

01-Representación de la información

1

Montaje y Mantenimiento de Equipos
Coordinador: Juan Carlos Valero







1. Introducción

1. Introducción

- ¿Cómo representan la información las máquinas?
- El sistema binario.
- Solamente los dígitos cero y uno (0 y 1).
- En computadores dos niveles de voltaje.
 - Encendido → 1
 - Apagado → 0



The background of the slide features a collection of wooden number blocks, including digits 0 through 9, arranged in a scattered pattern. A green arrow points from the left edge towards the center, passing behind the main title. The title itself is in a large, bold, green font.

2. Representar números

2. Representación de números

- Se define un sistema de numeración como:
 - El conjunto de símbolos utilizados para la representación de cantidades.
 - Las reglas que rigen dicha representación.



2. Representación de números

- Se define la **base de un sistema de numeración**:
 - El número de dígitos distintos que se utilizan en ese sistema.



2.1 Sistema decimal

- Base 10: Usa diez dígitos que son 0, 1, 2, 3, 4, 5, 6, 7, 8 y 9.
- ¿Por qué usamos este sistema?



2.1 Sistema decimal

- Sistema posicional
 - Cada dígito tiene un peso (un valor) en función de la posición que ocupa.

$$1234_{10} = 1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0 = 1234$$

$$325,24_{10} = 3 \times 10^2 + 2 \times 10^1 + 5 \times 10^0 + 2 \times 10^{-1} + 4 \times 10^{-2} = 325,24$$



2.2 Sistema binario

- La forma más conveniente de representar la información en un computador es el sistema binario.
- Base 2: Utiliza sólo dos dígitos, el 0 y 1
- Cada dígito se le denomina BIT
 - **BI**nary digi**T** → dígito binario

2.2 Sistema binario

➡ Sistema posicional.

$$1011,01_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} =$$
$$8 + 0 + 2 + 1 + 0 + 0,25 = 11,25_{10}$$

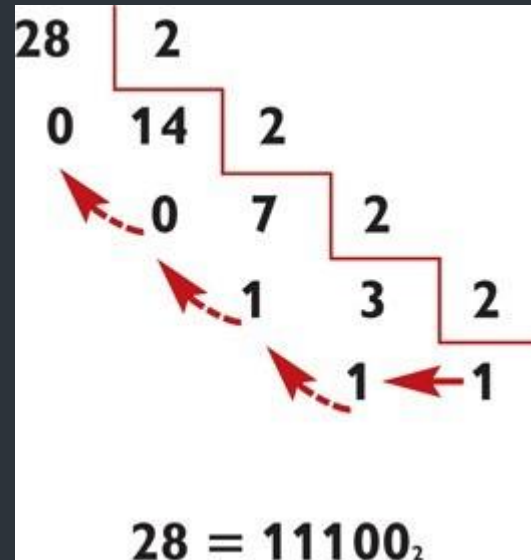
2.2 Sistema binario

- Convertir de **decimal a binario**. Método 1.
- Se realizan divisiones sucesivas por la base(2), hasta que el cociente sea menor que 2.
- La unión del último cociente y todos los restos escritos en orden inverso será el número expresado en binario.



2.2 Sistema binario

- Ejemplo de conversión del número decimal **28** a binario por el método 1:



2.2 Sistema binario

➤ Convertir a binario por el método 1 los números:

➤ 30 →

➤ 55 →

➤ 101 →



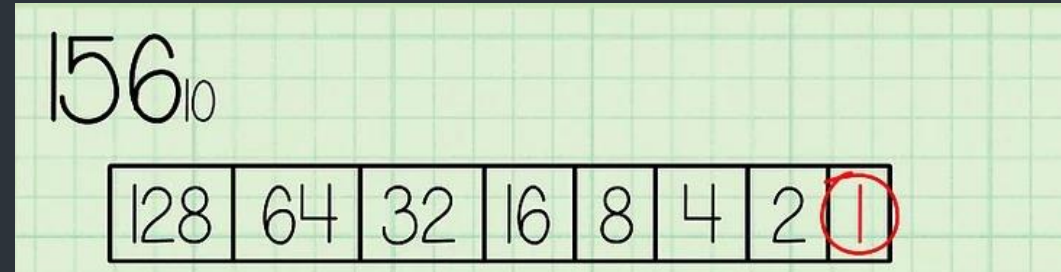
2.2 Sistema binario

- Convertir de **decimal a binario**. Método 2.
- Parece más complicado pero realmente es más sencillo.
- Se hace una tabla partiendo desde el 1 y se van escribiendo a la izquierda sucesivamente el doble del anterior hasta llegar al que sea mayor que el número decimal buscado.

