



Departamento de Lenguajes y Ciencias de la Computación  
Universidad de Málaga

## TEMA 1: El ordenador y la información

### Fundamentos de Informática (Ingeniería Técnica Industrial) E.U. Politécnica



E.U. Politécnica

Autores: M.C. Aranda, A. Fernández, J. Galindo, M. Trella

#### **Índice de contenidos**

1. Definición, elementos y aplicaciones de la Informática
2. Introducción a la codificación
3. Esquema funcional de un ordenador
4. El ordenador central
5. Clasificación de los ordenadores
6. Representación de la información en Sistemas Digitales

## **1. Definición, elementos y aplicaciones de la informática**

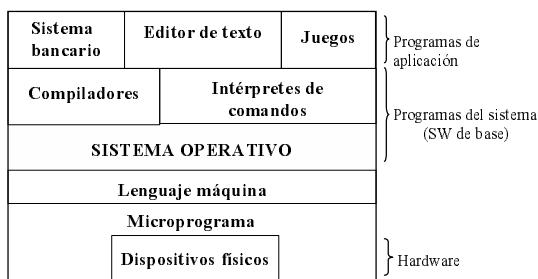
- Informática = INFORmación autoMÁTICA
- Definición:

*Conocimientos científicos + técnicas*  
↓ para  
*Tratamiento automático de la información*  
↓ usando  
*Computadoras electrónicas*
- Elementos básicos:
  - **HARDWARE:** soporte físico => teclado, pantalla, tarjetas, cables...
  - **SOFTWARE:** soporte lógico =>
    - programas: listas de instrucciones que el ordenador puede ejecutar
    - +  
datos sobre los que actúan los programas

3

## **1. Definición, elementos y aplicaciones de la informática**

- Sistemas Operativos (SO)
  - Programa que ayuda al usuario y simplifica el manejo de un ordenador
  - Funciones principales:
    - Interfaz hombre-máquina
    - Administrador de recursos
  - Tipos de SO:
    - DOS (Disk Operating System)
    - OS/2 de IBM
    - Windows 95, 98, NT, 2000, Millenium de Microsoft
    - UNIX
    - LINUX
    - ...



4

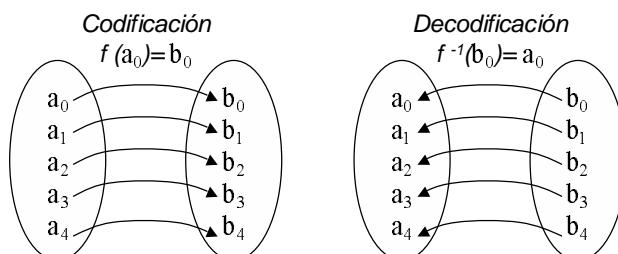
## **1. Definición, elementos y aplicaciones de la informática**

- Aplicaciones de la informática:
  - Investigación
    - Cálculos matemáticos
    - Análisis automático de textos
    - Simulación de sistemas complejos
    - ...
  - Aplicaciones técnicas y sanitarias
    - Diseño de circuitos, edificios... (CAD, CAM)
    - Robótica
    - Medicina
    - ...
  - Gestión
    - Contabilidad
    - Nóminas
    - Facturación
    - ...
  - Otras
    - Procesadores de texto
    - Bases de datos
    - Hojas de cálculo
    - Juegos
    - Enseñanza
    - ...

5

## **2. Introducción a la codificación**

- Codificación
  - Transformación de los elementos de un conjunto en elementos de otro conjunto siguiendo un método determinado, de tal forma que posteriormente se pueda efectuar el proceso inverso de decodificación



