
Capítulo 1

Introducción a los sistemas informáticos

Índice

1.1. Informática	3
1.2. Sistemas informáticos	4
1.2.1. Personas del sistema informático	5
1.2.2. Hardware de un ordenador	6
Unidad Central de Proceso (UCP)	7
Dispositivos de Almacenamiento	8
Unidades de Entrada/Salida	9
1.2.3. Software del ordenador	9
Clasificación del Software	10
1.2.4. Clasificación de los sistemas informáticos	13
1.3. Representación de la información	14
1.3.1. Sistemas de numeración	16
Conversión números binarios a octal	18
Conversión números octal a binario	18
Conversión números binarios a hexadecimal	19
Conversión números hexadecimales a binario	19
Aritmética binaria de números sin signo	19
1.3.2. Representación de números	22

1. Introducción a los sistemas informáticos

Números enteros positivos	22
Números enteros con signo	23
Números reales	24
1.3.3. Codificación de texto	28
1.3.4. Codificación de imágenes	29
1.3.5. Codificación del sonido	31
1.4. Ejercicios	31

SECCION 1.1

Informática

Desde hace muchos años, al hombre le ha sido necesaria la utilización de elementos que le permitieran la repetición de ciertos trabajos rutinarios, así como la evolución de esos elementos para hacerle la vida más grata y más descansada. Un elemento esencial en el progreso de la vida del hombre ha sido y sigue siendo la información.

La información es toda forma de representación de hechos, objetos, valores, conceptos, etc. que permite la comunicación entre las personas, además de adquirir conocimiento de las cosas que nos rodean.

La información ha sido utilizada y tratada de múltiples formas a lo largo de los tiempos; libros, periódicos, . . . En los años 40 la información comienza a ser manejada por las máquinas electrónicas dando lugar a la informática.

- **Definición de Informática:** Es la ciencia que estudia el tratamiento de la información de forma automática. La palabra informática se compone de las sílabas ‘**info**’ que proviene de la palabra información y ‘**mática**’ de la palabra automática.

Por lo tanto el objetivo de la informática es procesar información de forma automática.

Datos de entrada → Procesamiento información → Datos de salida

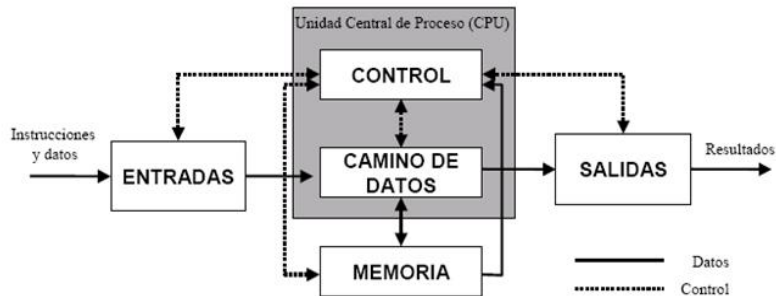


Figura 1.1: Esquema del procesamiento de información

La informática se ocupa de las siguientes tareas:

- Del desarrollo de nuevas máquinas.
- Del desarrollo de software para manejar las máquinas. (Sistemas Operativos, utilidades, compiladores, intérpretes. . .).
- Del desarrollo de aplicaciones informáticas o programas. (Software de gestión, software matemático, videojuegos. . .).

SECCION 1.2

Sistemas informáticos

Se denomina *sistema informático* al conjunto de elementos necesarios para la realización, utilización y mantenimiento de aplicaciones informáticas que realizan un determinado trabajo.

Un sistema informático está formado por una serie de elementos que se engloban en tres grupos:

1. **Las personas:** Usuarios, programadores, analistas, técnicos, . . .
2. **Ordenadores:**

1. Introducción a los sistemas informáticos

- **Hardware:** Dispositivos de entrada, unidad de procesamiento, dispositivos de salida,...
- **Software:** Sistemas Operativos, aplicaciones, lenguajes de programación,...

3. Redes: Internet (Asignatura de redes de área local)

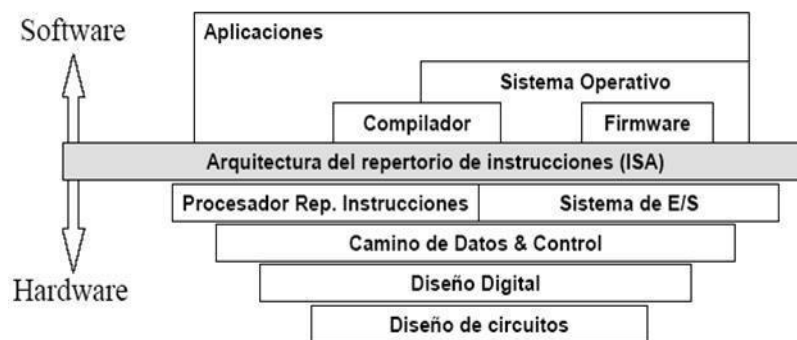


Figura 1.2: Estructura de un computador. Hardware y Software

1.2.1 Personas del sistema informático

Con la aparición de la informática surge un nuevo colectivo de profesionales que trata de abarcar todos los conocimientos que esta ciencia provoca, un colectivo de profesionales de la informática dedicados a la investigación, el desarrollo y explotación de los sistemas informáticos y la formación de los usuarios.

La clasificación de los profesionales de la informática está relacionada con el momento del proceso de implantación del sistema informático en el que realicen su trabajo:

- **Usuarios finales:** Destinatarios finales de cualquier sistema informático. El usuario es aquella persona que, sin necesidad de grandes conocimientos de informática (pero con gran conocimiento de las posibilidades de los programas que usa), es capaz de realizar su trabajo utilizando un sistema informático.

1. Introducción a los sistemas informáticos

- **Administradores, SuperUsuario o Root:** El objetivo principal de un administrador es mantener el correcto funcionamiento de un sistema informático. Es responsabilidad y tarea del administrador:
 - Administrar los recursos de un sistema informático.
 - Planificar, establecer y mantener las políticas de seguridad del sistema.
 - Evaluar el sistema informático para comprobar que cumple los requisitos para el que fue creado.
- **Operadores:** Encargados de ‘introducir datos’ al sistema informático.
- **Personal de dirección:** Encargados de dirigir y coordinar un ‘centro de proceso de datos’.
- **Personal de mantenimiento, técnicos:** Encargados del mantenimiento (generalmente físico) del sistema informático.
- **Otros:**
 - **Fabricantes:** Encargados de fabricar el elemento físico del sistema informático.
 - **Vendedores, comerciales:** Encargados de comercializar elementos físicos y lógicos del sistema informático.
 - **Formadores, informadores:** Encargados de transmitir sus conocimientos informáticos a los usuarios para su correcto aprovechamiento.