156₁₀

128	64	32	16	8	4	2	1
-----	----	----	----	---	---	---	---

2.2 Sistema binario



156₁₀

128	64	32	16	8	4	2	1
-----	----	----	----	---	---	---	---

- Partiendo desde la izquierda.
- Se compara el número de la tabla con el que se quiere convertir.
- Si el número de la tabla es menor o igual que el buscado.
 - Se marca esa posición como un 1 y se restan los dos números. El resultado es el nuevo número a convertir.
- Si el número de la tabla es mayor que el buscado.
 - Se marca esa posición como un 0 y se pasa a la siguiente posición de la tabla
- Se acaba cuando el número buscado es 0.

156_{10}

128	64	32	16	8	4	2	1
-----	----	----	----	---	---	---	---

1 0 0 1 1 1 0 0

$$156 - 128 = 28$$

$$28 - 16 = 12$$

$$12 - 8 = 4$$

$$4 - 4 = 0$$



Paso de binario a decimal.
Método 2.

2.2 Sistema binario

➤ Convertir a binario por el método 2 los números:

➤ 30 →

➤ 55 →

➤ 101 →



2.2 Sistema binario

- Para **convertir** una cifra de **binaria a decimal** se utiliza el teorema fundamental de la numeración.
- El resultado es la suma de los productos de los resultados de multiplicar cada dígito por la base elevado a la posición que ocupa.



2.2 Sistema binario

$$\text{Ej.: } 110,0011_2 = 6,1875_{10}$$

$$\begin{array}{cccccccccccc} 1 & \times & 2^2 & + & 1 & \times & 2^1 & + & 0 & \times & 2^0 & + & 0 & \times & 2^{-1} & + & 0 & \times & 2^{-2} & + & 1 & \times & 2^{-3} & + & 1 & \times & 2^{-4} \\ 4 & + & 2 & + & 0 & + & 0 & + & 0 & + & 0 & + & 0 & + & 0,125 & + & 0,0625 & = & 6,1875 \end{array}$$



128	64	32	16	8	4	2	1
1	0	0	1	1	0	1	1

$128 + 0 + 0 + 16 + 8 + 0 + 2 + 1$

$= 155$

2.2 Sistema binario

- Convertir a decimal los números obtenidos en los ejercicios anteriores.



2.2 Sistema binario

DECIMAL	BINARIO
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010

2.2 Sistema binario

➤ Operaciones en binario

SUMA	RESTA	PRODUCTO	DIVISIÓN
$0 + 0 = 0$	$0 - 0 = 0$	$0 * 0 = 0$	$0 / 0 = \text{ERROR}$
$0 + 1 = 1$	$0 - 1 = 11$	$0 * 1 = 0$	$0 / 1 = 0$
$1 + 0 = 1$	$1 - 0 = 1$	$1 * 0 = 0$	$1 / 0 = \text{ERROR}$
$1 + 1 = 10$	$1 - 1 = 0$	$1 * 1 = 1$	$1 / 1 = 1$



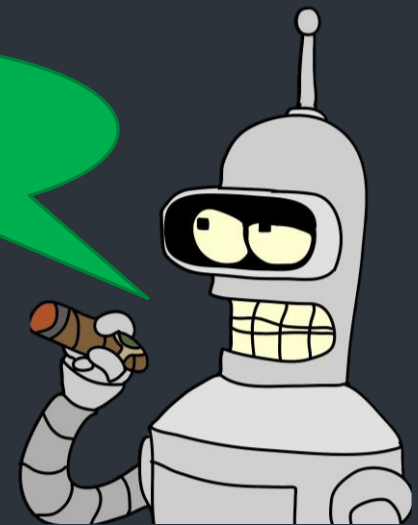
desmotivaciones.es

Sólo hay 10 tipos de personas
las que saben binario y las que no

2.2 Sistema binario

- Representación de números enteros.
- Es necesario la representación del signo.
- Se suele emplear el dígito más a la izquierda posible.
- Si es 0 es positivo y si es 1 negativo.

Demasiado
fácil



2.2 Sistema binario

- Signo y magnitud (SM)

- El signo se representa en el bit que está más a la izquierda del dato.