6

## 2. Introducción a la codificación

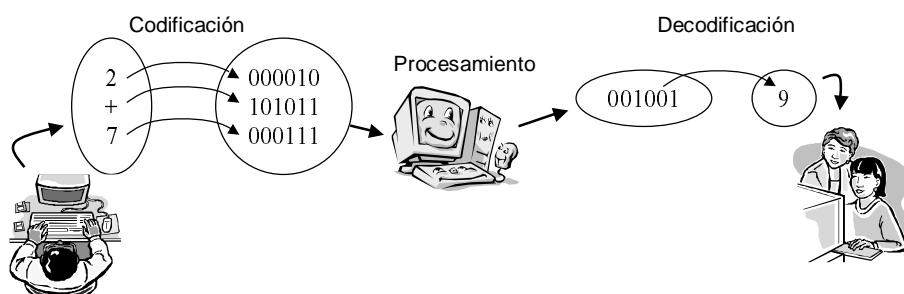
- Los programas y los datos en las computadoras se representan mediante cadenas de 0 y 1 (código binario)
- Los 0 y 1 son formas de diferenciar dos estados. Normalmente se representan físicamente como impulsos eléctricos con una determinada tensión o voltaje, por ejemplo:
  - 3.3 voltios para el 1
  - 0 voltios para el 0
- BIT (BInary digiT)
  - Unidad de información más pequeña que se puede almacenar en un ordenador (0,1)
- Otras unidades de medida de información

Unidad	Equivalencias
1 byte	8 bits
1 KiloByte (KB)	$2^{10}$ bytes = 1.024 bytes
1 MegaByte (MB)	$2^{20}$ bytes = 1.048.576 bytes = $2^{10}$ KB
1 GigaByte (GB)	$2^{30}$ bytes = 1.073.741.824 bytes = $2^{20}$ KB = $2^{10}$ MB
1 TeraByte (TB)	$2^{40}$ bytes = 1.099.511.627.776 bytes = $2^{30}$ KB = $2^{20}$ MB = $2^{10}$ GB

7

## 2. Introducción a la codificación

- Ejemplo de codificación/decodificación en un sistema informático:
  - Entrada (p.ej. un teclado): usuario introduce 2 + 7
  - Salida (p.ej. un monitor): ordenador realiza la suma y devuelve un 9



8

## 2. Introducción a la codificación

- Ejemplo de un código de 2 bits:

Símbolo a codificar	Código binario
0	00
1	01
2	10
3	11

Observa que para codificar 4 símbolos hemos necesitado cadenas de 2 bits

- ¿Cuántos bits serán necesarios para codificar  $m$  símbolos?

$$n \text{ bits, de modo que } 2^n \geq m$$

- Ejemplos:

Nº de símbolos ( $m$ )	Nº de bits ( $n$ )
4	2 ( $2^2 = 4 \geq 4$ )
256	8 ( $2^8 = 256 \geq 256$ )
257	9 ( $2^9 = 512 \geq 257$ )
512	9 ( $2^9 = 512 \geq 512$ )

9

## 2. Introducción a la codificación

- Existen muchos códigos binarios diferentes:
  - AIKEN, BCD, GRAY, ASCII ...
- El código ASCII (*American Standard Code for Information Interchange*) es el más usado para codificar los caracteres empleados habitualmente.
  - Es un conjunto **ordenado** de caracteres 256 símbolos (se emplean por tanto 8 bits para codificar cada símbolo).
  - Cada símbolo se codifica por un número que representa su posición dentro del conjunto.
  - Entre los 256 símbolos del código ASCII podemos encontrar entre otros.
    - Alfabéticos:
      - mayúsculas: (A-Z) => (65-90)
      - minúsculas: (a-z) => (97-122)
    - Numéricos
      - (0-9) => (47-57)
    - De puntuación
    - Especiales ( % & \$ # { } ...)

10

## 2. Introducción a la codificación

Lista de códigos ASCII

0	32	64	96	128	160	192	224
1	33	65	97	129	161	193	225
2	34	66	98	130	162	194	226
3	35	67	99	131	163	195	227
4	36	68	100	132	164	196	228
5	37	69	101	133	165	197	229
6	38	70	102	134	166	198	230
7	39	71	103	135	167	199	231
8	40	72	104	136	168	200	232
9	41	73	105	137	169	201	233
10	42	74	106	138	170	202	234
11	43	75	107	139	171	203	235
12	44	76	108	140	172	204	236
13	45	77	109	141	173	205	237
14	46	78	110	142	174	206	238
15	47	79	111	143	175	207	239
16	48	80	112	144	176	208	240
17	49	81	113	145	177	209	241
18	50	82	114	146	178	210	242
19	51	83	115	147	179	211	243
20	52	84	116	148	180	212	244
21	53	85	117	149	181	213	245
22	54	86	118	150	182	214	246
23	55	87	119	151	183	215	247
24	56	88	120	152	184	216	248
25	57	89	121	153	185	217	249
26	58	90	122	154	186	218	250
27	59	91	123	155	187	219	251
28	60	92	124	156	188	220	252
29	61	93	125	157	189	221	253
30	62	94	126	158	190	222	254
31	63	95	127	159	191	223	255

