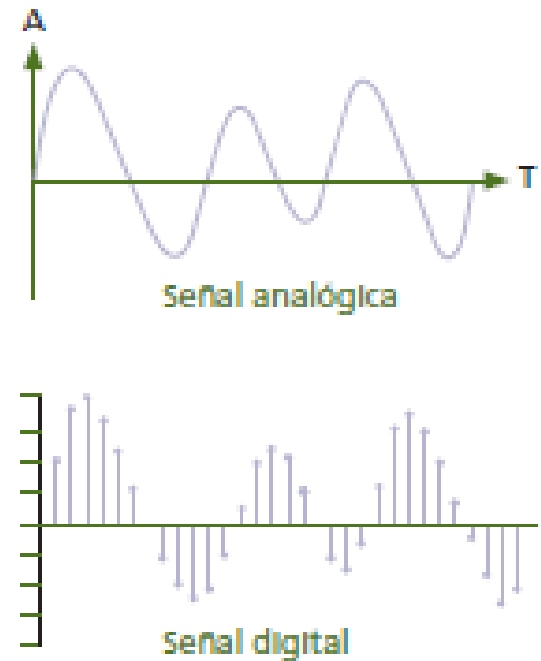


## **Tema 2 – Introducción a los Sistemas Informáticos**

CFGS DAW – Sistemas Informáticos

# Informática

- **¿Qué es?** → Ciencia que estudia el tratamiento (almacenamiento, proceso, etc.) automático de la información.
- **¿Qué es la información?** → datos/mensaje/señal
  - Analógica
  - Digital → Los ordenadores utilizan información digital
- La información es tratada por el **Sistema Informático**



# Sistema Informático

- El ordenador es una máquina compuesta de elementos físicos (**hardware**), en su mayoría de origen eléctrico-electrónico, capaz de realizar una gran variedad de trabajos a gran velocidad y con gran precisión.
- Estos componentes electrónicos necesitan de otros componentes no físicos que los pongan en funcionamiento, también llamados componentes lógicos: los programas (**software**).
- El sistema operativo es el componente software de un sistema informático capaz de hacer que los programas (*software*) procesen información (datos) sobre los componentes electrónicos de un ordenador o sistema informático (*hardware*).
- Entre *software* y *hardware* existe otro concepto importante dentro de un sistema informático: el **firmware**. Es la parte intangible (software) de componentes del *hardware*.



## Sistema Informático (II)

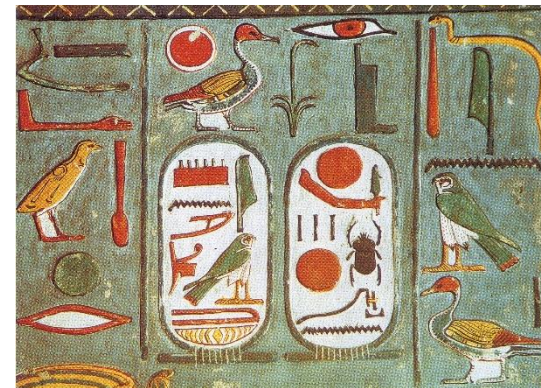
- **Evolución** → <https://www.youtube.com/watch?v=lsAeTXNQyIA>
- **Generaciones** → <https://www.youtube.com/watch?v=z5IIWdgdMEo>
- **Componentes** → <https://www.youtube.com/watch?v=r9p517ruN2s>
- **Funciones** → ¿Dónde encontramos sistemas informáticos?

## Codificación de la información

- Un **símbolo** puede ser manejado como sustituto de aquello que representa.
  - Un **código** es un conjunto de símbolos.
  - Se define **codificar** como la transformación de un dato a una representación preestablecida (símbolos) que sigue unas reglas (sintaxis).
  - Los ordenadores utilizan un **código binario** → los símbolos son 0 y 1.
- |   |      |   |       |   |      |   |       |
|---|------|---|-------|---|------|---|-------|
| A | ..-  | J | ..--- | S | ...  | 2 | ..... |
| B | .... | K | ---   | T | -    | 3 | ..... |
| C | .... | L | ....  | U | ...  | 4 | ..... |
| D | ---  | M | --    | V | .... | 5 | ..... |
| E | .    | N | --    | W | ---  | 6 | ..... |