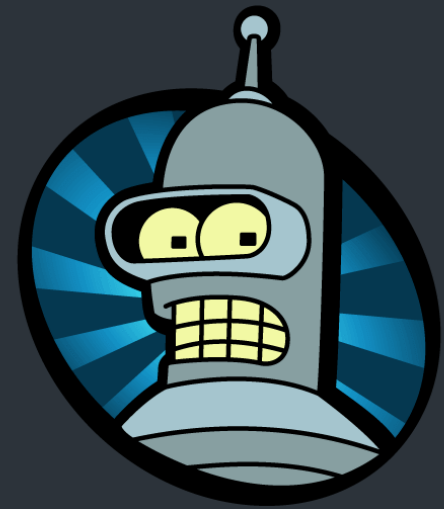
- En el resto de los bits se representa el valor del número en binario natural.

- Problemas:

- Doble representación del 0. (0000_2 y 1000_2)

- Circuitos diferentes para sumas y restas.

- $+6=0110_2$ $-6=1110_2$



2.2 Sistema binario

- Complemento a 1 (Ca1)
- El complemento a 1 de un número binario se obtiene cambiando cada 0 por 1 y viceversa.
- Doble representación del cero.
- Las restas son sumas.
- $\text{Ca1}(-7) = \text{INVERTIR}(7) = \text{INVERTIR}(0111) = 1000$

Complemento a uno	Decimal
0111	+7
0110	+6
0101	+5
0100	+4
0011	+3
0010	+2
0001	+1
0000	+0
1111	-0
1110	-1
1101	-2
1100	-3
1011	-4
1010	-5
1001	-6
1000	-7

2.2 Sistema binario

- Complemento a 2 (Ca2)
- Se calcula sumando 1 al Ca1 del número positivo.
- Las restas son sumas.
- No hay doble representación del 0.

Complemento a dos	Decimal
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	-8

2.2 Sistema binario

Se añaden tantos
ceros hasta llegar al
número de bits de la
máquina

Si el número es
positivo los tres
son iguales

- Convertir a binario en una máquina de cinco bits.

Decimal	Signo-Magnitud	Complemento a 1	Complemento a 2
+5	00101	00101	00101
-5	10101	11010	11011

Igual que el positivo
cambiando el primer
dígito de la izquierda
por 1

Igual que el positivo
cambiando todos los
0s por 1s y viceversa

Se suma 1 al
Ca1

2.2 Sistema binario

```

110111
+1
-----
111000
Como 1+1=10
Da 0 y me llevo 1

```

- Convertir a binario en una máquina de seis bits.

Decimal	Signo-Magnitud	Complemento a 1	Complemento a 2
+8	001000 ▼	001000	001000
-8	101000	110111	111000
+21	010101 ▼	010101	010101
-21	110101	101010	101011
+33	Sería 0100001, pero son necesarios siete bits para representarlo. Se produce un error llamado overflow (desbordamiento)		

2.2 Sistema binario

- Convertir a binario en una máquina de seis bits.

Decimal	Signo-Magnitud	Complemento a 1	Complemento a 2
+9			
-9			
+22			
-23			
+35			

2.3 Sistema octal

- **Base 8:** Utiliza ocho dígitos, el 0, 1, 2, 3, 4, 5, 6 y 7.



2.3 Sistema octal

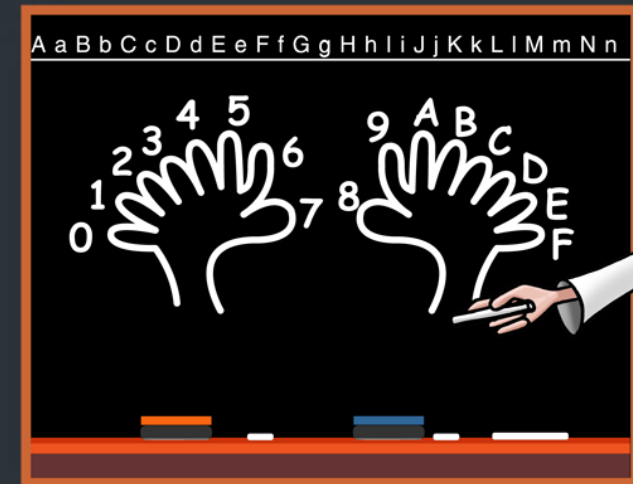
- ¿Por qué es importante este sistema?
- Como $2^3=8$, significa que un dígito en base 8 se corresponde con 3 dígitos binarios.
- Es una manera más corta de trabajar con binarios.

