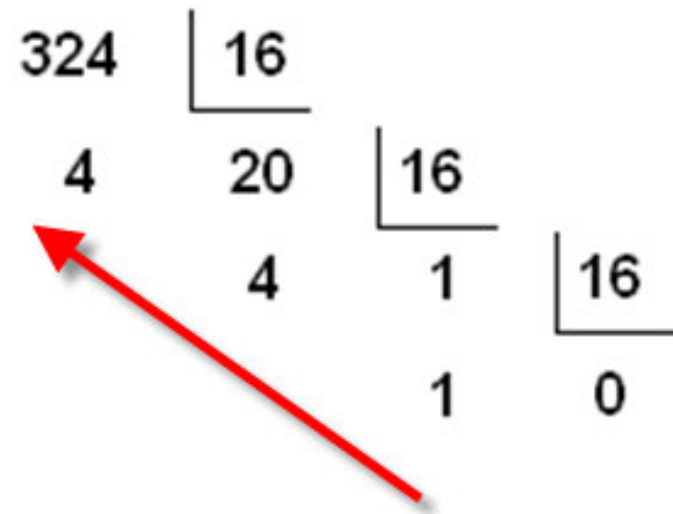
- Vale para todas las bases, incluso si son mayores que 10.

Ejemplo: $324_{(10)} = \text{¿¿??}_{(16)}$

Divisiones sucesivas

- Vale para todas las bases, incluso si son mayores que 10.

Ejemplo: $324_{(10)} = 144_{(16)}$



Multiplicaciones sucesivas

- Multiplicamos la parte fraccionaria del número por la base
- La parte entera de los resultados parciales, desde el primero hasta el último, serán el número
- El proceso puede terminar...
 - Si la parte fraccionaria del último cálculo es 0
 - Si hemos detectado un ciclo
 - Si no necesitamos más decimales

Multiplicaciones sucesivas

- Ejemplo: $0,375_{(10)} = \text{¿¿??}_{(2)}$

Multiplicaciones sucesivas

• Ejemplo: $0,375_{(10)} = 0,011_{(2)}$

• $0,375 * 2 = 0,75$

• $0,75 * 2 = 1,5$

• $0,5 * 2 = 1,0$

En las multiplicaciones sólo se utiliza la parte fraccionaria

Nos detenemos por ser 0 la parte fraccionaria

Multiplicaciones sucesivas

- Ejemplo: $0,2_{(10)} = 0,¿¿??_{(2)}$

Multiplicaciones sucesivas

- Ejemplo: $0,2_{(10)} = \overbrace{0,0011}_{(2)}$
 - $0,2 * 2 = 0,4$
 - $0,4 * 2 = 0,8$
 - $0,8 * 2 = 1,6$
 - $0,6 * 2 = 1,2$
 - $0,2 * 2 = 0,4$

Nos detenemos
por detectar un
ciclo

Multiplicaciones sucesivas

- Ejemplo: $0,2(10 = \overbrace{0,0011}^{(2)}$
- Un valor exacto en una base puede ser periódico en otra.
- Puesto que el espacio de almacenamiento es finito, necesariamente tendremos que alterar el resultado para almacenarlo:
 - Redondeando
 - Ejemplo: $0.\overbrace{46} = 0.46465$
 - Truncando
 - Ejemplo: $0.\overbrace{46} = 0.46464$
- A veces no podremos obtener una representación exacta del número
- Al operar con él aparecerán errores que no habrían ocurrido si hubiésemos operado en base decimal.

Sistema Binario

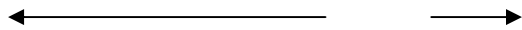
- Es el que más coste de almacenamiento tiene:
 - $138 = 10001010$
 - $4102 = 1000000000110$
 - $547.505 = 10000101101010110001$
 - $89.254.271 = 101010100011110100101111111$
- El coste de tratamiento es muy bajo: operar es muy sencillo

Sistema Binario

- Conversión Binario → Decimal
 - Se puede usar la ecuación general
 - Existe un método abreviado. Los factores de multiplicación serán 0 ó 1
 - Disponemos los pesos.
 - Si hay 1 se suma
 - Si hay 0 no se suma

Sistema Binario

- Conversión Binario → Decimal
 - Ejemplo: obtener el valor decimal del número binario 10101011,11



Ocho pesos a la izquierda de la coma, dos pesos a la derecha









2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}
128	64	32	16	8	4	2	1	0,5	0,25



Aquí
estará
la coma

Sistema Binario

- Conversión Binario \rightarrow Decimal
 - Ejemplo: obtener el valor decimal del número binario 10101011,11

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>	
1	0	1	0	1	0	1	1	,	1	1
										

$$128 + 32 + 8 + 2 + 1 + 0,5 + 0,25 = 171,75$$

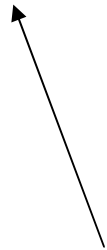
Sistema Binario

- Conversión Decimal → Binario
 - Se pueden usar las divisiones y multiplicaciones sucesivas
 - Si se prefiere, se puede usar el mismo método abreviado
 - Disponemos los pesos
 - De izquierda a derecha, vamos poniendo 1s y 0s
 - Distinguimos parte entera y parte decimal