1.2.2 Hardware de un ordenador

Un sistema informático está formado por un conjunto de diferentes elementos físicos (electrónicos, mecánicos...) encargados cada uno de ellos de una labor muy específica dentro de todo el conjunto.

Podemos definir el **elemento físico o hardware** como: ‘Todo mecanismo, dispositivo, etc., que se puede tocar o ver de un ordenador, constituye el elemento material, es decir, físico de un sistema informático’.

Desde el punto de vista de la función que realizan dentro del sistema informático. Encontramos:

1. Introducción a los sistemas informáticos

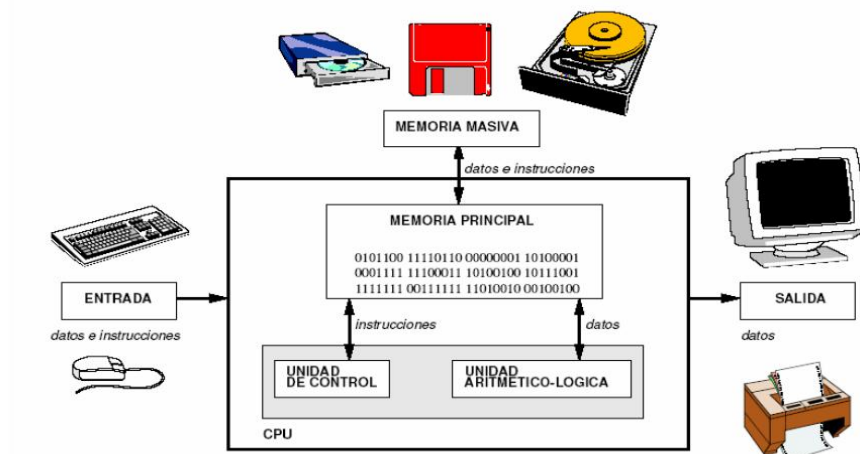


Figura 1.3: Arquitectura de John Von Neumann

Unidad Central de Proceso (UCP)

La Unidad Central de Proceso, del inglés Central Processing Unit (CPU), es la parte encargada de almacenar y procesar la información suministrada a la máquina. Es lo que se conoce comúnmente como procesador, se compone de:

- **Unidad de control (UC):** Supervisa y ordena las acciones a realizar (señales de control). Manipula los datos según las instrucciones que posee.
- **Unidad Aritmético-Lógica (ALU):** Se encarga del tratamiento de la información. Son circuitos combinatoriales que trabajan con números enteros.
- **Unidad de coma flotante:** Está en el coprocesador. Trabaja con números decimales.
- **Elementos de almacenamiento interna:** Memoria caché.
- **Registros:** Son memorias muy pequeñas, temporales. Están en el microprocesador.
- **Elementos de transporte de información:** Buses, líneas que comunican todos los elementos de la unidad central.

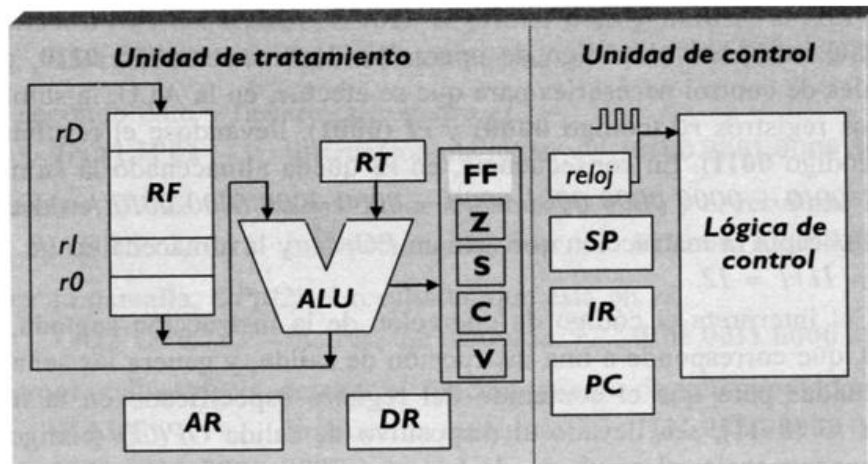


Figura 1.4: Esquema de un microprocesador

Dispositivos de Almacenamiento

Su función es suministrar, recibir y almacenar la información (datos y programas) que procesa la unidad central de proceso.

- **Elementos de almacenamiento interno o memoria principal.** Es el elemento encargado de almacenar instrucciones, programas o datos con los que el procesador está trabajando. No se puede ejecutar ningún programa si no está cargado en memoria interna o central. Existen dos tipos de memorias internas:
 - **RAM (Random Access Memory).** Memoria de acceso aleatorio. En ella es posible almacenar y modificar información.
 - **ROM (Read Only Memory).** Es una memoria de solo lectura, cuya información no puede ser modificada.
- **Elementos de almacenamiento externo:** Son dispositivos con gran capacidad de almacenamiento de información, son más lentos que las memorias internas pero no se pierde la información al dejar de suministrar fluido eléctrico al ordenador. Existen diferentes tipos:

1. Introducción a los sistemas informáticos

- **Disco duro:** Son de elevada capacidad y alta velocidad. Se utilizan para instalar en ellas el software de los sistemas operativos y la mayor parte del software de aplicaciones.
- **Unidades de disquete:** Son periféricos de E/S que permiten almacenar o extraer información de los disquetes.
- **CD-ROM, DVD, ...**
- **Memorias Flash**

Unidades de Entrada/Salida

Suministran información a la Unidad Central y/o reciben la información procesada, es decir, son las encargadas de la comunicación con el exterior y de traducir sus diferentes lenguajes.

- **Periféricos:** Son dispositivos que sirven para introducir datos y programas en el ordenador y mostrar el resultado al exterior. Se clasifican en
 - **Periféricos de entrada:** Ratón, teclado, escáner, micrófono, webcam, cámara digital,...
 - **Periféricos de salida:** Monitor, impresora, altavoces,...
 - **Periféricos de entrada/salida:** Tarjeta de red, módem.
- **Elementos de transporte:** Transportan la información entre los diferentes componentes desde los componentes: Buses, puertos, cables, slots,...

1.2.3 Software del ordenador

Toda información introducida en un ordenador pasa por una serie de procesos hasta conseguir el resultado deseado. El elemento lógico es el que se encarga de indicar el tratamiento a seguir con la información, para que el sistema físico funcione y preste el servicio esperado al ser humano.

Una de las definiciones más utilizada dice que el elemento lógico: 'Es la parte *no visible*, inmaterial, que indica lo que debe hacer el elemento físico para realizar los trabajos encomendados al sistema informático.'