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



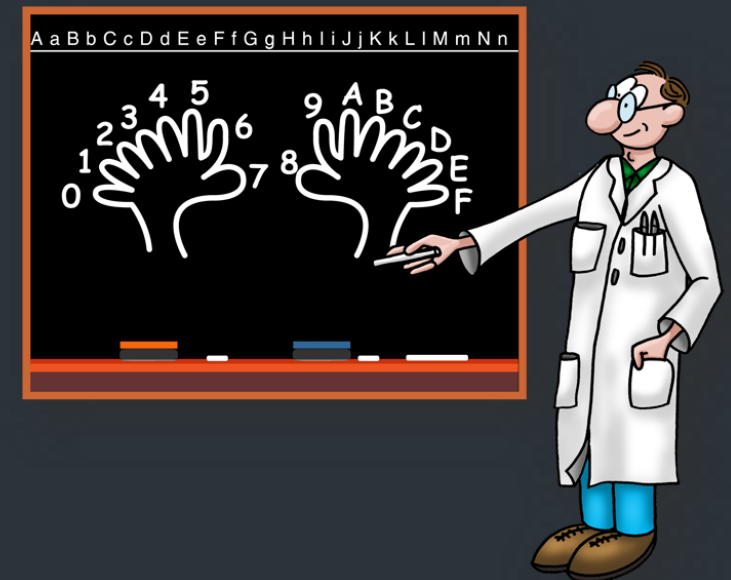
2.4 Sistema hexadecimal

- **Base 16:** Utiliza 16 dígitos.
- Se añaden letras de la A a la F.
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E y F.



2.4 Sistema hexadecimal

- Al igual que el sistema octal, hay una correspondencia entre el sistema binario y el hexadecimal.
- Como $2^4 = 16$, cada dígito hexadecimal se puede representar en binario con 4 dígitos.



2.4 Sistema hexadecimal



Decimal	Binario	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	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

2.5 Conversión de sistemas

Empezamos siempre por cero.

Decimal	Binario					Octal	Hexadecimal
0	0	0	0	0	0	0	0
1	0	0	0	1	1	1	1
2	0	0	1	0	2	2	2
3	0	0	1	1	3	3	3
4	0	1	0	0	4	4	4
5	0	1	0	1	5	5	5
6	0	1	1	0	6	6	6
7	0	1	1	1	7	7	7
8	1	0	0	0	10	10	8
9	1	0	0	1	11	11	9
10	1	0	1	0	12	12	A
11	1	0	1	1	13	13	B
12	1	1	0	0	14	14	C
13	1	1	0	1	15	15	D
14	1	1	1	0	16	16	E
15	1	1	1	1	17	17	F

2.5 Conversión de sistemas

- Convertir de decimal a octal. Ej.: 1011000111
- Se toman grupos de tres dígitos partiendo por la derecha y se usa la tabla de conversión.

Binario	1	011	000	111
Octal	1	3	0	7

➤ $1011000111_2 \rightarrow 1307_8$

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

2.5 Conversión de sistemas

- Convertir de octal a decimal. Ej.: 1234
- Cada dígito se sustituye por su equivalente de la tabla de conversión.

Octal	1	2	3	4
Binario	001	010	011	100

➤ $1234_8 \rightarrow 1010011100_2$

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

2.5 Conversión de sistemas

- Convertir de decimal a hexadecimal. Ej.: 10 1100 0111
- Se toman grupos de cuatro dígitos partiendo por la derecha y se usa la tabla de conversión.

Binario	0010	1100	0111
Hexadecimal	2	C	7

➤ $1011000111_2 \rightarrow 2C7_{16}$

Decimal	Binario	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	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