## ¿Cómo se representan números, letras, etc. utilizando un código binario?

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



# Codificación de la información (II)

**Códigos alfanuméricos** → representación de letras, números y caracteres especiales

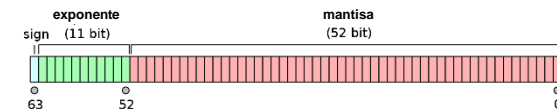
- ASCII:
  - 7 bits (8 bits → ASCII Extendido), 128 caracteres
- EBCDIC:
  - 8 bits, 256 caracteres
  - Primero que se ideó para uso en ordenadores, por IBM
- UNICODE:
  - De 8 bits, 16 bits (65536 caracteres) y 32 bits → UTF
  - Para utilizar alfabetos distintos al inglés/americano

**Códigos numéricos** → representación de cantidades utilizando sistemas de numeración

- Binario
- Complementos a 1 y a 2 → permiten la representación de números negativos
- IEEE 754 → coma flotante (notación científica)

## ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[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	[NEGATIVE 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	[ENG OF 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	[SUBSTITUTE]	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	[DEL]



# Codificación de la información (III)

**Sistemas de numeración** → conjunto de reglas y símbolos para construir todos los números válidos

- Binario:
  - Su base es 2 → utiliza 2 símbolos (0, 1)
- Octal:
  - Su base es 8 → utiliza 8 símbolos (0, 1, 2, 3, 4, 5, 6, 7)
- Decimal:
  - Su base es 10 → utiliza 10 símbolos (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
- Hexadecimal
  - Su base es 16 → utiliza 16 símbolos (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)

Decimal	Binario	Base 8	Base 16
0	00000	0	0
1	00001	1	1
2	00010	2	2
3	00011	3	3
4	00100	4	4
5	00101	5	5
6	00110	6	6
7	00111	7	7
8	01000	10	8
9	01001	11	9
10	01010	12	A
11	01011	13	B
12	01100	14	C
13	01101	15	D
14	01110	16	E
15	01111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13

# Codificación de la información (IV)

## Sistemas de numeración (cont.)

- Teorema fundamental de la numeración:

$$N_i = \sum_{i=-d}^n (\text{dígito})_i \cdot (\text{base})^i$$

donde:

- $i$  = posición respecto a la coma.
  - Para los dígitos de la derecha la  $i$  es negativa, empezando en  $-1$ ; para los de la izquierda es positiva, empezando en  $0$ .
- $d$  = número de dígitos a la derecha de la coma.
- $n$  = número de dígitos a la izquierda de la coma  $-1$ .
- dígito = cada uno de los que componen el número.
- base = base del sistema de numeración.

¿Cómo se construye el número 114 (decimal)?



# Codificación de la información (V)

## Sistemas de numeración (cont.)

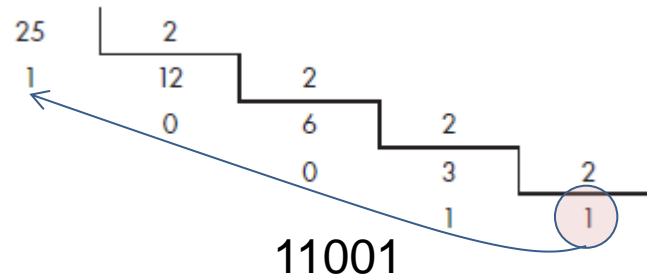
- Sistema Binario  $\leftrightarrow$  Sistema decimal

- 11001  $\rightarrow$  a decimal

	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	
Número binario	1	1	0	0	1	
Suma decimal	16	+	8	+	1	= 25

$2^0$	1
$2^1$	2
$2^2$	4
$2^3$	8
$2^4$	16
$2^5$	32
$2^6$	64
$2^7$	128
$2^8$	254
$2^9$	512
$2^{10}$	1024

- 25  $\rightarrow$  a binario



# Codificación de la información (VI)

## Sistemas de numeración (cont.)

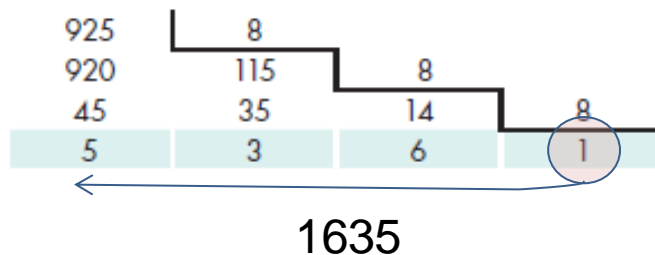
- Sistema Octal  $\leftrightarrow$  Sistema decimal

$8^0$	1
$8^1$	8
$8^2$	64
$8^3$	512

- $1635_8 \rightarrow$  a decimal

	$8^3$	$8^2$	$8^1$	$8^0$
Número octal	1	6	3	5
Suma decimal	$1 \cdot 8^3 + 6 \cdot 8^2 + 3 \cdot 8^1 + 5 \cdot 8^0 = 925$			

- $925_{10} \rightarrow$  a octal



# Codificación de la información (VII)

## Sistemas de numeración (cont.)

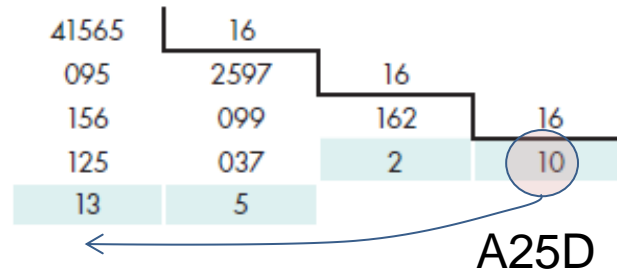
- Sistema Hexadecimal  $\leftrightarrow$  Sistema decimal

$16^0$	1
$16^1$	16
$16^2$	256
$16^3$	4096

- $A25D_{16} \rightarrow$  a decimal

	$16^3$	$16^2$	$16^1$	$16^0$
Número hexadecimal	A	2	5	D
Suma decimal	$10 \cdot 16^3 + 2 \cdot 16^2 + 5 \cdot 16^1 + 13 \cdot 16^0 = 41565$			

- $41565_{10} \rightarrow$  a hexadecimal



# Trabajando con la información: Operaciones lógicas y aritméticas binarias

- Álgebra de Boole → utiliza sistema binario
- Operaciones:

- Lógicas → e.g.: condición1 Y condición2

- OR
- AND
- NOT

- Derivadas → NOR, NAND, XOR

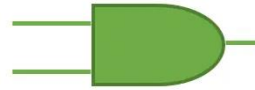
- Aritméticas → e.g.: Suma

- Suma  $\equiv$  XOR junto con el Acarreo
- Resta  $\equiv$  sumar el primer número con el complementario del segundo
  - Complemento a 1  $\equiv$  NOT al número “negativo”
  - Complemento a 2  $\equiv$  Complemento a 1 + “1” al número “negativo”

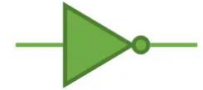
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1



A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1



Q	Q'
0	1
1	0



$$\begin{array}{r} \textcolor{red}{1} \\ + \quad 01_2 \\ \quad 01_2 \\ \hline 10_2 \end{array}$$

# Representación interna de la información

- Bit:
  - Unidad mínima de información
  - 0 (“apagado”) y 1 (“encendido”)
- *Byte*:
  - Agrupación de 8 bits
  - También se le llama “octeto”
- Palabra:
  - Tamaño de información manejada en paralelo por los componentes del sistema (memoria, registros o buses)
  - Comunes → 8, 32, 64, 128 y 256 bits.
  - A mayor tamaño de palabra en un ordenador → mayor precisión y potencia de cálculo

# Medidas de la información

## Almacenamiento de la información

- La información es almacenada (y tratada) en un sistema informático
- Existen unidades de medida de la cantidad de información
- Actualmente, la capacidad de la memoria RAM se mide en GB, y la capacidad de los discos duros en GB o TB.

[https://www.youtube.com/watch?v=8co2ADsKOUo&list=PLXZi7lIJeTrojhf1u7ZMuRIjwJZb3\\_B3b&index=6](https://www.youtube.com/watch?v=8co2ADsKOUo&list=PLXZi7lIJeTrojhf1u7ZMuRIjwJZb3_B3b&index=6)

[https://www.youtube.com/watch?v=Mv\\_Ape519ys&list=PLXZi7lIJeTrojhf1u7ZMuRIjwJZb3\\_B3b&index=7](https://www.youtube.com/watch?v=Mv_Ape519ys&list=PLXZi7lIJeTrojhf1u7ZMuRIjwJZb3_B3b&index=7)

Bit = mínima unidad de información.

4 Bits = Nibble o cuarteto.

8 Bits = 1 Byte.

1 024 Bytes = 1 Kilobyte.

1 024 Kilobytes = 1 Megabyte (MB).

1 024 Megabytes = 1 Gigabyte (GB).

1 024 Gigabytes = 1 Terabyte (TB).

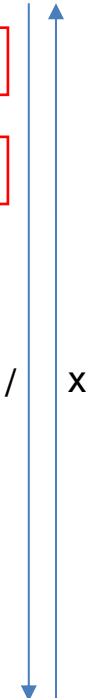
1 024 Terabytes = 1 Petabyte (PB).

1 024 Petabytes = 1 Exabyte (EB).

1 024 Exabytes = 1 Zettabyte (ZB).

1 024 Zettabytes = 1 Yottabyte (YB).

1 024 Yottabytes = 1 Brontobyte (BB).



# Medidas de la información (II)

## Almacenamiento de la información

- Más correcto:
  - Uso de “prefijos binarios”. E.g.:
    - KiB (Kibi Byte)  $\rightarrow$  Kibi = *Binary Kilo* ( $2^{10}$ )
    - MiB (Mebi Byte)  $\rightarrow$  Mebi = *Binary Mega* ( $2^{20}$ )
    - GiB (Gibi Byte)  $\rightarrow$  Gibi = *Binary Giga* ( $2^{30}$ )
  - Uso de prefijos del Sistema Internacional: E.g.:
    - KB (Kilo Byte)  $\rightarrow$  ( $10^3$ )
    - MB (Mega Byte)  $\rightarrow$  ( $10^6$ )
    - GB (Giga Byte)  $\rightarrow$  ( $10^9$ )
- Pero nosotros vamos a utilizar los prefijos del Sistema Internacional con potencias binarias (lo más habitual):
  - Esto es, 1024 KB hacen 1MB, etc.

Nombre (símbolo)	Sistema Internacional de Unidades (SI) Estándar (uso decimal)	Prefijo binario (uso binario)	Nombre (símbolo)
Kilobyte (KB)	$1000^1 = 10^3$ bytes	$1024^1 = 2^{10}$ bytes	Kibibyte (kib)
Megabyte (MB)	$1000^2 = 10^6$ bytes	$1024^2 = 2^{20}$ bytes	Mebibyte (Mib)
Gigabyte (GB)	$1000^3 = 10^9$ bytes	$1024^3 = 2^{30}$ bytes	Gibibyte (Gib)
Terabyte (TB)	$1000^4 = 10^{12}$ bytes	$1024^4 = 2^{40}$ bytes	Tebibyte (Tib)
Petabyte (PB)	$1000^5 = 10^{15}$ bytes	$1024^5 = 2^{50}$ bytes	Pebibyte (Pib)
Exabyte (EB)	$1000^6 = 10^{18}$ bytes	$1024^6 = 2^{60}$ bytes	Exbibyte (Eib)
Zettabyte (ZB)	$1000^7 = 10^{21}$ bytes	$1024^7 = 2^{70}$ bytes	Zebibyte (Zib)
Yottabyte (YB)	$1000^8 = 10^{24}$ bytes	$1024^8 = 2^{80}$ bytes	Yobibyte (Yib)

# Clasificación de los sistemas informáticos

+ Tamaño  
+ Potencia



- Superordenadores
- Macroordenadores o *mainframes*
- Servidores y estaciones de trabajo (*workstation*)
- Ordenadores personales
  - Sobremesa
  - Portátil
  - Convertible 2-1
  - *Thin clients*
  - *Single Board Computer*
    - ¿ejemplo?
- Tablets, smartphones