11

## 3. Esquema funcional de un ordenador

- Ordenador o computadora

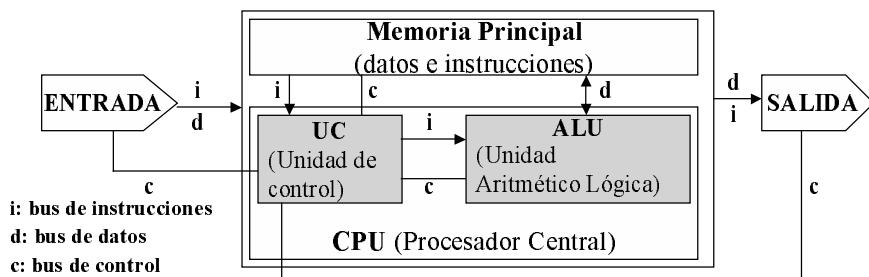
- Máquina capaz de
- |  |
|--|
| 1. Aceptar unos <b>datos de entrada</b><br>2. Efectuar con ellos <b>operaciones lógicas y/o aritméticas (procesamiento)</b><br>3. Proporcionar la información resultante a través de unos <b>datos de salida</b> |
|--|

- Todas estas acciones se realizan bajo el control de un *conjunto ordenado de instrucciones = programa*

12

### 3. Esquema funcional de un ordenador

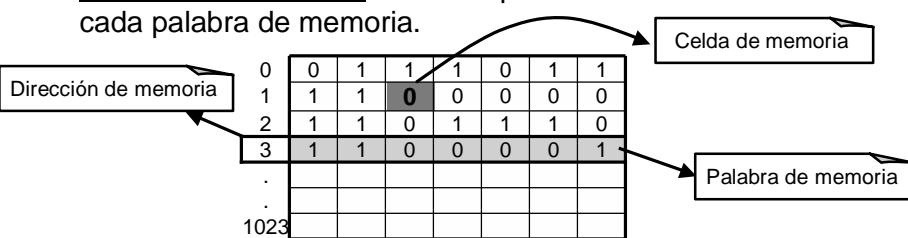
- Arquitectura de Von Neumann



13

### 4. El ordenador central

- Memoria principal
  - Unidad de almacenamiento de *instrucciones* y *datos*
- Formada por circuitos electrónicos integrados capaces de almacenar valores binarios (0 ó 1) en cada celda de memoria
- Palabra de memoria: menor conjunto de celdas de memoria que se pueden leer o escribir simultáneamente.
- Dirección de memoria: número que identifica de forma única cada palabra de memoria.



Ejemplo de memoria de 1024 palabras de 1 byte cada una = memoria de 1KB

14

## 4. El ordenador central

- Tipos de memoria principal
  - **RAM** (*Random Access Memory, memoria de acceso aleatorio*)
  - **ROM** (*Read Only Memory, memoria de sólo lectura*)
- Memoria RAM
  - **Acceso aleatorio o directo**: se puede acceder a ella a cualquier posición.
  - El tiempo de acceso a un dato no depende de su posición.
  - Operaciones que se pueden realizar:
    - **Lectura**, un dato puede ser leído tantas veces como queramos
    - **Escritura**, al escribir un dato se pierde el que hubiera antes en esa misma posición
  - Es una memoria **volátil**: se borra al desconectarle la alimentación.
  - Información que contiene:
    - Código binario de:
      - Programas del usuario en ejecución (editor de texto, juego,...)
      - Sistema Operativo
      - Controladores de dispositivos (ratón, tarjeta de sonido,...)
    - Datos que necesitan los programas codificados en binario