1. Introducción a los sistemas informáticos

Para poder comunicar dicha información a la máquina se utilizan los programas donde se describen las acciones a realizar (generalmente escribiéndolas) agrupándolas (según una determinada estructura/organización). Para escribir los programas se utilizan los *lenguajes de programación*, que son notaciones utilizadas para escribir programas mediante un conjunto de instrucciones.



Figura 1.5: Jerarquía de Software

Clasificación del Software

1. Según se accede a la información (instrucciones y datos):

- **Software.** Los programas en sí. Estos programas residen en memorias externas y son cargados a la memoria interna cuando se pone en marcha el ordenador o cuando se requiera alguno de ellos.
- **Firmware.** Elementos lógicos que traen las computadoras grabado en el momento de su fabricación y que pueden estar en memoria del sistema (ROM) o incorporado en sus circuitos/cableado.

2. Según su utilidad:

- **Software de programación.** Está formado por los programas y utilidades que facilitan la construcción de programas.
 - **Editores.** Permiten la escritura de los programas.
 - **Traductores (Compiladores, intérpretes).** Traducen los programas escritos en lenguajes de programación al lenguaje máquina.

- **Ensambladores.** Los ensambladores traducen el lenguaje ensamblador a lenguaje máquina.
- **Cargadores.** Llevan el programa ejecutable a memoria y lo preparan para su ejecución.
- **Utilidades de rastreo o depuración de errores.** Son utilidades que nos permiten ejecutar los programas de diversas formas (línea a línea, detenerse en alguna línea, etc.) para hacer un seguimiento de las variables y así poder encontrar posibles errores.

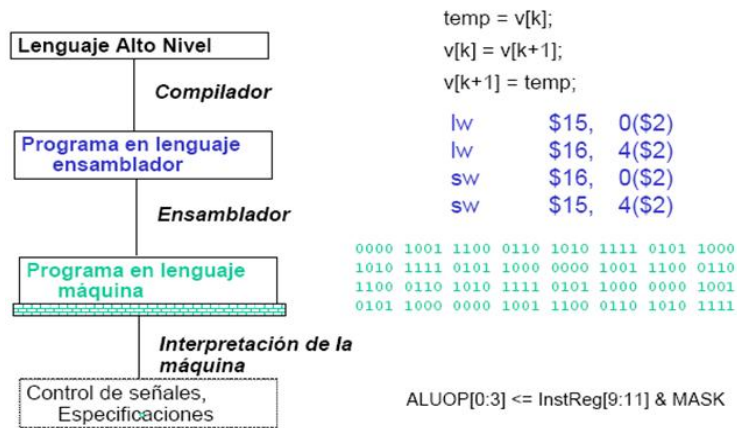


Figura 1.6: Generación de código máquina por un computador

- **Software del sistema operativo.** Son el conjunto de programas que se encargan de controlar el funcionamiento y la gestión de los elementos físicos y lógicos del sistema, de forma *transparente* para el usuario.
 - **Sistema operativo (Núcleo, Kernel).** Manejador de memoria, procesador, periféricos (E/S) y sistemas de almacenamiento.
 - **Interfaz.** Programas que comunican al hombre con la máquina.
 - **Textual.** Consolas de Linux, consola de Ms-Dos.
 - **Gráfico.** X-Windows en Linux (Xfree86), ventanas Windows.
 - **Drivers (controladores).** Programas que hay que cargar para que funcionen los periféricos. Es el software que sabe cómo funciona específicamente el dispositivo físico.
- **Mantenimiento y diagnóstico.** Controla el funcionamiento del sistema operativo. Son las llamadas herramientas del sistema.

- **Software de diagnóstico y mantenimiento.** Es el software utilizado por el personal encargado de la puesta a punto de los equipos. Con este software se pretende localizar averías de un periférico o encontrar el mal funcionamiento de un paquete software, como por ejemplo los programas *TuneUp utilities*, *Norton utilities*,...
- **Software de aplicación.** Creados a partir de los lenguajes de programación y diseñados para resolver problemas específicos o generales de las personas.
 - **Procesadores de texto.** Diseñados para la elaboración de documentos. Simula el funcionamiento de una máquina de escribir aunque pueden combinar texto e imágenes.
 - **Hojas de cálculo.** Creados para trabajar con gran cantidad de datos numéricos, realizar operaciones de cálculo complejas. Permiten obtener gráficos a partir de los datos manipulados.
 - **Bases de datos.** Permiten manipular información de distintos tipos, mediante fichas o registros. Se suelen agrupar por características comunes formando los ficheros de datos, que podrán ser modificados y actualizados.
 - **Paquetes integrados.** Reúnen en una sola aplicación/software las características fundamentales de los tres anteriores. Suelen incluir algún otro tipo de programa (dibujos, comunicaciones,...).
 - **Diseño gráfico.** Dedicados a la elaboración y manipulación de cualquier tipo de imagen.
 - **Autoedición.** Dirigidos a profesionales de las publicaciones impresas, permite combinar texto e imágenes para componer periódicos, revistas, publicidad, etc.
- Según para quién y por qué se crean:
 - **Verticales/Propósito especial/A medida.** Creados para tareas determinadas y que sólo resuelven un problema concreto, suelen ser solicitadas por una empresa de un sector determinado para resolver única y exclusivamente sus necesidades.
 - **Propósito general/Estándar.** Son programas de uso general especialmente diseñados para su lanzamiento al mercado. Estos programas son utilizados por un gran número de usuarios y sobre

1. Introducción a los sistemas informáticos

diferentes sistemas informáticos, pues son creados para resolver problemas generales de trabajo y que con la utilización de sus diferentes opciones, permite al usuario obtener su objetivo final. Estos programas fueron diseñados para evitar los costes de los programas verticales y ser utilizados para diversos fines.

- Según su distribución/comercialización:
 - **De pago.** Son los programas comercializados normalmente.
 - **Shareware.** Su principal característica reside en que se distribuyen de forma gratuita; generalmente son versiones de prueba en la que alguna de sus opciones no se puede ejecutar. Si después de ser probados se está interesado en la versión final, ésta se puede solicitar al precio del producto.
 - **Freeware.** Son los programas de libre distribución comercial.

1.2.4 Clasificación de los sistemas informáticos

- **Sistemas monousuario.** Son los sistemas en los que sólo una persona puede estar conectado a la máquina en un instante dado. (Sistemas operativos: Ms-Dos, Windows 98, Windows Milenium, . . .).
- **Sistemas multiusuario.** Son los sistemas en los que varias personas se pueden conectar simultáneamente a una máquina. (Sistemas operativos: Linux, Windows 2003 Server).
- **Sistemas en red.** Son los sistemas en los que las máquinas se comunican entre sí a largas distancias.