2.5 Conversión de sistemas

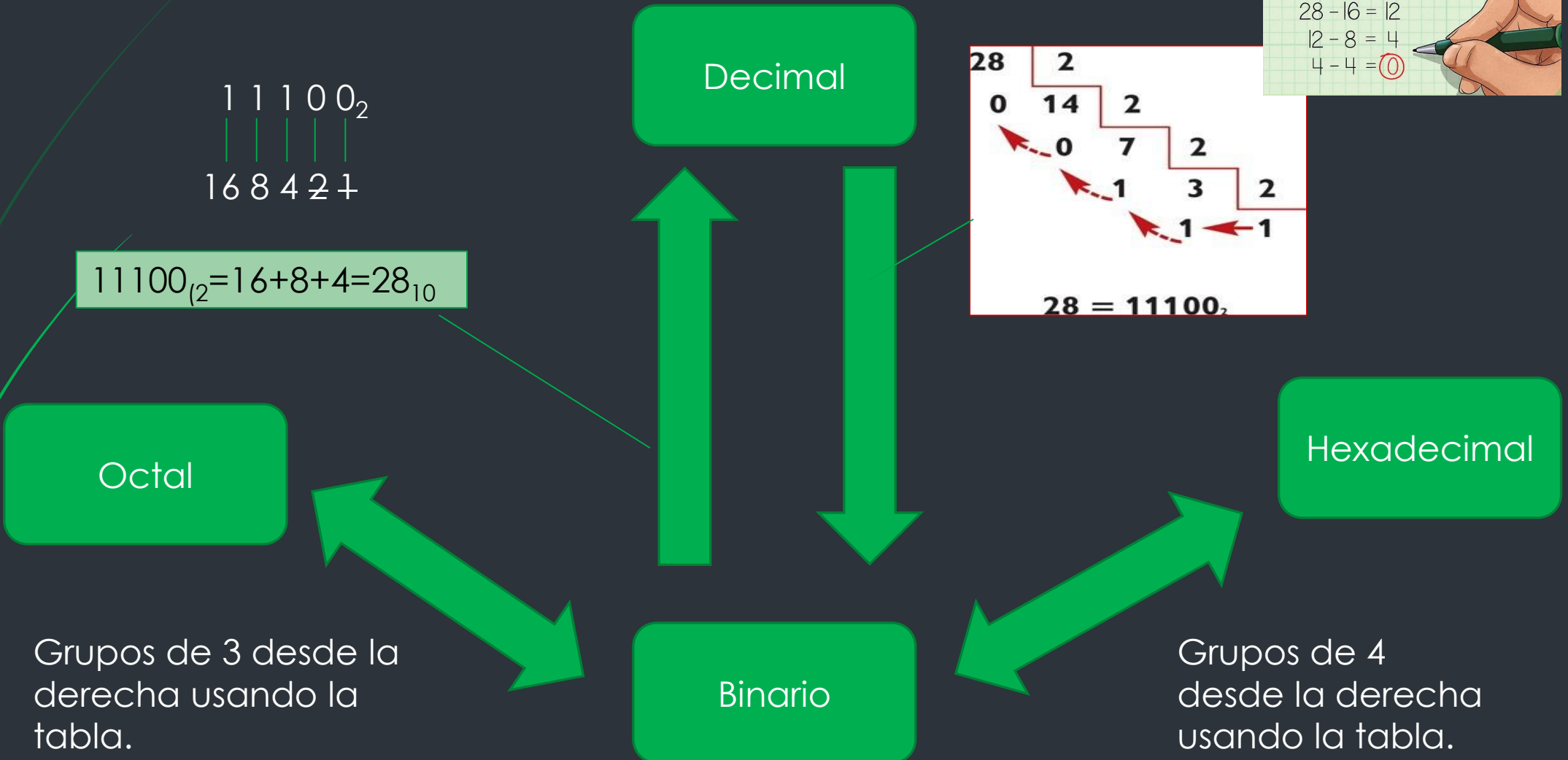
- Convertir de hexadecimal a decimal. Ej.: BEBE
- Cada dígito se sustituye por su equivalente de la tabla de conversión.

Hexadecimal	B	E	B	E
Binario	1011	1110	1011	1110

➤ $\text{BEBE}_{16} \rightarrow 1011111010111110_2$

Decimal	Binario	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	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

2.5 Conversión de sistemas



2.5 Conversión de sistemas

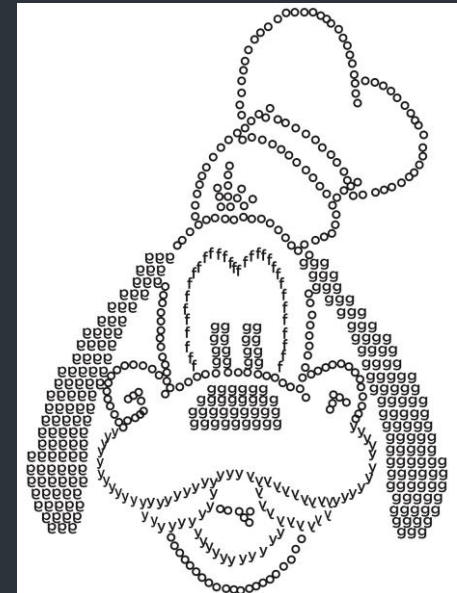
➤ Rellenar la siguiente tabla con todas las conversiones.