Sistema Binario

- Conversión Decimal → Binario
 - Ejemplo: obtener el valor binario del número 171.75

128 64 32 16 8 4 2 1 0,5 0,25



Disponemos los pesos a partir del primer número menor que el pedido. El siguiente sería $2^8 = 256$, que sería demasiado grande

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
<hr/>									
1									

¿171 \geq 128? Sí \rightarrow Ponemos un 1 en 128 y nos queda 171 – 128 = 43

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
<u>1</u>	<u>0</u>								

¿43 \geq 64? No \rightarrow Ponemos un 0 en 64

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1							

¿ $43 \geq 32$? Sí \rightarrow Ponemos un 1 en 32 y nos queda $43 - 32 = 11$

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1	0						

¿11 \geq 16? No \rightarrow Ponemos un 0 en 16

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1	0	1					

¿ $11 \geq 8$? Sí \rightarrow Ponemos un 1 en 8 y nos queda $11 - 8 = 3$

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1	0	1	0				

¿3 \geq 4? No \rightarrow Ponemos un 0 en 4

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1	0	1	0	1			

¿ $3 \geq 2$? Sí \rightarrow Ponemos un 1 en 2 y nos queda $3 - 2 = 1$

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1	0	1	0	1	1		

¿ $1 \geq 1$? Sí \rightarrow Ponemos un 1 en 1 y hemos terminado la parte entera

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1	0	1	0	1	1	1	

¿ $0.75 \geq 0.5$? Sí \rightarrow Ponemos un 1 en 0.5 y nos queda $0.75 - 0.5 = 0.25$

Sistema Binario

- Conversión Decimal \rightarrow Binario
 - Ejemplo: obtener el valor binario del número 171.75

<i>128</i>	<i>64</i>	<i>32</i>	<i>16</i>	<i>8</i>	<i>4</i>	<i>2</i>	<i>1</i>	<i>0,5</i>	<i>0,25</i>
1	0	1	0	1	0	1	1	1	1

¿ $0.25 \geq 0.25$? Sí \rightarrow Ponemos un 1 en 0.25
y hemos terminado

Sistema Octal y Hexadecimal

- Permiten representar los datos de forma más compacta
 - El sistema octal utiliza 8 símbolos (0,1,2,3,4,5,6,7)
 - El 8 se representará como 10, el 9 como 11...
 - El sistema hexadecimal utiliza 16 símbolos (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)
 - El 15 será F, el 16 será 10, el 17 será 11...

Sistema Octal y Hexadecimal

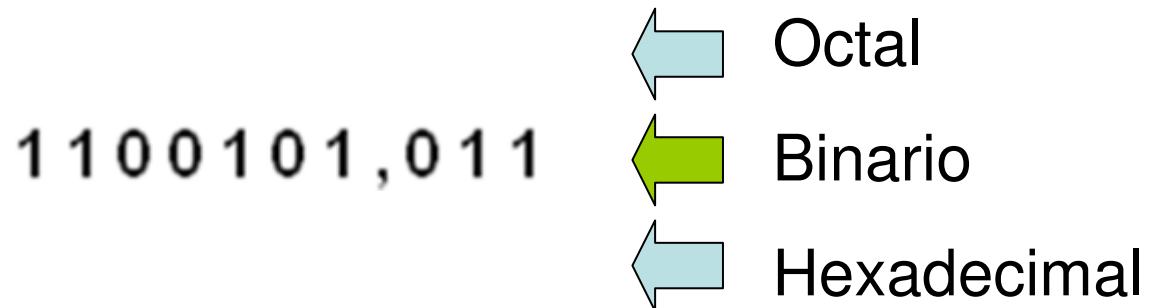
- Al ser potencias de 2, tienen estrecha relación con el sistema binario
- La conversión entre bases en las que una es potencia de la otra ($2^3 = 8$, $2^4 = 16$) es inmediata

Sistema Octal y Hexadecimal

- Conversión Binario → Octal / Hexadecimal
 - Agrupar sus dígitos en grupos de 3 (octal) ó 4 (hexadecimal)
 - A partir del punto de referencia (la coma):
 - Agrupar hacia la izquierda la parte entera
 - Agrupar hacia la derecha la parte decimal
 - Si es necesario se completa con 0s