Ejercicio 1.1 *¿Qué significa la palabra informática?*

Ejercicio 1.2 *¿Cuál es el objetivo de la informática?*

Ejercicio 1.3 *Diferencia entre hardware y software.*

Ejercicio 1.4 *Indica tres elementos hardware y otros tres elementos software.*

1. Introducción a los sistemas informáticos

Ejercicio 1.5 *Describir los seis niveles de la estructuración jerárquica de un computador (capas de software).*

Ejercicio 1.6 *Comentar cuál es el efecto de eliminar la memoria RAM de un computador.*

Ejercicio 1.7 *Comentar cuál es el efecto de eliminar la memoria de almacenamiento externo.*

Ejercicio 1.8 *La parte software de los componentes hardware se denomina:*

1. ROM
2. SETUP
3. FIRMWARE
4. BIOS
5. CMOS

Ejercicio 1.9 *La información contenida en disquetes y discos duros, ¿es software o hardware?*

Ejercicio 1.10 *¿Qué es un sistema informático? Describe las partes y funciones de un sistema informático.*

Ejercicio 1.11 *¿Puede funcionar un ordenador sin software básico? ¿Puede funcionar un ordenador sin unidad de disco duro?*

Ejercicio 1.12 *¿De qué tres campos se ocupa la informática?*

Ejercicio 1.13 *Razona la diferencia fundamental entre red de computadores e Internet*

SECCION 1.3

Representación de la información

Conviene tener en cuenta que una computadora no es más que un sistema de procesamiento de la información, y que antes de poder iniciar ese procesamiento,

1. Introducción a los sistemas informáticos

hay que reemplazar dicha información de alguna manera para que la máquina sea capaz de reconocerla y utilizar.

En el momento actual los elementos con que se construyen las computadoras son dispositivos electrónicos capaces de representar una de dos posibilidades: 0 y 1 , y en consecuencia es necesaria la existencia de una etapa de traducción entre la representación de la información usada en el mundo real, fácil de entender y manipular por los seres humanos, compuesta por estímulos sonoros, luminosos, olfativos, o de cualquier otro tipo, y la representación adecuada para la computadora, que como ya se ha dicho, consiste únicamente en secuencias de 0 y 1 .

Cuando la computadora finaliza el tratamiento de los datos, en la mayoría de las ocasiones será necesario realizar la traducción inversa: desde la secuencia de 1 y 0 usada por la máquina hasta un conjunto de símbolos, colores, sonidos, o estímulos a los que el ser humano pueda ser sensible.

Cuando se habla de computadoras como máquinas capaces de tratar automáticamente la información (realmente lo que manipulan son representaciones simbólicas de dicha información), se está haciendo referencia a multitud de posibles procesamientos, entre los que se pueden indicar:

- **Adquisición automática de la información.** Información del mundo real que será codificada a fin de lograr una representación de dicha información en el *interior* de la computadora.
- **Almacenamiento de la información.** Quizás la información introducida y/o procesada no es requerida en el mismo instante en el que es adquirida.
- **Transferencia automática de la información.** Permitiendo así la comunicación de dicha información entre distintos sistemas de computación o seres humanos.
- **Representación de la información.** Transferida y/o procesada en un formato inteligible o útil para el ser humano u otra máquina.

Como hemos mencionado anteriormente, los circuitos digitales son capaces de almacenar únicamente dos estados: 0 y 1 . Sin embargo, una computadora es capaz de manejar y almacenar todo tipo de información: **números enteros, reales, imágenes, sonidos, texto, etc.** Por tanto, toda esta información debe reducirse a su representación mediante 0 y 1 .

1. Introducción a los sistemas informáticos

En algunos casos (como los números) su representación binaria puede resultar relativamente sencilla, aunque para la mayoría de los casos será necesario realizar algún tipo de codificación que permita su representación en binario.

Como ejemplo, veremos cómo se representan algunos de los tipos más comunes de información.

1.3.1 Sistemas de numeración

Definimos al sistema de numeración como el conjunto de reglas, convenios y símbolos combinados con palabras que nos permiten expresar verbal y gráficamente los números.

Existen dos sistemas de numeración:

1. **Posicionales.** En los que el valor del dígito depende de la posición que ocupe el dígito respecto a los demás dígitos. Por ejemplo: La numeración decimal.
2. **No posicionales.** En los que el valor del dígito es independiente de la posición que ocupe el dígito respecto a los demás dígitos. Por ejemplo: El sistema romano donde el símbolo V vale 5 tanto en IV como en VI.

El sistema decimal común es un sistema de numeración posicional que emplea 10 símbolos y donde la base es 10. En un sistema de numeración posicional, un número se representa por una sucesión de dígitos, en los que a cada dígito se le asocia un peso en función de su posición. Si la base de un sistema de numeración es b , el peso que se asocia al dígito i es b^i . Por ejemplo:

$$1458 = 1 \cdot 10^3 + 4 \cdot 10^2 + 5 \cdot 10^1 + 8 \cdot 10^0$$

Los sistemas digitales pueden representar de manera natural números en base 2, usando los símbolos $\{0, 1\}$. Por ejemplo el número binario 1010 se puede expresar como:

$$1010 = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$$

1. Introducción a los sistemas informáticos

El número 1010 equivale al número decimal 10. Podemos aplicar la misma idea a un número expresado en cualquier base usando una base b , una cifra x de n cifras se puede representar como:

$$x = X_{n-1} \cdot b^{n-1} + \dots + X_0 \cdot b^0$$

Para representar un número expresado en base 10 en cualquier otra base, basta con dividir sucesivamente dicho número por la base, extrayendo los restos. Por ejemplo, véase la figura 1.7.

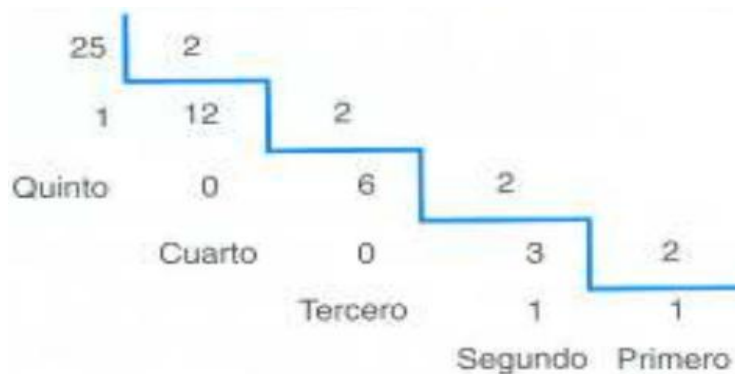


Figura 1.7: Conversión de decimal a binario

En la siguiente tabla se muestran la cantidad de números que se pueden representar dependiendo del número de bits que se utilice.