15

## 4. El ordenador central

- Memoria ROM
  - **Acceso aleatorio o directo.**
  - **Sólo** se puede acceder a ella para realizar una operación de **lectura**.
  - Es una memoria **no volátil**: **no** se borra al desconectarle la alimentación.
  - Información que contiene:
    - Programa para la puesta en marcha del ordenador:
      - Cómo cargar el SO
      - Control del teclado
      - Programas de comprobación del *hardware*: comprobación de la memoria instalada, disqueteras, disco duro,...
    - BIOS (*Basic Input Output System, sistema básico de entrada/salida*)
  - Tipos de memoria ROM
    - PROM (*Programable ROM*): una vez grabadas no se pueden borrar.
    - EPROM (*Erasable PROM*): se pueden borrar con rayos ultravioletas y volver a grabar.
    - EEPROM (*Electricaly EPROM*): son EPROM que se borran eléctricamente.

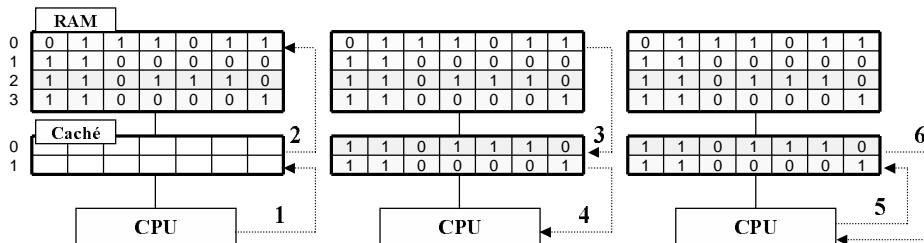
16

#### 4.1 La Memoria Principal

## 4. El ordenador central

- Memoria Caché

- Memoria **cara, rápida y pequeña** se suele ubicar entre la memoria principal y la CPU.
- Acelera los accesos a memoria principal
- Funcionamiento:



1. La CPU necesita una palabra de memoria (la 2) y se la pide a la caché
2. La caché no la tiene y se la solicita a la RAM.
3. La RAM envía a la caché además de la palabra solicitada (la 2), la siguiente (la 3).
4. La caché le envía a la CPU la palabra que pidió (la 2)
5. La CPU pide una nueva palabra a la caché (la 3)
6. En esta ocasión la caché tiene la palabra que se solicita con lo que nos ahorraremos el acceso a la RAM que es bastante más lento.

17

#### 4.2 La CPU

## 4. El ordenador central

- **CPU**, Unidad Central de Proceso = Procesador

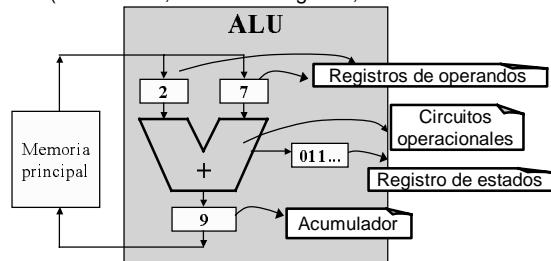
- Elemento que controla y realiza la mayoría de las operaciones que se llevan a cabo en el ordenador.
- Para llevar a cabo estas operaciones la CPU necesita al resto de los componentes del ordenador:
  - Memoria principal
  - Periféricos y dispositivos de entrada/salida
- Componentes de la CPU:
  - **ALU**, Unidad Aritmético Lógica
  - **UC**, Unidad de Control
  - **Registros**
    - Unidades de almacenamiento temporal
    - Normalmente tienen capacidad para almacenar una palabra de memoria
    - Velocidad de lectura/escritura de datos más alta que la memoria principal y la caché

18

#### 4. El ordenador central

- **ALU, Unidad Aritmético Lógica**

- Realiza operaciones
  - Aritméticas: sumas, restas, multiplicaciones,...
  - Lógicas: comparaciones, NOT, AND,...
- Componentes de la ALU
  - **Circuitos operacionales:** circuitos digitales que realizan las operaciones
  - **Registros:** almacenan temporalmente los datos de E/S de los circuitos operacionales
    - **Registro acumulador:** almacena temporalmente el resultado de las operaciones
    - **Registros de operandos**
    - **Registro de estados:** registro especial que indica el estado de la última operación (si ha sido 0, si ha sido negativo, si ha habido desbordamiento,...)