Binario	Octal	Hexadecimal	Decimal
			74
10100111			
	107		
		111	
			424
111101111			
	665		
		16F	

3. Representación alfanumérica.

3. Representación alfanumérica

- La información que un usuario introduce en un ordenador está formado por letras, números y otros signos.
- Para representar esta información se usan los **estándares de codificación**.



3.1 ASCII

- American Standard Code for Information Interchange
- Código Estadounidense Estándar para el Intercambio de Información.
- Utiliza grupos de **7 bits** por carácter, permitiendo $2^7=128$ **caracteres diferentes**.

ASCII

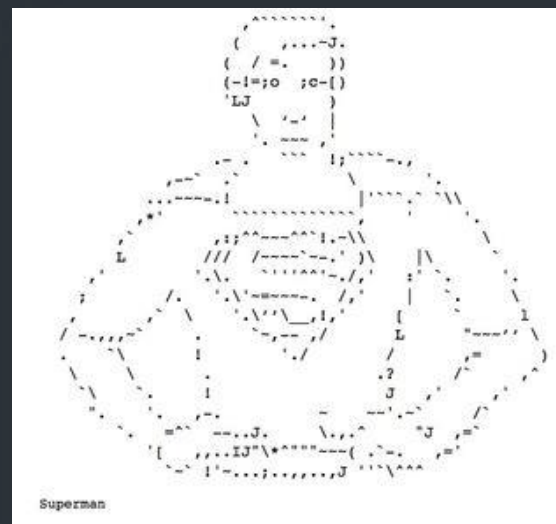
3.1 ASCII

- Las primeras 32 combinaciones están reservadas a caracteres de control como:
 - Saltos de página, retorno de carro, borrado de pantalla, etc.
 - Desde ésta hasta la 128 están representadas todas las letras (mayúsculas y minúsculas), números y demás caracteres.

ASCII

3.1 ASCII

- ASCII era muy limitado.
- **ASCII Extendido**, usa **8 bits** por carácter.
- $2^8 = 256$ caracteres.
- Añade símbolos latinos de idiomas diferentes del inglés y algún símbolo gráfico.



3.1 ASCII

1	25	49	73	97	121	145	169	193	217	241
2	26	50	74	98	122	146	170	194	218	242
3	27	51	75	99	123	147	171	195	219	243
4	28	52	76	100	124	148	172	196	220	244
5	29	53	77	101	125	149	173	197	221	245
6	30	54	78	102	126	150	174	198	222	246
7	31	55	79	103	127	151	175	199	223	247
8	32	56	80	104	128	152	176	200	224	248
9	33	57	81	105	129	153	177	201	225	249
10	34	58	82	106	130	154	178	202	226	250
11	35	59	83	107	131	155	179	203	227	251
12	36	60	84	108	132	156	180	204	228	252
13	37	61	85	109	133	157	181	205	229	253
14	38	62	86	110	134	158	182	206	230	254
15	39	63	87	111	135	159	183	207	231	255
16	40	64	88	112	136	160	184	208	232	PRESIONA LA TECLA
17	41	65	89	113	137	161	185	209	233	Alt
18	42	66	90	114	138	162	186	210	234	MÁS EL NUMERO
19	43	67	91	115	139	163	187	211	235	CORTESÍA DE:
20	44	68	92	116	140	164	188	212	236	
21	45	69	93	117	141	165	189	213	237	
22	46	70	94	118	142	166	190	214	238	
23	47	71	95	119	143	167	191	215	239	
24	48	72	96	120	144	168	192	216	240	

3.1 ASCII

- En el bloc de notas aprieta la tecla ALT y sin soltar escribe 65. ¿Qué aparece? ¿Por qué?



3.2 UNICODE

- **UNICODE (UNiversal CODE).**
- Es un código de 16 bits.
- Es uno de los más modernos y usados.
- Es un estándar.
- Compatible con muchos SO y con todos los exploradores actuales.



3.2 UNICODE

- Objetivo: codificar cualquier idioma para el uso informático.
- Codifica distintos juegos de caracteres (árabe, cirílico, japonés, emojis, etc.).



3.2 UNICODE

- En un editor de texto de Windows, aprieta la tecla WIN+.
- ¿Qué aparece? ¿Funciona? ¿Por qué es eso?
- Prueba a cambiar un nombre de archivo con este truco.



4. Representación multimedia

4. Representación multimedia

- Toda información multimedia se representa mediante código binario siguiendo normas propias.
- Imagen: BMP, JPEG, GIF, PNG,...
- Audio: WAV, MP3, OGG...
- Vídeo: MPEG, AVI, MP4, MKV,...
- Ofimática: DOCX, ODT, XLSX, OST,...
- Otros: SVG, XML, HTML, CSS,...



4. Representación multimedia

➤ ¿Para qué se usan las siguientes extensiones?

- PDF
- CSV
- EXE
- DLL
- ZIP
- RAR
- ISO



5. Medidas de la información

5. Medidas de la información

- Los ordenadores disponen de dispositivos para almacenar la información.
- Dichos dispositivos son elementos físicos con una capacidad determinada.
- Para medir dicha capacidad utilizamos una serie de medidas.

5. Medidas de la información

- La unidad mínima de información es el **bit**.
- Tiene dos valores posibles: 0 y 1.
- 8 bits son un byte.
- Antiguamente representaba a un único símbolo (ASCII).
- A partir del byte existen otras medidas múltiplos de ésta.

5. Medidas de la información

- Hay dos tipos de medidas.
- Las del Sistema Internacional que van de 1000 en 1000.
- Las de la ISO que van de 1024 en 1024.
- Aunque casi siempre se nombran las del SI, los equipos usan internamente las ISO.
- Por comodidad, usaremos las del SI.

5. Medidas de la información

Nombre	Símbolo	Factor y valor en el SI
byte	B	$10^0 = 1$
kilobyte	kB	$10^3 = 1\ 000$
megabyte	MB	$10^6 = 1\ 000\ 000$
gigabyte	GB	$10^9 = 1\ 000\ 000\ 000$
terabyte	TB	$10^{12} = 1\ 000\ 000\ 000\ 000$
petabyte	PB	$10^{15} = 1\ 000\ 000\ 000\ 000\ 000$
exabyte	EB	$10^{18} = 1\ 000\ 000\ 000\ 000\ 000\ 000$
zettabyte	ZB	$10^{21} = 1\ 000\ 000\ 000\ 000\ 000\ 000\ 000$
yottabyte	YB	$10^{24} = 1\ 000\ 000\ 000\ 000\ 000\ 000\ 000\ 000$

5. Medidas de la información

Nombre	Símbolo	Factor y valor en el el ISO/IEC 80000-13
byte	B	$2^0 = 1$
kibibyte	KiB	$2^{10} = 1\,024$
mebibyte	MiB	$2^{20} = 1\,048\,576$
gibibyte	GiB	$2^{30} = 1\,073\,741\,824$
tebibyte	TiB	$2^{40} = 1\,099\,511\,627\,776$
pebibyte	PiB	$2^{50} = 1\,125\,899\,906\,842\,624$
exbibyte	EiB	$2^{60} = 1\,152\,921\,504\,606\,846\,976$
zebibyte	ZiB	$2^{70} = 1\,180\,591\,620\,717\,411\,303\,424$
yobibyte	YiB	$2^{80} = 1\,208\,925\,819\,614\,629\,174\,706\,176$