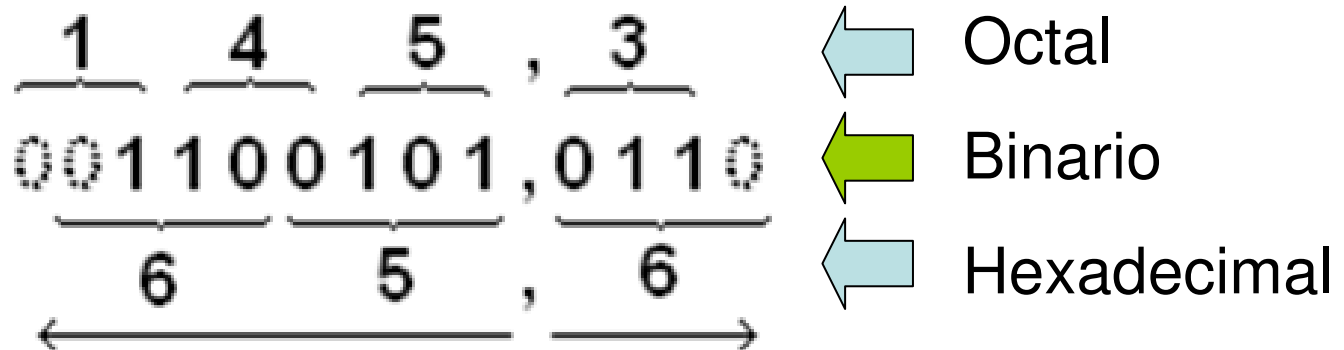
Sistema Octal y Hexadecimal

- Conversión Binario → Octal / Hexadecimal
 - Ejemplo: convertir $1100101,011_{(2)}$ a octal y hexadecimal:






Sistema Octal y Hexadecimal

- Conversión Binario → Octal / Hexadecimal
 - Ejemplo: convertir $1100101,011_{(2)}$ a octal y hexadecimal:



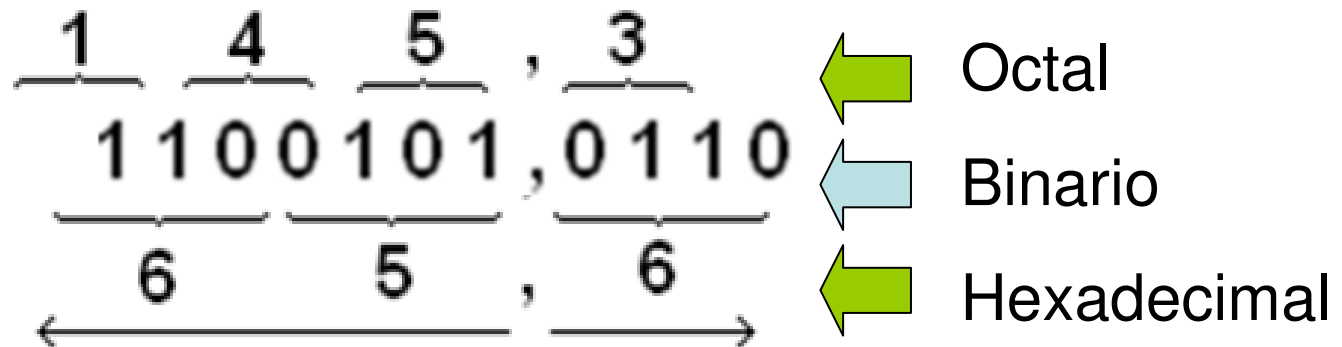
Sistema Octal y Hexadecimal

- Conversión Octal / Hexadecimal → Binario
 - Lo único que hay que hacer es expresar cada número en binario, por ejemplo $6_{(16)} = 0110_{(2)}$
 - Ejemplo: convertir $145,3_{(8)}$ y $65,6_{(16)}$ a binario:

1	4	5	,	3		Octal
						Binario
6	5	,	6		Hexadecimal	

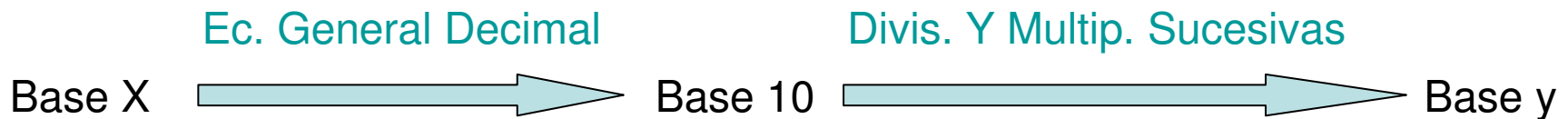
Sistema Octal y Hexadecimal

- Conversión Octal / Hexadecimal → Binario
 - Lo único que hay que hacer es expresar cada número en binario, por ejemplo $6_{(16)} = 0110_{(2)}$
 - Ejemplo: convertir $145,3_{(8)}$ y $65,6_{(16)}$ a binario:



Decimal	Hexadecimal	Octal	Binario
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	5	101
6	6	6	110
7	7	7	111
8	8	10	1000
9	9	11	1001
10	A	12	1010
11	B	13	1011
12	C	14	1100
13	D	15	1101
14	E	16	1110
15	F	17	1111
16	10	20	10000
17	11	21	10001
...
31	1F	37	11111
32	20	40	100000

En resumen...



Con esto me vale para pasar de cualquier base X a cualquier base Y

Pero para algunos cambios de base concretos hay métodos abreviados:

