


# UT01 – INTRODUCCIÓN A LOS SISTEMAS INFORMÁTICOS



A diagram illustrating a monolithic operating system architecture. It features a central circle with the text "SISTEMAS OPERATIVOS MONOPUESTO". This central circle is surrounded by a thick, dark brown ring. To the left of this ring, there is a large, light gray arc that overlaps the ring. Further to the left, there is a smaller, light beige arc. A horizontal white line extends from the right side of the central circle, passing through the dark brown ring and ending in a small white rectangle on the right edge of the diagram.

SISTEMAS  
OPERATIVOS  
MONOPUESTO

SISTEMAS MICROINFORMÁTICOS Y REDES

## Tabla de Contenido

<b>1. El sistema informático.....</b>	<b>3</b>
<b>2. Hardware.....</b>	<b>3</b>
<b>3. Software .....</b>	<b>5</b>
<b>3.1. Tipos de Software.....</b>	<b>6</b>
<b>3.2. Firmware .....</b>	<b>7</b>
<b>4. Componentes humanos .....</b>	<b>7</b>
<b>5. Funcionamiento de un ordenador. ....</b>	<b>7</b>
<b>6. Clasificación de los sistemas informáticos .....</b>	<b>8</b>
<b>7. Representación de la información .....</b>	<b>10</b>
<b>7.1. Sistemas de numeración .....</b>	<b>11</b>
<b>7.1.1 Conversión Decimal ⇔ Binario.....</b>	<b>12</b>
<b>7.1.2 Conversión de Decimal ⇔ Octal.....</b>	<b>14</b>
<b>7.1.3 Conversión Decimal ⇔ Hexadecimal .....</b>	<b>15</b>
<b>7.1.4 Conversión Binario ⇔ Octal .....</b>	<b>16</b>
<b>7.1.5 Conversión Binario ⇔ Hexadecimal .....</b>	<b>17</b>
<b>7.1.6 Conversión Octal ⇔ Hexadecimal .....</b>	<b>18</b>
<b>7.2. Sistemas de codificación alfanumérica .....</b>	<b>19</b>
<b>7.2.1 Código ASCII (American Standard Code for Information Interchange) ....</b>	<b>19</b>
<b>7.2.2 Código UNICODE (Universal Code) .....</b>	<b>19</b>
<b>8. Bibliografía de referencia.....</b>	<b>20</b>

# UT01 – INTRODUCCIÓN A LOS SISTEMAS INFORMÁTICOS

## 1. El sistema informático

Un sistema informático está formado por el conjunto de elementos necesarios para la realización y explotación de aplicaciones informáticas. Un sistema informático típico emplea un ordenador que usa dispositivos para capturar, almacenar y procesar los datos. Dicho ordenador, junto con la persona que lo maneja y los periféricos que lo envuelven, resultan de por sí un ejemplo de sistema informático.

Los componentes que forman un sistema informático son:

- **Componentes físicos o hardware:** son todos los elementos electrónicos y mecánicos que forman parte del ordenador como el teclado, ratón, pantalla, disco duro, etc.
- **Componentes lógicos o software:** son la parte intangible del ordenador: el sistema operativo, los programas y los datos.
- **Componentes humanos:** son las personas que participan en la dirección, diseño, desarrollo, implantación y explotación de un sistema informático. Es el conjunto formado por el personal informático y los usuarios.



## 2. Hardware

El hardware de un sistema informático es la parte física o tangible. Está formada por todos los elementos electrónicos y mecánicos como el microprocesador, el disco duro, el teclado etc.

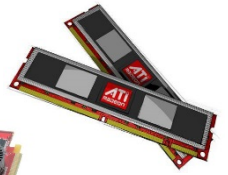
En un ordenador podemos encontrar los siguientes componentes físicos:

- **La caja** o carcasa del ordenador es el componente sobre el que se montará el resto de los dispositivos del ordenador, sirve de soporte y protección para los dispositivos instalados dentro de ella. Existen cajas de diferentes formas, tamaños, materiales, estilos y colores.
- **La placa base** es uno de los elementos principales de ordenador ya que a ella se conectan todos los demás dispositivos (disco duro, procesador, memoria,...) permitiendo que se comuniquen entre ellos. De la placa base dependerá el tipo y número de componentes que podamos instalar.
- **El procesador** es componente principal de ordenador. Se encarga de ejecutar las instrucciones de los programas que se están ejecutando en un ordenador así como dirigir y controlar al resto de componentes.



- El **disipador-ventilador** del procesador se encarga de que éste no se sobrecaliente.
- La **memoria RAM** se encarga de almacenar los programas y archivos mientras se están utilizando para que podamos verlos o ejecutarlos. Es una memoria volátil, desaparece cuando se corta el suministro eléctrico.
- La **tarjeta de vídeo** permite reproducir en pantalla lo que estamos haciendo en el ordenador. A ella conectamos la pantalla del ordenador.
- El **disco duro** constituye el medio de almacenamiento de información más importante del ordenador ya que permite almacenar y recuperar gran cantidad de información. En el disco duro es donde se suele instalar el sistema operativo y los programas, y donde se puede almacenar todo tipo de archivos (documentos, vídeos, música, fotos, etc.).
- Las **unidades ópticas** son aquellas que nos permiten grabar o reproducir discos de CD, DVD o Blu-ray.
- **Otras tarjetas de expansión:** entre ellas podemos destacar las tarjetas de red (que permiten conectarnos a una red interna o a Internet), tarjetas de televisión (para poder ver la televisión), tarjetas de sonido (para reproducir audio del ordenador) etc.
- La **fuentes de alimentación** permite suministrar energía a la placa base y al resto de componentes del ordenador como el disco duro, unidades ópticas etc.
- Los **periféricos** son dispositivos que se conectan al ordenador permitiendo almacenar información y comunicar al ordenador con el mundo exterior. Se pueden clasificar en:
  - Periféricos de entrada: introducen datos externos al ordenador para su posterior tratamiento. Algunos de los periféricos de entrada que podemos encontrar son el teclado, el ratón, la webcam, el micrófono o el escáner.
  - Periféricos de salida: reciben la información del ordenador y la reproducen para que sea perceptible por el usuario. Ejemplos: monitor, impresora, altavoces, auriculares, etc. También existen periféricos que son tanto de entrada como de salida como por ejemplo una pantalla táctil.
  - Periféricos de almacenamiento: se encargan de guardar datos. Pueden ser internos (se encuentran dentro de la carcasa del pc) o externos. Entre ellos podemos encontrar los discos duros, unidades de DVD, memorias USB, etc.
  - Periféricos de comunicación: facilitan la interacción entre dos o más ordenadores o entre un ordenador y otro periférico externo. Entre ellos se encuentra las tarjetas de red, tarjetas bluetooth, controladores de puertos etc.

Muchos periféricos necesitan de un software especial que permita al sistema operativo reconocer al periférico y utilizarlo de forma correcta, estos programas se denominan **drivers**.



### 3. Software

La **parte lógica o software** está formada por todos los elementos no físicos. Son fundamentalmente los programas y los datos (sistema operativo, aplicaciones, drivers etc.)

Un **programa** es un conjunto de instrucciones que una vez ejecutadas realizarán una o varias tareas en una computadora. Ese conjunto de instrucciones estará escrito en un determinado lenguaje de programación.

Un **lenguaje de programación** es un conjunto de símbolos combinados siguiendo una sintaxis determinada, que permite la introducción de instrucciones en el ordenador.

#### Ejemplo (Javascript)

```
if (edad >= 18) {  
  
    alert("Eres mayor de edad");  
  
} else {  
  
    alert("Eres menor de edad");  
  
}
```

Existen diferentes **tipos de lenguajes de programación**:

- **Lenguajes de bajo nivel.** Son los más cercanos al hardware del ordenador. A su vez se pueden clasificar en:
  - **Lenguaje máquina.** Es el único que es comprensible directamente por el ordenador. Depende del sistema que usemos ya que cada tipo de procesador tiene un lenguaje máquina diferente. El resto de programas hechos en otro lenguaje se tendrán que traducir a lenguaje máquina para que el ordenador pueda entenderlo.
  - **Lenguaje ensamblador.** Cada instrucción en lenguaje ensamblador es equivalente a una instrucción del lenguaje máquina. La ventaja es que se traduce de forma rápida al lenguaje máquina, pero sus instrucciones varían dependiendo del microprocesador.
- **Lenguajes de alto nivel.** Son lenguajes que se pueden ejecutar en diferentes máquinas ya que son independientes del hardware (sus instrucciones no varían en función del microprocesador). Para que el ordenador lo entienda deben ser traducidos a lenguaje máquina. Este proceso se puede realizar de dos maneras:
  - Mediante intérpretes.
  - Mediante compiladores.

Ejemplos de lenguajes de alto nivel son: C++, Java, C#, Visual Basic, etc.

### 3.1. Tipos de Software

Según la función que realizan, podemos clasificar los distintos tipos de software en:

- **Software de base**

Aquella parte del software sin la cual el ordenador no puede funcionar. Permiten al usuario utilizar el PC sin que deban conocer las características hardware del equipo o cómo funciona internamente. Ejemplos de software de base son **el sistema operativo y los drivers**.



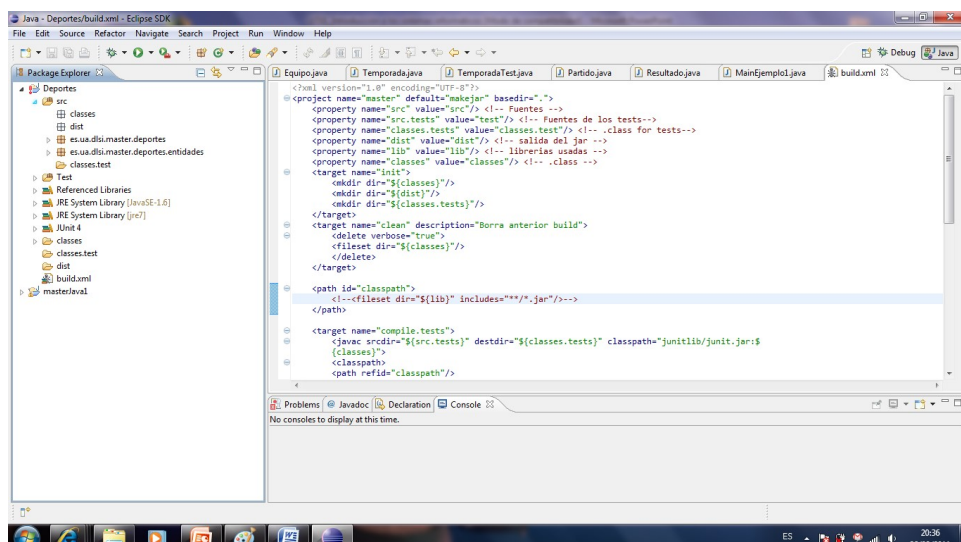
- **Software de aplicación**

Son los programas o aplicaciones que se utilizan para tareas específicas. Por ejemplo: aplicaciones ofimáticas, antivirus, juegos, aplicaciones empresariales, etc. Hay que tener en cuenta que un software de aplicación no se podrá ejecutar sin un software de base como Windows o Linux.



- **Software de programación**

Son programas que proveen a los programadores de herramientas y utilidades para poder desarrollar programas que después serán utilizados por usuarios. Se incluyen compiladores, intérpretes, entornos de desarrollo, etc. Ejemplos: Microsoft Visual Studio, Eclipse, NetBeans, etc.



### 3.2. Firmware

Es un conjunto de instrucciones grabadas normalmente en una memoria tipo ROM (ROM, EPROM, EEPROM, flash, etc.) y establece la lógica de más bajo nivel que controla los circuitos electrónicos de un dispositivo.

Se considera parte del hardware por estar integrado en la electrónica del dispositivo, pero también es software, pues proporciona la lógica y está programado por algún tipo de lenguaje de programación. El firmware recibe órdenes externas y responde operando el dispositivo. En resumen, un firmware es el software que maneja al hardware.

El programa BIOS de una computadora es un firmware cuyo propósito es activar una máquina desde su encendido y preparar el entorno para cargar un sistema operativo en la memoria RAM.

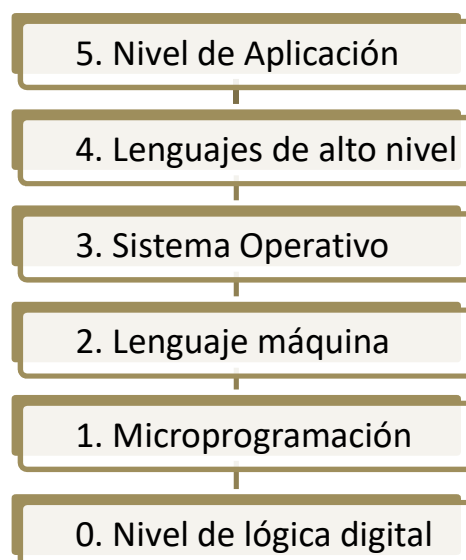
El firmware no es fácilmente modificable, una vez que se graba en un componente hardware queda prácticamente invariable a lo largo de la vida del ordenador.

## 4. Componentes humanos

El último componente del sistema informático está constituido por las personas que participan en la dirección, diseño, desarrollo, implantación y explotación de un sistema informático, es decir, los usuarios y el personal informático.

## 5. Funcionamiento de un ordenador.

Para entender el funcionamiento de un ordenador podemos establecer una visión global de su estructura de forma jerárquica, o por niveles relacionados entre sí. Estos niveles de funcionamiento son los siguientes:





- **Nivel 5. Nivel de aplicación.** Nivel en el cuál el usuario utiliza las aplicaciones para interactuar con el sistema informático
- **Nivel 4. Lenguajes de alto nivel.** Permiten crear programas independientes de la máquina.
- **Nivel 3. Sistema operativo.** Software de base que permite la interacción con el hardware.
- **Nivel 2. Lenguaje máquina y lenguaje ensamblador.** Lenguajes de más bajo nivel que puede utilizar un usuario y son dependientes de la máquina.
- **Nivel 1. Microprogramación.** Juego de instrucciones con las que es capaz de trabajar un microprocesador. A estas microinstrucciones se les llama firmware del procesador.
- **Nivel 0. Lógica digital.** Hardware del ordenador.

## 6. Clasificación de los sistemas informáticos

Los sistemas informáticos se pueden clasificar atendiendo a su uso o a sus prestaciones.

Por su uso podemos clasificarlos en:

1. **Sistemas informáticos de uso general:** se utilizan para varios tipos de aplicaciones, como es el caso de los ordenadores personales.
2. **Sistemas informáticos de uso específico:** se caracterizan por ejecutar uno o unos pocos programas como es el caso de los robots industriales.

Por sus prestaciones podemos hacer una **clasificación de los sistemas informáticos** en:

### 1. Superordenadores

Son ordenadores no comerciales, que son creados específicamente para grandes organizaciones nacionales o internacionales con el fin de realizar tareas específicas avanzadas relacionadas con los campos científico, militar o tecnológico (ej: investigación del genoma humano, predicción de desastres naturales...) Están compuestos por cientos de estaciones y ocupan grandes salas que requieren de una refrigeración industrial.



### 2. Macroordenadores (Mainframes)

Es un computador grande, potente y bastante caro que es utilizado principalmente por empresas para el procesamiento de una cantidad de datos. Por ejemplo: el procesamiento de las transacciones bancarias.





## Diferencias entre Superordenadores y Macroordenadores

Las diferencias principales entre superordenadores y macroordenadores son las siguientes:

- **Cantidad de procesadores.** Los superordenadores suelen tener miles de procesadores mientras que los mainframes o macroordenadores unas decenas como mucho.
- **Dificultad de programación.** Debido a que los superordenadores tienen tantos procesadores trabajando en paralelo la programación es mucho más complicada.
- **Optimización en áreas diferentes.** Los superordenadores están optimizados para cálculos complicados que se realizan en la RAM mientras que los mainframes son optimizados para cálculos simples pero que implican grandes cantidades de datos.
- **Sectores diferentes.** Los superordenadores suelen dedicarse a la ciencia y al ejército mientras que los mainframes o macroordenadores a empresas y aplicaciones administrativas del gobierno.
- **Tamaño.** Los superordenadores requieren de grandes salas de muchos metros cuadrados y de refrigeración industrial mientras que los mainframes ocupan muy pocos metros cuadrados.

### 3. Servidores y estaciones de trabajo (Workstation)

Son ordenadores que pueden ofrecer servicios a otros ordenadores dentro de una red como conexión a Internet, acceso a periféricos, etc. El tamaño de los servidores puede ser más grande o más grande o más pequeño, dependiendo de los servicios ofrecidos y del tamaño de la red. Las estaciones de trabajo, aunque tienen más prestaciones que los ordenadores personales, no suelen ser mucho más grandes que éstos.



### 4. Ordenadores personales o PC

Surgen tras la aparición de los microprocesadores. Se utilizan tanto en el ámbito doméstico como en el profesional. A su vez pueden ser de sobremesa o portátiles.



## 7. Representación de la información

Las personas utilizamos información en forma de números, letras, imágenes o sonidos, pero la información representada de esta forma no es entendible por una máquina. Un circuito electrónico sólo reconoce el paso de corriente o la ausencia de esta por lo que se dice que la máquina sólo reconoce dos estados, apagado o encendido. A estos estados se les asocia los símbolos 0 y 1, llamados bits (binary digit). Por lo que para que una máquina sea capaz de almacenar o interpretar la información debemos traducirla a esos símbolos.

Sin embargo, cuando se almacena la información no se trabaja a nivel de bit ya que sería muy engorroso, por ello se trabaja con grupos de bits múltiplos de dos:

1bit	Un 0 ó un 1
1 byte	8 bits.
1 Kilobyte (KB)	1024 bytes
1 Megabyte (MB)	1024 KB
1 Gigabyte (GB)	1024 MB
1 Terabyte (TB)	1024 GB
1 Petabyte (PB)	1024 TB
1 Exabyte (EB)	1024 PB

En informática se utilizan las potencias de dos para representar las medidas de información, sin embargo se ha extendido el uso de potencias de 10 del sistema decimal ya que es el que se utiliza en el sistema Internacional de Unidades. Esto quiere decir que en primer lugar se utilizó el término Kilobyte para indicar 1000 bytes ya que se asociaba al termino Kilo del sistema decimal (kilogramo=1000g, kilómetro=1000m), cuando debería ser 1024 bytes.

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)

## 7.1. Sistemas de numeración

Un sistema de numeración es un sistema que permite representar cualquier cantidad numérica. Está formado por un conjunto de símbolos y unas reglas que permiten representar esos datos numéricos. La mayoría de estos sistemas se caracterizan por:

- Utilizar un conjunto de símbolos, llamados **dígitos**, para representar el número.
- Tienen asociada una **base**, que es el número diferente de símbolos que podemos utilizar en ese sistema de numeración. Por ejemplo el sistema decimal es un sistema en base 10, los dígitos utilizados son: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
- Son sistemas **posicionales**, el valor de cada dígito dependerá de la posición en la que se encuentre.

En nuestra vida cotidiana solemos usar el sistema decimal, sin embargo, los sistemas de codificación que utiliza normalmente un sistema informático son:

- ✓ **Sistema binario.** Sistema posicional que utiliza sólo dos símbolos: 0 y 1, por lo que su base es 2. A los valores 0 y 1 ya hemos comentado que se les denomina bit (binarydigit), siendo la unidad mínima de información capaz de ser procesada por un ordenador.
- ✓ **Sistema octal.** Utiliza los símbolos del 0 al 7, por lo que su base es 8. Es un sistema posicional.
- ✓ **Sistema hexadecimal.** Utiliza los símbolos del 0 al 9 y las letras A, B, C, D, E y F, por lo que su base es 16. Al igual que los anteriores, es un sistema posicional.

### 7.1.1 Conversión Decimal ↔ Binario



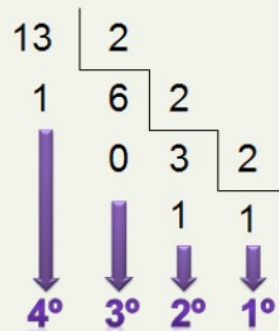
#### Conversión de Decimal a Binario



#### Parte entera del número

##### :: Modo 1

Dividir el número sucesivamente por 2 hasta que el cociente sea menor que 2. El número binario estará formado por el último cociente y los restos desde abajo hacia arriba.



$$13_{10} = 1101_2$$

##### :: Modo 2

¿Qué  $2^x$  es mayor que el número a convertir y es más cercano a él?

Ejemplo:  $13 \Rightarrow 2^4 = 16$

La potencia seleccionada nos indicará el número de bits que necesitamos. En el ejemplo serían 4 bits.

$2^3$	$2^2$	$2^1$	$2^0$
1	1	0	1

$$8 + 4 + 0 + 1 = 13_{10}$$

#### Parte decimal del número

Multiplicamos la parte fraccionaria sucesivamente por 2 hasta que la parte fraccionaria sea 0 o tengamos suficientes dígitos decimales. La parte entera de cada multiplicación formará los bits del número binario.

**Ejemplo:**  $13,125_{10} = 1101,001_2$

Parte entera:  $13 \Rightarrow 1101$

Parte fraccionaria :  $0,125 \Rightarrow 001$

$$\begin{array}{l}
 0,125 \times 2 = \underline{0},250 \\
 0,250 \times 2 = \underline{0},5 \\
 0,5 \times 2 = \underline{1},0
 \end{array}$$

↓



## Conversión de Binario a Decimal



Se aplica el teorema fundamental de la numeración. El número decimal será la suma de los productos de los resultados de multiplicar cada dígito por la base (en este caso 2) elevado a la posición que ocupa.

$$1101,01_2 = (1 \cdot 2^3) + (1 \cdot 2^2) + (0 \cdot 2^1) + (1 \cdot 2^0) + (0 \cdot 2^{-1}) + (1 \cdot 2^{-2}) =$$

$$1 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 + 0 \cdot 1/2^1 + 1 \cdot 1/2^2 =$$

$$8 + 4 + 0 + 1 + 0 + 0,25 = 13,25_{10}$$

### 7.1.2 Conversión de Decimal ⇔ Octal



#### Conversión de Decimal a Octal



La **parte entera** se obtendrá dividiendo el número sucesivamente por 8 hasta que el cociente sea menor que 8. La parte entera del número octal estará formado por el último cociente y los restos desde abajo hacia arriba.

La **parte fraccionaria** se obtendrá multiplicando la parte fraccionaria del número decimal sucesivamente por 8 hasta que la parte fraccionaria sea 0 o tengamos suficientes dígitos decimales. La parte entera de cada multiplicación formará los bits del número octal.

Parte entera	Parte fraccionaria
$  \begin{array}{r}  215 \div 8 = 26 \text{ resto } 7 \\  26 \div 8 = 3 \text{ resto } 2 \\  3 \div 8 = 0 \text{ resto } 3  \end{array}  $ <p style="text-align: center;"> <math>\downarrow \quad \downarrow \quad \downarrow</math>  <math>3^\circ \quad 2^\circ \quad 1^\circ</math> </p>	$  \begin{array}{l}  0,12 \times 8 = 0,96 \\  0,96 \times 8 = 7,68 \\  0,68 \times 8 = 5,44 \\  0,44 \times 8 = 3,52  \end{array}  $ <p style="text-align: center;"> <math>\downarrow</math> </p>

$$215,12_{10} = 327,0753_8$$



#### Conversión de Octal a Decimal



Se aplica el teorema fundamental de la numeración. El número decimal será la suma de los productos de los resultados de multiplicar cada dígito por la base (en este caso 8) elevado a la posición que ocupa.

$$327,0753_8 = (3 \cdot 8^2) + (2 \cdot 8^1) + (7 \cdot 8^0) + (0 \cdot 8^{-1}) + (7 \cdot 8^{-2}) + (5 \cdot 8^{-3}) + (3 \cdot 8^{-4}) =$$

$$3 \cdot 64 + 2 \cdot 8 + 7 \cdot 1 + 0 \cdot 1/8^1 + 7 \cdot 1/8^2 + 5 \cdot 1/8^3 + 3 \cdot 1/8^4 =$$

$$192 + 16 + 7 + 0 + 0,109 + 0,009 + 0,002 = 215,12_{10}$$



### 7.1.3 Conversión Decimal ⇌ Hexadecimal



#### Conversión de Decimal a Hexadecimal

La **parte entera** se obtendrá dividiendo el número sucesivamente por 16 hasta que el cociente sea menor que 16. La parte entera del número hexadecimal estará formado por el último cociente y los restos desde abajo hacia arriba. Si alguno de los restos o el cociente es un número entre 10 y 15 se sustituirá por la letra correspondiente según la tabla siguiente:

Símbolo	Valor
A	10
B	11
C	12
D	13
E	14
F	15

La **parte fraccionaria** se obtendrá multiplicando la parte fraccionaria del número decimal sucesivamente por 16 hasta que la parte fraccionaria sea 0 o tengamos suficientes dígitos decimales. La parte entera de cada multiplicación formará los dígitos del número hexadecimal. Debemos tener en cuenta que si al multiplicar por 16 nos da un número entero entre 10 y 15 lo sustituiremos por su letra correspondiente.

Parte entera	Parte fraccionaria
$  \begin{array}{r}  574 \div 16 = 35 \text{ resto } 14 \rightarrow E \\  35 \div 16 = 2 \text{ resto } 3 \\  2 \div 16 = 0 \text{ resto } 2  \end{array}  $ <p>↓ ↓ ↓</p> <p><b>3° 2° 1°</b></p>	$  \begin{array}{l}  0,12 \times 16 = 1,92 \rightarrow 1 \\  0,92 \times 16 = 14,72 \rightarrow E \\  0,72 \times 16 = 11,52 \rightarrow B \\  0,52 \times 16 = 8,32 \rightarrow 8  \end{array}  $

$$574,12_{10} = 23E,1EB8_{16}$$



#### Conversión de Hexadecimal a Decimal

Se aplica el teorema fundamental de la numeración. El número decimal será la suma de los productos de los resultados de multiplicar cada dígito por la base (en este caso 16) elevado a la posición que ocupa.

$$\begin{aligned}
 23E,1EB8_{16} &= (2 \cdot 16^2) + (3 \cdot 16^1) + (14 \cdot 16^0) + (1 \cdot 16^{-1}) + (14 \cdot 16^{-2}) + (11 \cdot 16^{-3}) + (8 \cdot 16^{-4}) = \\
 &= 2 \cdot 256 + 3 \cdot 16 + 14 \cdot 1 + 1 \cdot 1/16 + 14 \cdot 1/16^2 + 11 \cdot 1/16^3 + 8 \cdot 1/16^4 = \\
 &= 512 + 48 + 14 + 0,0625 + 0,0547 + 0,0027 + 0,0001 = 574,12_{10}
 \end{aligned}$$



### 7.1.4 Conversión Binario ↔ Octal



#### Conversión de Binario a Octal



Para convertir un número binario a octal **se agruparán los dígitos de 3 en 3 a partir del punto decimal hacia la izquierda y después del punto decimal hacia la derecha**. Cada grupo de dígitos binarios se sustituirá por su equivalente octal. En el caso de que no podamos completar un grupo de 3 dígitos lo completaremos con un 0 que siempre se pondrá en los extremos.

Ejemplo:

0 1 0 1 0 1 1 0 0 , 1 1 0  
2 5 4 , 6

$$10101100,110_2 = 254,6_8$$



#### Conversión de Octal a Binario



Para convertir un número octal a binario se sustituirá cada dígito octal por un grupo de tres dígitos binarios. Los ceros al principio y al final del número binario resultante se podrán eliminar.

Ejemplo:

2 5 4 , 6  
0 1 0 1 0 1 1 0 0 , 1 1 0

$$254,6_8 = 10101100,11_2$$

### 7.1.5 Conversión Binario ↔ Hexadecimal



#### Conversión de Binario a Hexadecimal



Para convertir un número binario a hexadecimal se agruparán los dígitos de 4 en 4 a partir del punto decimal hacia la izquierda y después del punto decimal hacia la derecha. Cada grupo de dígitos binarios se sustituirá por su equivalente hexadecimal. En el caso de que no podamos completar un grupo de 4 dígitos lo completaremos con un 0 que siempre se pondrá en los extremos.

Si alguno de los números está entre 10 y 15 se sustituirá por su letra correspondiente.

Ejemplo:

$$\begin{array}{ccc}
 10101001,1100 \\
 \text{---} \quad \text{---} \quad \text{---} \\
 10 \quad 9 \quad 12 \\
 \downarrow \quad \downarrow \quad \downarrow \\
 A \quad 9 \quad C \\
 \\
 10101001,11_2 = A9,C_8
 \end{array}$$



#### Conversión de Hexadecimal a Binario



Para convertir un número hexadecimal a binario se sustituirá cada dígito hexadecimal por un grupo de cuatro dígitos binarios. Los ceros al principio y al final del número binario resultante se podrán eliminar. Si alguno de los números hexadecimales es una letra se pasará primero al número correspondiente.

Ejemplo:

$$\begin{array}{ccc}
 A \quad 9, \quad C \\
 \downarrow \quad \downarrow \quad \downarrow \\
 10 \quad 9, \quad 12 \\
 \swarrow \quad \downarrow \quad \searrow \\
 1010 \quad 1001 \quad 1100 \\
 \\
 A9,C_{16} = 10101001,11_2
 \end{array}$$

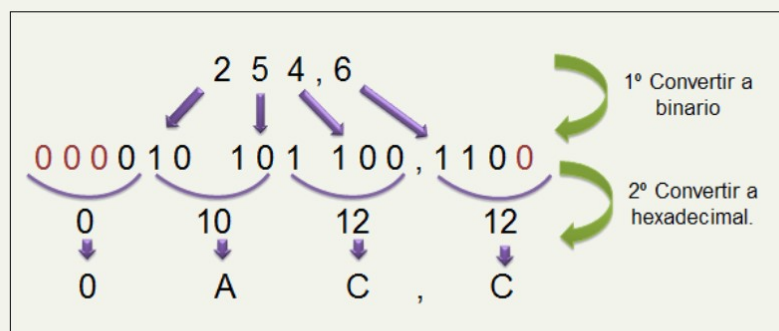
### 7.1.6 Conversión Octal $\Leftrightarrow$ Hexadecimal



#### Conversión de Octal a Hexadecimal

Para convertir un número de octal a hexadecimal primero se realizará una conversión del número octal a binario y después de binario a hexadecimal.

**Ejemplo:**



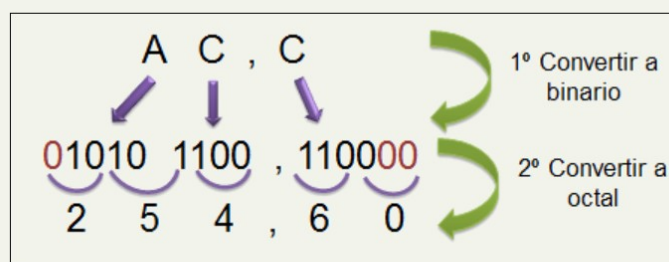
$$254,6_8 = AC,C_{16}$$



#### Conversión de Hexadecimal a Octal

Para convertir un número de hexadecimal a octal primero se realizará una conversión del número hexadecimal a binario y después de binario a octal.

**Ejemplo:**



$$AC,C_{16} = 254,6_8$$

## 7.2. Sistemas de codificación alfanumérica

Son aquellos que nos permiten representar caracteres alfanuméricos como son las cifras decimales, letras, caracteres especiales y de control. Algunos de los principales sistemas de codificación alfanumérica son: ASCII y UNICODE.

### 7.2.1 Código ASCII (American Standard Code for Information Interchange)

El código ASCII se empezó a utilizar para el intercambio de información entre sistemas. Al principio utilizaba únicamente 7 bits para la codificación de letras mayúsculas y minúsculas, signos de puntuación y signos gráficos, pero más tarde se amplió a 8 bits para incluir caracteres de otros idiomas diferentes del inglés como la ñ o los acentos. Aun así, es un sistema que no permite dar soporte a todos los caracteres específicos de los diferentes idiomas que existen en el mundo.

Caracteres ASCII de control			Caracteres ASCII imprimibles			ASCII extendido				
00	NULL	(carácter nulo)	32	espacio	64	@	96	`	128	Ç
01	SOH	(inicio encabezado)	33	!	65	A	97	a	129	ü
02	STX	(inicio texto)	34	"	66	B	98	b	130	é
03	ETX	(fin de texto)	35	#	67	C	99	c	131	â
04	EOT	(fin transmisión)	36	\$	68	D	100	d	132	ä
05	ENQ	(consulta)	37	%	69	E	101	e	133	à
06	ACK	(reconocimiento)	38	&	70	F	102	f	134	á
07	BEL	(timbre)	39	'	71	G	103	g	135	ç
08	BS	(retroceso)	40	(	72	H	104	h	136	ê
09	HT	(tab horizontal)	41	)	73	I	105	i	137	ë
10	LF	(nueva línea)	42	*	74	J	106	j	138	è
11	VT	(tab vertical)	43	+	75	K	107	k	139	ï
12	FF	(nueva página)	44	,	76	L	108	l	140	î
13	CR	(retorno de carro)	45	-	77	M	109	m	141	í
14	SO	(desplaza afuera)	46	.	78	N	110	n	142	Ä
15	SI	(desplaza adentro)	47	/	79	O	111	o	143	Å
16	DLE	(esc.vínculo datos)	48	0	80	P	112	p	144	É
17	DC1	(control disp. 1)	49	1	81	Q	113	q	145	æ
18	DC2	(control disp. 2)	50	2	82	R	114	r	146	Æ
19	DC3	(control disp. 3)	51	3	83	S	115	s	147	ô
20	DC4	(control disp. 4)	52	4	84	T	116	t	148	ö
21	NAK	(conf. negativa)	53	5	85	U	117	u	149	ó
22	SYN	(inactividad sínc)	54	6	86	V	118	v	150	û
23	ETB	(fin bloque trans)	55	7	87	W	119	w	151	ù
24	CAN	(cancelar)	56	8	88	X	120	x	152	ÿ
25	EM	(fin del medio)	57	9	89	Y	121	y	153	Ö
26	SUB	(sustitución)	58	:	90	Z	122	z	154	Ü
27	ESC	(escape)	59	;	91	[	123	{	155	ø
28	FS	(sep. archivos)	60	<	92	\	124		156	£
29	GS	(sep. grupos)	61	=	93	]	125	}	157	Ø
30	RS	(sep. registros)	62	>	94	^	126	~	158	×
31	US	(sep. unidades)	63	?	95	_			159	f
127	DEL	(suprimir)							160	á
									161	í
									162	ó
									163	ú
									164	ñ
									165	Ñ
									166	ª
									167	º
									168	¿
									169	®
									170	¬
									171	½
									172	¼
									173	¿
									174	«
									175	»
									176	™
									177	™
									178	™
									179	
									180	¡
									181	À
									182	Â
									183	Ã
									184	©
									185	ª
									186	¶
									187	§
									188	¶
									189	¥
									190	¥
									191	ƒ
									192	Ł
									193	ł
									194	Ł
									195	ł
									196	—
									197	†
									198	‡
									199	Ä
									200	Å
									201	ƒ
									202	Ł
									203	Ł
									204	Ł
									205	=
									206	Ł
									207	Ł
									208	ø
									209	ø
									210	Ê
									211	Ê
									212	Ê
									213	ı
									214	ı
									215	ı
									216	ı
									217	ı
									218	ı
									219	ı
									220	ı
									221	ı
									222	ı
									223	ı
									224	Ó
									225	ß
									226	Ô
									227	Õ
									228	ö
									229	Ö
									230	µ
									231	þ
									232	þ
									233	Ú
									234	Û
									235	Ü
									236	Ý
									237	Ÿ
									238	—
									239	’
									240	≡
									241	±
									242	—
									243	¼
									244	¶
									245	§
									246	÷
									247	°
									248	…
									249	•
									250	•
									251	•
									252	•
									253	•
									254	■
									255	nbsp

### 7.2.2 Código UNICODE (Universal Code)

Es el sistema de codificación más utilizado en la actualidad y sobre todo en Internet ya que permite codificar sin problemas la mayoría de los idiomas. Para ello, utiliza 16 bits, permitiendo codificar hasta 65536 caracteres. En Unicode podemos utilizar tres tipos de codificación: UTF-8, UTF-16 y UTF-32 en función de la cantidad de bits que vayamos a utilizar y si su longitud es variable o no.

## 8. Bibliografía de referencia

- “Cuadernillo de ejercicios de SOM”. Cefire. Autoras: Jana Taboada, Carmen Fernández y Elisa Noguera.
- “Sistemas Operativos Monopuesto”. Editorial RA-MA.
- “Sistemas Operativos Monopuesto”. Editorial Paraninfo.