19

#### 4. El ordenador central

- **UC, Unidad de Control**

- Dirige las operaciones más importantes del ordenador, estableciendo la comunicación entre la ALU, la memoria principal y el resto de componentes.
- Controla la ejecución de cada instrucción de un programa
- Componentes de la UC
  - **PC (Program Counter, registro Contador de Programa),** contiene la dirección de memoria de la siguiente instrucción a ejecutar.
  - **RI (Registro de Instrucción),** contiene el código de la instrucción que se está ejecutando.
  - **Decodificador,** circuitos especiales encargados de determinar qué se debe hacer en función de:
    - El código de la instrucción a ejecutar
    - Señales de estado de los dispositivos

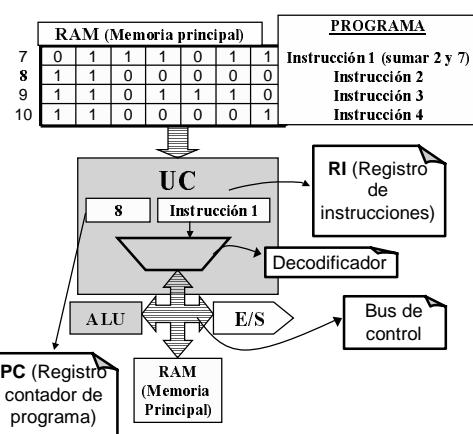
20

## 4. El ordenador central

- **UC, Unidad de Control**

- Fases de ejecución de una instrucción

1. Búsqueda de la instrucción a ejecutar:
  - Carga en el RI la instrucción de memoria principal que se encuentre en la dirección que indique el PC
  - Se modifica el PC para que apunte a la siguiente instrucción
2. Ejecución
  - Se decodifica el código de la instrucción almacenada en el RI
  - Se generan las señales de control necesarias para la ejecución de la instrucción, que se envían a los dispositivos correspondientes a través del bus de control:
    - » ALU
    - » Memoria principal
    - » Canales de entrada / salida



21

## 4. El ordenador central

- **El reloj**

- Circuito encargado de sincronizar el funcionamiento de todos los elementos del ordenador (UC, ALU, memoria,...).
- **Ciclo máquina:** pulsos regulares de tiempo que emite el reloj.
  - Cada instrucción se ejecuta en un tiempo igual a un múltiplo de ciclos máquina.
- **Frecuencia:** velocidad de latido del reloj.
  - Se mide en MegaHerzios (Mhz)=> 1 Mhz = 1.000.000 ciclos máquina / segundo.
  - A mayor frecuencia, mayor velocidad de ejecución de las instrucciones.
  - Si aumentamos mucho la frecuencia del reloj los circuitos se calientan por lo que no se puede acelerar demasiado un procesador. Cada procesador está diseñado para trabajar a una velocidad máxima.

22

## 4. El ordenador central

- **Los buses**

- Conjuntos de cables (hilos, pistas) encargados de que comunican todos los módulos del ordenador entre sí.
- Por cada hilo o pista se puede transmitir un bit
- Tipos de transmisiones:
  - Serie, transmitir varios bits por el mismo hilo uno detrás de otro
  - Paralelo, transmitir conjuntos de bits a la vez uno por cada hilo.
- Tipos de buses:
  - Según el tipo de información que circula por ellos
    - **Bus de datos**, transmite instrucciones y datos.
    - **Bus de direcciones**, transmite direcciones de memoria.
      - » Con un bus de  $n$  hilos podemos direccionar  $2^n$  posiciones de memoria
    - **Bus de control**, transmite:
      - » señales para controlar y sincronizar todos los componentes del ordenador
      - » señales que indican el estado de los componentes
  - Según los elementos que conectan
    - **Bus local**, establece un camino entre la CPU y la memoria. Los datos se transfieren a la máxima velocidad que permite el procesador.
    - **Bus de expansión o del sistema**, establece un camino entre la CPU y el resto de componentes. Es más lento que el bus local