Bits	Números que se almacenan
8	256
16	65536
24	16.777.216
32	4.294.967.296

Ejercicio 1.14 Convierte a binario los números decimales:

- 6
- 1024
- 43231
- 255

1. Introducción a los sistemas informáticos

El sistema de numeración que emplea solamente los símbolos 0 y 1 se denominado sistema binario. Además, existen otros sistemas de numeración importantes como se puede ver a continuación:

- **Sistema Octal.** Es un sistema en base 8 que utiliza los símbolos del 0 al 7 para representar cantidades, las cuales quedan representadas posicionalmente por potencias de 8.
- **Sistema Hexadecimal.** Es un sistema de numeración en base 16. Utiliza 16 símbolos diferentes, del 0 al 9 y los dígitos valores (o letras) *A, B, C, D, E, F*. Estas letras representan, respectivamente, los dígitos 10, 11, 12, 13, 14, 15 del sistema decimal. Este sistema tiene una correspondencia directa con el sistema binario, ya que cada símbolo en base 16 se puede representar mediante una combinación de 4 bits.

Conversión números binarios a octal

El paso de binario a octal se realiza teniendo en cuenta la expresión $8 = 2^3$. De tal forma que se necesitan tres bits del número en binario para construirlo en el sistema octal. Entonces dado un número N se dividirá en grupos de tres bits comenzando desde el punto decimal y avanzando hacia la izquierda para la cantidad entera y hacia la derecha en la parte fraccionaria. Si no fuese múltiplo entero de 3 se tomarán ceros hasta formar una terna.

$$11110101,0011_{(2)} = 011110101,001100 = 365,14_{(8)}$$

Ejercicio 1.15 *Convierte a octal los siguientes números:*

- $111010100010_{(2)}$
- $34_{(10)}$
- $184,749_{(10)}$

Conversión números octal a binario

El paso de octal a binario se hace en sentido inverso, se sustituye el dígito octal por tres binarios.

Ejercicio 1.16 *Convierte a decimal los siguientes números:*

- $123,321_{(8)}$
- $45_{(8)}$

1. Introducción a los sistemas informáticos

Conversión números binarios a hexadecimal

El paso de binario a hexadecimal se realiza teniendo en cuenta la expresión $16 = 2^4$. El proceso es similar al apartado 1.3.1 salvo que se dividen en grupos de cuatro bits.

$$11110101,0011_{(2)} = 11110101,0011 = F5,3$$

Ejercicio 1.17 Convierte a hexadecimal los siguientes números:

- $0111101001001010_{(2)}$
- $34_{(10)}$
- $184,749_{(10)}$

Conversión números hexadecimales a binario

El paso del sistema hexadecimal a binario se hace en sentido inverso, se sustituye el dígito hexadecimal por cuatro binarios.

Ejercicio 1.18 Convierte a decimal los siguientes números:

- $ABC,DEF_{(16)}$
- $4FA5_{(16)}$

Aritmética binaria de números sin signo

La Unidad Aritmético Lógica, en la CPU del procesador, es capaz de realizar operaciones aritméticas, con datos numéricos expresados en el sistema binario. Naturalmente, esas operaciones incluyen la adición, la sustracción, el producto y la división. Las operaciones se hacen del mismo modo que en el sistema decimal, pero debido a la sencillez del sistema de numeración, pueden hacerse algunas simplificaciones que facilitan mucho la realización de las operaciones. Las tablas son realmente simples:

Suma	Resta	Producto	División
$0 + 0 = 0$	$0 - 0 = 0$	$0 * 0 = 0$	$0/0 = \text{Indeterminado}$
$0 + 1 = 1$	$0 - 1 = 1$	$1 * 0 = 0$	$0/1 = 0$
$1 + 0 = 1$	$1 - 0 = 1$	$1 * 0 = 0$	$1/0 = \text{inf}$
$1 + 1 = 0$	$1 - 1 = 0$	$1 * 1 = 1$	$1/1 = 1$

1. Introducción a los sistemas informáticos

Las sumas $0 + 0$, $0 + 1$ y $1 + 0$ son evidentes:

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

Pero la suma de $1+1$, que sabemos que es 2 en el sistema decimal, debe escribirse en binario con dos cifras (10) y, por tanto $1+1$ es 0 y se arrastra una unidad, que se suma a la posición siguiente a la izquierda. Veamos algunos ejemplos:

$$010 + 101 = 111 \quad (2_{(10)} + 5_{(10)} = 7_{(10)})$$

$$001101 + 100101 = 110010 \quad (13_{(10)} + 37_{(10)} = 50_{(10)})$$

$$1011011 + 1011010 = 10110101 \quad (91_{(10)} + 90_{(10)} = 181_{(10)})$$

$$110111011 + 100111011 = 1011110110 \quad (443_{(10)} + 315_{(10)} = 758_{(10)})$$

La técnica de la resta en binario es, nuevamente, igual que la misma operación en el sistema decimal.

Las restas $0 - 0$, $1 - 0$ y $1 - 1$ son evidentes:

$$0 - 0 = 0$$

$$1 - 0 = 1$$

$$1 - 1 = 0$$

La resta $0 - 1$ se resuelve, igual que en el sistema decimal, tomando una unidad prestada de la posición siguiente: $10 - 1$, es decir, $2_{(10)} - 1_{(10)} = 1$. Esa unidad prestada debe devolverse, sumándola, a la posición siguiente. Veamos algunos ejemplos:

$$111 - 101 = 010 \quad (7_{(10)} - 5_{(10)} = 2_{(10)})$$

$$10001 - 01010 = 00111 \quad (17_{(10)} - 10_{(10)} = 7_{(10)})$$

$$11011001 - 10101011 = 00101110 \quad (217_{(10)} - 171_{(10)} = 46_{(10)})$$

$$111101001 - 101101101 = 001111100 \quad (489_{(10)} - 365_{(10)} = 124_{(10)})$$

La multiplicación en binario es más fácil que en cualquier otro sistema de numeración. Como los factores de la multiplicación sólo pueden ser CEROS o

1. Introducción a los sistemas informáticos

UNOS, el producto sólo puede ser CERO o UNO. En otras palabras, las tablas de multiplicar del cero y del uno son muy fáciles de aprender.

En un ordenador, sin embargo, la operación de multiplicar se realiza mediante sumas repetidas. Eso crea algunos problemas en la programación porque cada suma de dos UNOS origina un arrastre, que se resuelven contando el número de UNOS y de arrastres en cada columna. Si el número de UNOS es par, la suma es un CERO y si es impar, un UNO. Luego, para determinar los arrastres a la posición superior, se cuentan las parejas de UNOS.

Veamos, por ejemplo, una multiplicación en la figura 1.8.

```
      110100010101
    x           1101
    -----
      110100010101
     000000000000
    110100010101
   110100010101
   -----
  1010101000010001
```

Figura 1.8: Multiplicación en Binario

Para comprobar que el resultado es correcto, convertimos los factores y el resultado al sistema decimal: $3349 * 13 = 43537$.

Igual que en el producto, la división es muy fácil de realizar, porque no son posibles en el cociente otras cifras que UNOS y CEROS. Consideremos el siguiente ejemplo, $42 : 6 = 7$, véase la figura 1.9.

Se intenta dividir el dividendo por el divisor, empezando por tomar en ambos el mismo número de cifras (101 entre 110, en el ejemplo). Si no puede dividirse, se intenta la división tomando un dígito más (1010 entre 100).

Si la división es posible, entonces, el divisor sólo podrá estar contenido una vez en el dividendo, es decir, la primera cifra del cociente es un UNO. En ese

```
101010 | 110
- 110
 1001
- 110
  0110
   110
    000
```

Figura 1.9: División en Binario

caso, el resultado de multiplicar el divisor por 1 es el propio divisor. Restamos las cifras del dividendo del divisor y bajamos la cifra siguiente.

El procedimiento de división continúa del mismo modo que en el sistema decimal.

Ejercicio 1.19 *Sumar, restar, multiplicar y dividir los siguientes números binarios:* 1100 y 1111; 110011 y 010101; 0011, 11 y 0101, 01; 1101 y 1111; 110011 y 010101; 0011, 11 y 0101, 01.

En el ejemplo anterior, se observa que multiplicar por 10_2 (es decir, por 2 en decimal) equivale a añadir un cero a la derecha, o *desplazar* el punto decimal a la derecha, siendo esto similar a multiplicar por 10_{10} .

1.3.2 Representación de números

Números enteros positivos

1. **Codificación en binario.** Es la más usual. El número entero se codifica mediante su representación en base 2.

1. Introducción a los sistemas informáticos

$$4310)_{10} = 1000011010110)_2$$

2. **Codificación BCD.** El número se representa en base 10 y cada dígito se codifica en binario usando 4 bits. Esta representación tiene el grave problema de usar más bits de los necesarios.

$$4310)_{10} = 0100001100010000)_2$$

Números enteros con signo

Codificación signo-magnitud. El bit más significativo indica el signo (0 positivo y 1 negativo), mientras que los restantes bits indican el valor absoluto en binario. En este formato es preciso conocer con cuantos bits se representa el número. Si utilizamos n bits podremos representar números dentro del rango: $-(2^{n-1} - 1)$ y $(2^{n-1} - 1)$.

$$\begin{aligned} 43 &= 00101011 \\ -43 &= 10101011 \end{aligned}$$

Un posible problema asociado a esta representación es que existe una doble representación para el cero: $000\dots,0$ y $100\dots,0$. Además, las operaciones aritméticas resultan más complejas que con otras representaciones. Si el número es negativo, se representa en complemento a 1 (C1) o complemento a 2 (C2).

Complemento a 2. Si el número es positivo se representa por su expresión en binario, y si es negativo se representa por la expresión en binario de 2^n menos su valor absoluto, donde n es el número de bits. El número ha de estar comprendido entre -2^{n-1} y $2^{n-1} - 1$. Ejemplo, $n = 8$

$$\begin{aligned} 43 &= 00101011 \\ -43 &= (256 - 43) = 213 = 11010101 \end{aligned}$$

Este sistema de representación es el empleado en todos los procesadores actuales.

1. Introducción a los sistemas informáticos

Complemento a 1. Si el número es positivo se representa por su expresión en binario, y si es negativo se representa por la expresión en binario de $2^{n-1} - 1$ menos su valor absoluto. El número ha de estar comprendido entre $-2^{n-1} + 1$ y $2^{n-1} - 1$. Ejemplo, $n = 8$.

$$43 = 00101011$$
$$-43 = (255 - 43) = 212 = 11010100$$

El complemento a 1 de un número negativo se puede obtener complementando bit a bit el valor binario de su valor absoluto, es decir, cambiando el 0 por el 1 y viceversa.

Dado que el complemento a 2 y el complemento a 1 de un número negativo difieren sólo en una unidad, una forma simple de calcular el complemento a 2 de un número negativo es calcular su complemento a 1 y sumarle 1.

$$43 = 00101011$$
$$-43 = 11010100 - \text{Complemento 1}$$
$$-43 = 11010101 - \text{Complemento 2}$$

Ejercicio 1.20 Convierte los siguientes números a los distintos formatos de representación que aparecen en la tabla, utilizando 8 bits. Indica aquellos que no puedan ser representados y porqué.

Número	binario	octal	hexadecimal	BCD	Signo-Mag	C2	C1
8							
25							
-52							
325							
259							
240							
-128							
-255							
89							

1. Introducción a los sistemas informáticos

Números reales

En un sistema de numeración posicional los números situados a la derecha del punto decimal corresponden a peso de magnitud exponencial negativa:

$$18,374 = 1 \cdot 10^1 + 8 \cdot 10^0 + 3 \cdot 10^{-1} + 7 \cdot 10^{-2} + 4 \cdot 10^{-3}$$

Y utilizando el mismo método se pueden representar números reales en binario

$$10,101 = 1 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = 2,625$$

Para calcular la parte real en binario se multiplica por 2 la parte real en decimal sucesivamente hasta que ésta tome el valor cero. El dígito correspondiente será 0 o 1 dependiendo del valor de la parte entera en cada paso. Veamos un ejemplo:

$0,125 \cdot 2 = 0,25$	0,0
$0,25 \cdot 2 = 0,50$	0,00
$0,50 \cdot 2 = 1,00$	0,001
$0,00 \cdot 2 = 0,00$	0,0010

Fundamentalmente, existen dos formas de codificar los números reales en un ordenador:

Representación en punto fijo. Se actualiza una cantidad fija de bits para representar la parte fraccionaria de la magnitud. El principal problema que presenta esta representación es su falta de flexibilidad: no permite representar números muy grandes o números muy pequeños.

Representación en punto flotante. Para superar las limitaciones de las representaciones en punto fijo de números muy grandes o muy pequeños se suele utilizar la notación científica, en la que el número se divide en mantisa y exponente: $N = M \cdot B^E$.

$$6,023 \cdot 10^{23} \rightarrow \text{Mantisa} = 6,023; \text{Exponente} = 23; \text{Base} = 10$$

1. Introducción a los sistemas informáticos

Este método se puede utilizar también para la representación de números binarios. Un número real se representa entonces mediante cuatro datos: signo del número, mantisa, signo del exponente y exponente.

S_m	<i>Mantisa</i>	S_e	<i>Exponente</i>
-------	----------------	-------	------------------

Cada procesador fija la longitud en bits de los diferentes campos, así como el código que se elige para la representación, y la base de la exponencial. Afortunadamente, existe el estándar de IEEE 754-1985 que normaliza la representación de números en punto flotante. En el estándar se especifican dos niveles de precisión: *precisión simple (32 bits)* y *precisión doble (64 bits)*. En esta tabla se muestra la asignación de bits para cada una de las partes que se representan.

	Signo	Exponente	Mantisa	Sesgo (exp)
Precisión simple	1 [31]	8 [30-23]	23 [22-0]	127
Precisión doble	1 [63]	11 [62-52]	52 [51-00]	1023

El *bit de signo* es tan simple como cabe esperar: si tiene el valor 0 el número será positivo y si vale 1 será negativo.

La parte del *exponente* debe representar tanto los exponentes negativos como los positivos. Para conseguir esto, se utiliza un sesgo que se resta al valor almacenado para obtener el valor real del exponente. Para precisión simple este valor es de 127. Así pues, un exponente de cero significa que 127 es el valor que hay almacenado. Por otro lado, un valor almacenado de 200 implica un exponente de $200 - 127$, es decir 73. Los valores -127 (todo 0) y 128 (todo 1) están reservados para poder representar situaciones especiales, como infinito, valores *NaN* (not a number), etc. Que se pueden obtener en determinadas operaciones. Se le suma a N el sesgo 2^{n-1} , por lo que el resultado siempre es positivo.

La *mantisa* se compone de un bit inicial *implícito* igual a 1, seguido por n bits que representan la magnitud. Para establecer el valor del bit implícito debemos considerar que cualquier número se puede expresar en notación científica e muchas maneras. Por ejemplo, el número 5 se puede expresar como: $5,00 \cdot 10^0$ o $0,05 \cdot 10^2$ o $5000,00 \cdot 10^{-3}$. Con la idea de maximizar la cantidad de números representables, las magnitudes en punto-flotante se almacena en forma normalizada: básicamente, se coloca el punto decimal detrás del primer dígito distinto de cero. De esta manera, 5 se expresa como $5,0 \cdot 10^0$.

1. Introducción a los sistemas informáticos

En base 2 el único valor distinto de cero es 1, con lo cual podemos suponer un bit inicial que siempre vale uno y por tanto no necesitamos representarlo explícitamente. Así pues, tendremos un bit más de resolución en la mantisa, es decir, se asume que la mantisa tiene el formato $1.f$, donde f es la parte fraccionaria almacenada.

Algunas situaciones especiales:

- **Si $e = 0$.** La mantisa se almacena desnormalizada y el valor de $E = e - 2^{ne-1} + 2$.
- **El número N=0.** Se representa con todos los bits del exponente y la mantisa a 0.
- **Si todos los bits del exponente son 1:**
 - Si $m = 0 \rightarrow$ representa $\pm \text{inf}$
 - Si $m \neq 0 \rightarrow$ representa NaN

Veamos a continuación un ejemplo. $N = -37,48$, $n = 16$ bits y $ne = 8$.

1. $37,48_{(10)} = 111101,01111010_{(2)}$.
2. Normalizar el número y ponerlo de forma exponencial $1,11101011110102^5$.
3. Signo: 1 = Número negativo.
4. Exponente: $e = E + S = 5 + 2^{8-1} - 1 = 132_{(10)} = 11000100_{(2)}$
5. Mantisa: $nm = 16 - 1 - 8 = 7 \rightarrow m = 1110101$
6. N: 111001001110101 \rightarrow Signo Exponente Mantisa

Ejercicio 1.21 Realizar las siguientes conversiones

1. Convierte 1011 1001 0100 1100 0000 0000 0000 0000 (IEEE-754 simple precisión) a decimal
2. Convierte 4.625 a formato IEEE 754 simple precisión (32-bit)

S	C	M
---	---	---
3. Convierte 0.0234375 a formato IEEE 754 simple precisión

S	C	M
---	---	---
4. Convierte 154.25 a formato IEEE 754 simple precisión

S	C	M
---	---	---

Código Standard ASCII (Caracteres Alfanuméricos)

Código Decimal	Carácter ASCII	Código Decimal	Carácter ASCII	Código Decimal	Carácter ASCII	Código Decimal	Carácter ASCII	Código Decimal	Carácter ASCII	Código Decimal	Carácter ASCII
33	!	49	1	65	A	81	Q	97	a	113	q
34	"	50	2	66	B	82	R	98	b	114	r
35	#	51	3	67	C	83	S	99	c	115	s
36	\$	52	4	68	D	84	T	100	d	116	t
37	%	53	5	69	E	85	U	101	e	117	u
38	&	54	6	70	F	86	V	102	f	118	v
39	'	55	7	71	G	87	W	103	g	119	w
40	(56	8	72	H	88	X	104	h	120	x
41)	57	9	73	I	89	Y	105	i	121	y
42	*	58	:	74	J	90	Z	106	j	122	z
43	+	59	;	75	K	91	[107	k	123	{
44	,	60	<	76	L	92	\	108	l	124	
45	-	61	=	77	M	93]	109	m	125	}
46	.	62	>	78	N	94	^	110	n	126	~
47	/	63	?	79	O	95	_	111	o	127	-
48	0	64	@	80	P	96	`	112	p		

Figura 1.10: Código estándar ASCII

1.3.3 Codificación de texto

La representación de caracteres alfanuméricos en un computador es muy sencilla, y se realiza generalmente mediante una tabla de códigos, en la que se asigna a cada carácter un código binario de 7 u 8 bits. Además de los símbolos gráficos se incluyen símbolos de control como *nueva línea*, *fin de fichero*, *nueva página*, *escape*,...

El código más utilizado es el código ASCII, pensado para el inglés y utiliza 7 bits para codificar cada carácter. En total son 128 símbolos.

También podemos encontrar otras codificaciones, entre las cuales podemos destacar:

- **ISO-8859-1 (Latín 1).** Es una extensión del ASCII, usa 8 bits (256 símbolos) e incluye los lenguajes de Europa occidental.
- **ISO-8859-15 (Latín 9).** Modificación de Latín 1, para incorporar el símbolo del Euro, además de otras modificaciones.
- **ISO-8859-2 a ISO-8859-14.** Extensiones del ASCII para diferentes len-

1. Introducción a los sistemas informáticos

Código Standard ASCII extendido (Caracteres Alfanuméricos)