5. Medidas de la información

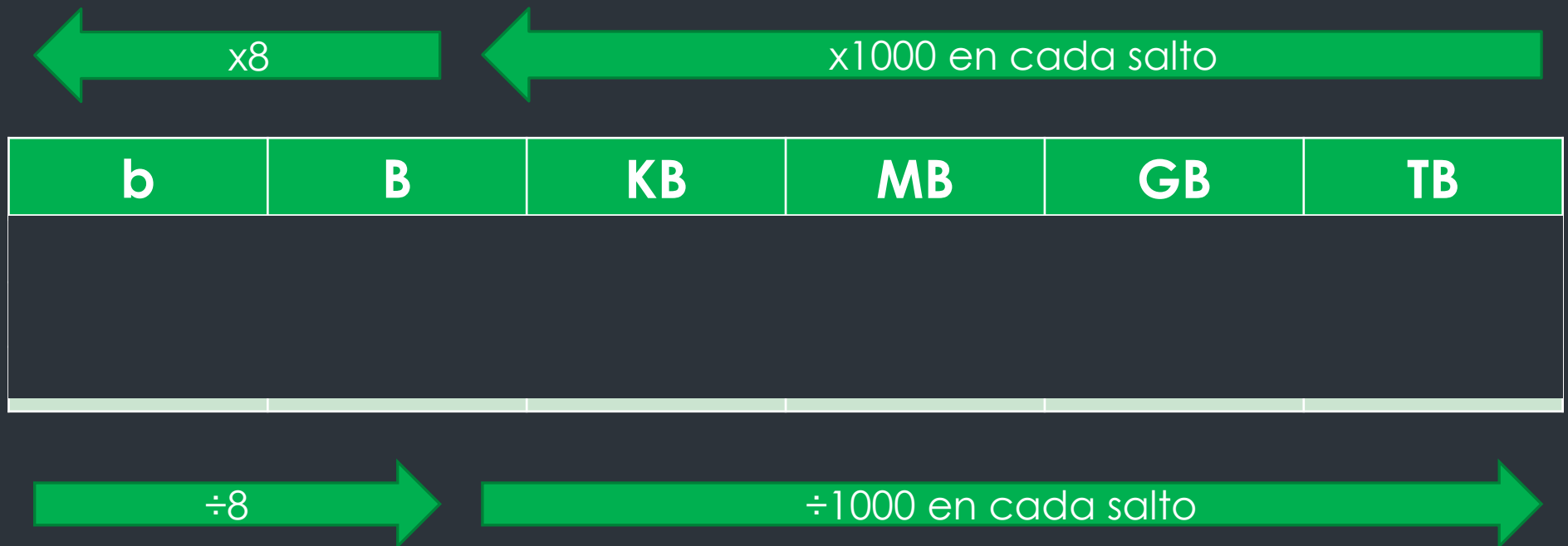
➡ Convertir al resto de unidades: 1 TB, 3.000 MB y 1.000.000 B



b	B	KB	MB	GB	TB
8.000.000.000.000	1.000.000.000.000	1.000.000.000	1.000.000	1.000	1
24.000.000.000	3.000.000.000	3.000.000	3.000	3	0,003
8.000.000	1.000.000	1.000	1	0,001	0,000.001

5. Medidas de la información

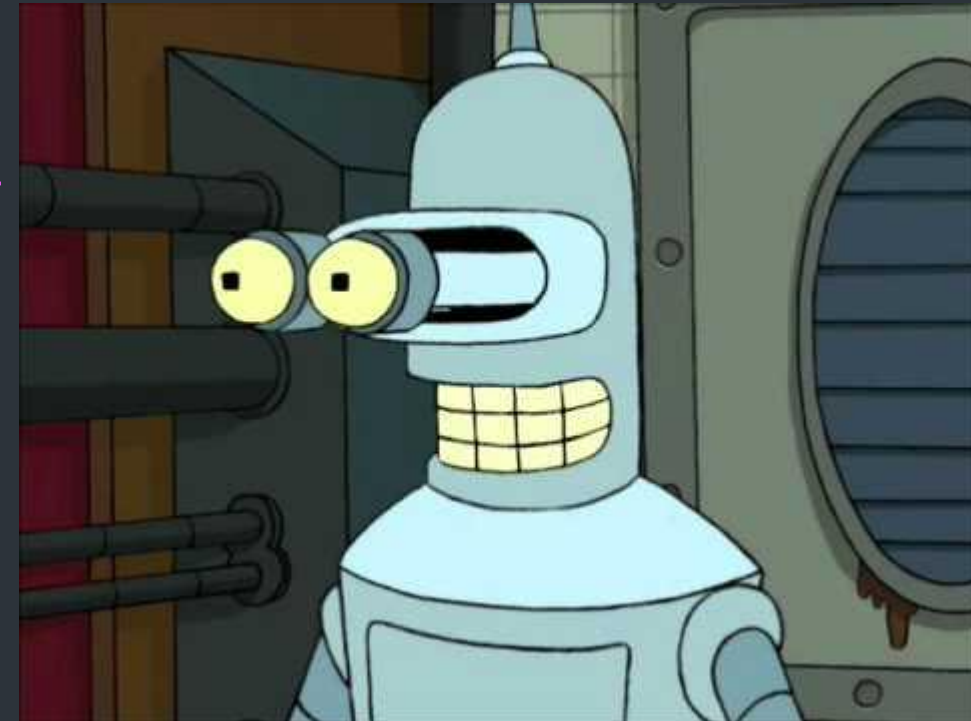
- Convertir al resto de unidades: 7 GB, 6.000.00 MB y 16.000.000 b



6. Ampliación

6. Ampliación

- Nate Gentile
 - Binario y puertas lógicas.
- Electrónica FP
 - Sistemas numéricos
 - Binario a decimal
 - Binario a hexadecim
 - Binario a octal





**GRACIAS POR
VUESTRA ATENCIÓN**