23

## 5. Clasificación de los ordenadores

- Mainframes (supercomputadoras)
  - Máquinas de gran:
    - Capacidad de proceso
    - Velocidad
    - Fiabilidad
    - Tamaño
    - Precio
  - Multiusuario, puede haber cientos de usuarios trabajando a la vez con el mismo ordenador.
  - Se usan para gestión de grandes empresas.
  - Ejemplos: IBM 3090S, UNISYS 2200
- Miniordenadores (superminicomputadoras)
  - Menor potencia, velocidad, tamaño, etc. que los mainframes.
  - Multiusuario, el número de posibles usuarios conectados a la vez suele ser menor de 150
  - Se usan para gestión de empresas medianas, de pequeño volumen de negocios
  - Ejemplos: VAX-8350, AS/400, MV-10000
- Microordenadores o PC's (*Personal Computer, Ordenador Personal*)
  - Ordenadores de
    - pequeño tamaño y precio
    - alta potencia y fiabilidad
  - Monousuario
  - Multitarea, se pueden ejecutar varios programas a la vez

24

## 6. Ergonomía y ecología informática

- **Ergonomía**, ciencia que estudia la comodidad y la salud en el trabajo con un ordenador.
- **Ecología**, ciencia que estudia cómo aprovechar mejor los recursos de forma que un ordenador contamine y consuma lo menos posible.
- ....

25

## 6. Representación de la información en sistemas digitales

### 6.1. Sistemas de numeración usuales en informática

- Sistema de numeración en base **b**
    - Utiliza para representar los números de un alfabeto compuesto por **b** símbolos
    - Cada símbolo que compone un número contribuye con un valor que depende de:
      - la cifra en sí
      - la posición dentro del número
- Base:  $b$   
Conjunto de símbolos:  $\{n_i / i \in \mathbb{Z}\}$   
Número:  $\dots n_4 n_3 n_2 n_1 n_0 n_{n-1} n_{n-2} \dots$   
Valor:  $\dots + n_4 b^4 + n_3 b^3 + n_2 b^2 + n_1 b^1 + n_0 b^0 + n_{n-1} b^{n-1} + n_{n-2} b^{n-2} + \dots$

Ejemplo:

Base:	10
Conjunto de símbolos:	$\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
Número:	5678.89
Valor:	$5678.89 = 5 \cdot 10^3 + 6 \cdot 10^2 + 7 \cdot 10^1 + 8 \cdot 10^0 + 8 \cdot 10^{-1} + 9 \cdot 10^{-2}$

26

## 6. Representación de la información en sistemas digitales

### 6.1. Sistemas de numeración usuales en informática

- Sistema de numeración en base 2 o **binario**

- Sistema empleado por los ordenadores

Base: **2**

Conjunto de símbolos: **{0,1}**

- Conversión de **binario a decimal**, sumar los pesos ( $2^n$ ) de las posiciones en las que hay un 1.

Número:  $01110_2$

$$\text{Valor: } 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 14_{10}$$

- Conversión de **decimal a binario**, métodos de las divisiones y multiplicaciones.

Número:  $34.1875_{10}$

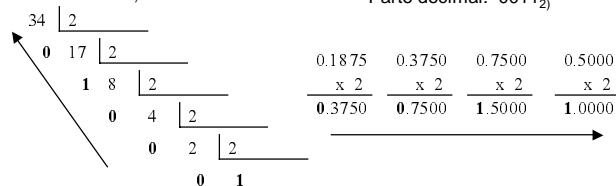
Valor:  $100010.0011_2$

Parte entera:  $100010_2$

Parte decimal:  $0011_2$

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

Números  
binarios con 3  
bits



27

## 6. Representación de la información en sistemas digitales

### 6.1. Sistemas de numeración usuales en informática

- Sistema de numeración en base 8 u **octal**

Base: **8**

Conjunto de símbolos: **{0,1,2,3,4,5,6,7}**

- Conversión de **octal a decimal**

Número:  $432_8$

$$\text{Valor: } 4 \cdot 8^2 + 3 \cdot 8^1 + 2 \cdot 8^0 = 282_{10}$$

- Conversión de **decimal a octal**, métodos de las divisiones y multiplicaciones con divisor y multiplicador 8

- Conversión de **binario a octal**, agrupar cifras de 3 en 3 y transformar mediante tabla de conversión

Número:  $0111011101.0110_2$

$$\text{Valor: } 000.111 \cdot 011.101 \cdot 011.000_2 = 735.3_8$$

- Conversión de **octal a binario**, utilizar tabla de conversión para cada cifra

Número:  $3245.765_8$

$$\text{Valor: } 11010100101.111110101_2$$

28

## 6. Representación de la información en sistemas digitales

### 6.1. Sistemas de numeración usuales en informática

- Sistema de numeración en base **16** o **hexadecimal**

Base: **16**

Conjunto de símbolos: **{0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F}**

- Conversión de *hexadecimal* a *decimal*, sumar los pesos ( $2^n$ ) de las posiciones en las que hay un 1.

Número:  $857A_{16}$

Valor:  $8 \cdot 16^3 + 5 \cdot 16^2 + 7 \cdot 16^1 + A \cdot 16^0 = 34170_{10}$

- Conversión de *decimal* a *hexadecimal*, métodos de las divisiones y multiplicaciones con divisor y multiplicador 16

- Conversión de binario a octal, agrupar cifras de 3 en 3 y transformar mediante tabla de conversión

Número:  $0111011101.0110_{16}$

Valor:  $0001.1101.1101 \cdot 0110_{2} = 1DD.6_{16}$

- Conversión de octal a binario, utilizar tabla de conversión para cada cifra

Número:  $6A5.1F5_{16}$

Valor:  $11010100101.111110101_{2}$

Decimal	Hexadecimal	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
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Cifras hexadecimales en decimal y binario

29

## 6. Representación de la información en sistemas digitales

### 6.2. Operaciones aritméticas y booleanas

- Operaciones aritméticas

Suma	
$0 + 0$	0
$0 + 1$	1
$1 + 0$	1
$1 + 1$	0 (y me llevo 1)

Resta	
$0 - 0$	0
$0 - 1$	1 (y debo 1)
$1 - 0$	1
$1 - 1$	0

Producto	
$0 \times 0$	0
$0 \times 1$	0
$1 \times 0$	0
$1 \times 1$	1

División	
$0 : 0$	(indeterminado)
$0 : 1$	0
$1 : 0$	(infinito)
$1 : 1$	1

30

## 6. Representación de la información en sistemas digitales

### 6.2. Operaciones aritméticas y booleanas

- Operaciones booleanas

OR	
0 OR 0	0
0 OR 1	1
1 OR 0	1
1 OR 1	1

AND	
0 AND 0	0
0 AND 1	0
1 AND 0	0
1 AND 1	1

NOT	
NOT 0	1
NOT 1	0

NOR (NOT OR)	
0 NOR 0	1
0 NOR 1	0
1 NOR 0	0
1 NOR 1	0

NAND (NOT AND)	
0 NAND 0	1
0 NAND 1	1
1 NAND 0	1
1 NAND 1	0

XOR (eXclusive OR)	
0 XOR 0	0
0 XOR 1	1
1 XOR 0	1
1 XOR 1	0

31

## Bibliografía

- Bibliografía principal

- A. Prieto, A. Lloris y J.C. Torres. "Introducción a la Informática". (1<sup>a</sup> ó 2<sup>a</sup> Edición). McGraw-Hill, 1995.
- Galindo, Sánchez, Yáñez, Escolano, Del Jesus, Aguilera, Rodríguez, Sánchez y Argudo. "Fundamentos Informáticos". Servicio de Publicaciones de la Universidad de Cádiz, 1996.
- Tremblay, Bunt. "Introducción a la Ciencia de las Computadoras. Enfoque algorítmico". McGraw-Hill, 1988.
- Peter Norton. "Introducción a la Computación". McGraw-Hill, 1995.

- Bibliografía adicional

- Roger S. Walker. "Informática Básica". Anaya Multimedia.
- Peter Bishop. "Conceptos de Informática". Anaya Multimedia.
- Ll. Guilera Agüera. "Introducción a la Informática". Edunsa.
- R. Peña Marí. "Diseño de Programas: Formalismo y Abstracción". Prentice Hall, 1998.

32