Cod	Caract	Cod	Caract	Cod	Caract	Cod	Caract	Cod	Caract	Cod	Caract	Cod	Caract	Cod	Caract
128	€	144	□	160	ı	176	ˆ	193	À	209	Ñ	225	á	241	ñ
129	◻	145	ˆ	161	ı̇	177	±	194	Á	210	Ò	226	â	242	ò
130	,	146	ˆ	162	€	178	²	195	Â	211	Ó	227	ã	243	ó
131	f	147	“	163	£	179	³	196	Ã	212	Ô	228	ä	244	ô
132	„	148	”	164	¤	180	´	197	Ä	213	Õ	229	å	245	õ
133	…	149	•	165	¥	181	µ	198	Å	214	Ö	230	æ	246	ö
134	†	150	-	166	¦	182	¶	199	Ç	215	×	231	ç	247	÷
135	‡	151	-	167	§	183	·	200	È	216	Ø	232	è	248	ø
136	^	152	-	168	¨	184	,	201	É	217	Ù	233	é	249	ù
137	‰	153	™	169	©	185	¹	202	Ê	218	Ú	234	ê	250	ú
138	§	154	š	170	ª	186	º	203	Ë	219	Û	235	ë	251	û
139	‹	156	œ	171	«	187	»	204	Ì	220	Ü	236	ì	252	ü
140	Œ	157	◻	172	¬	188	¼	205	Í	221	Ý	237	í	253	ý
141	◻	158	ž	173	™	189	½	206	Î	222	Þ	238	î	254	þ
142	Ž	159	Ÿ	174	®	190	¾	207	Ï	223	ß	239	ï	255	ÿ
143	◻	192	À	175	¯	191	¿	208	Ð	224	à	240	ò		

Figura 1.11: Código extendido ASCII

guas: cirílico, árabe, griego, hebreo, etc.

- **ISO-10646 (UNICODE)**. Código de 16 bits que engloba a todas las codificaciones e idiomas conocidos.

Ejercicio 1.22 *Codificación de mensaje ASCII: Escribir un mensaje sencillo con 3 ó 4 palabras. Codificarlo en ASCII (hex). Pasarlo a binario. Entregar el mensaje en binario a otro estudiante para que lo decodifique.*

1.3.4 Codificación de imágenes

Una imagen natural capturada con una cámara, un telescopio, un microscopio o cualquier otro tipo de instrumento óptico presenta una variación de sombras y tonos continua. Las imágenes de este tipo se llaman imágenes analógicas.

Para que una imagen analógica, en blanco y negro, en escala de grises (las llamadas comúnmente, imágenes en blanco y negro), o color, pueda ser *manipulada* usando un ordenador, primero deben convertirse a un formato adecuado. Este formato es la imagen digital correspondiente.

La transformación de una imagen analógica a otra digital se llama digitalización y es el primer paso en cualquier aplicación de procesamiento de imágenes digitales. El proceso de digitalización consta de dos partes: **muestreo y cuantificación**.

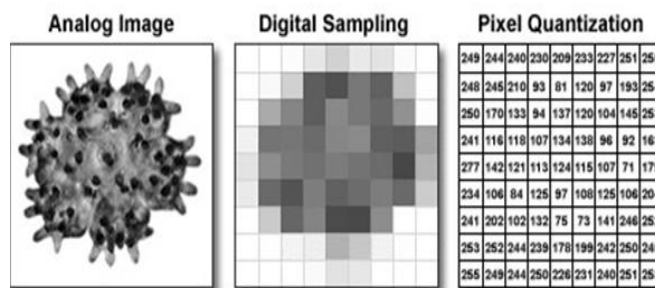


Figura 1.12: Muestreo y cuantificación de una imagen

Un *muestreo* consiste en una subdivisión de la imagen analógica en porciones cuadradas. Estos polígonos representan sensores sensibles a la intensidad de la luz. La salida de estos sensores es un valor (amplitud) dentro de una escala (color). La salida pueden ser, o bien un único valor (escala de grises) o bien un vector con tres valores por polígono (RGB) que se corresponden con la intensidad de color rojo (R), verde (G) y azul (B). La escala de colores también tiene un rango discreto (por ejemplo, de 8-bits = 256 valores).

Las imágenes en escala de grises con solo 2 colores: blanco y negro (0 y 1, respectivamente), se llaman imágenes binarias.

A este proceso de discretización del color se llama *cuantificación*. Una porción de color constante se le llamará píxel.

Como podemos apreciar, una imagen digital (sea del tipo que sea) no es más que una matriz numérica.

La resolución (número de píxeles en pantalla) de una imagen en escala de grises depende directamente de dos parámetros: del muestreo (tamaño de la matriz) y de la cuantificación (número de bits por píxel). Cuanto más se incrementa la matriz, más se aproxima la imagen digitalizada a la original, sin embargo el espacio necesario para almacenarla también se incrementará.

Las imágenes digitales a color están gobernadas por los mismos conceptos de muestreo, cuantificación y resolución que las imágenes en escala de grises. Sin

embargo, en lugar de un único valor de intensidad que expresa el nivel de gris, los píxeles de las imágenes a color están cuantificados usando tres componentes independientes uno por cada color primario (RGB). Combinando distintas intensidades de estos tres colores, podemos obtener todos los colores visibles.

1.3.5 Codificación del sonido

La grabación digital captura el sonido almacenando los valores de amplitud de una onda a intervalos regulares de tiempo. La amplitud (altura) de una onda de sonido determina su volumen; la frecuencia (medida en Hertzios o Hz) determina su escala (lo grave o aguda que suena).

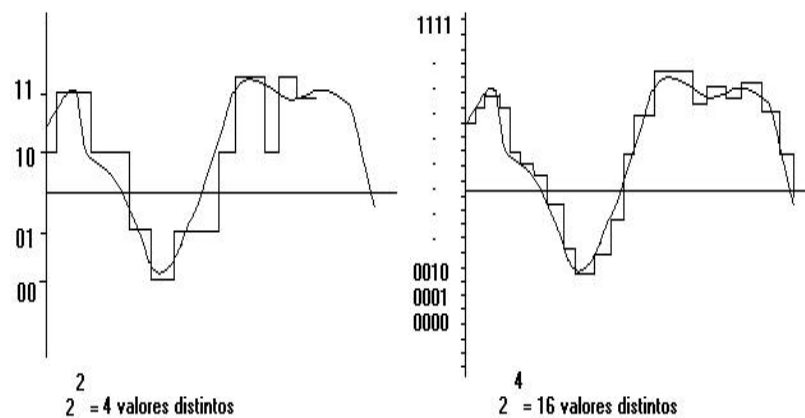


Figura 1.13: Muestreo y cuantificación del sonido

Las ondas de sonido son continuas en la naturaleza, pero como ya sabemos, un ordenador sólo puede trabajar con información digital (0,1). Así que el ordenador almacena la amplitud de una señal a intervalos regulares (muestreo), usando un número determinado de bits (cuantificación). La frecuencia de muestreo indica cuántas muestras del sonido se toman en un segundo. Así una frecuencia de 22 KHz indica 22.000 muestras por segundo.

El número de bits también influye en la calidad de la grabación ya que indica el número de niveles de amplitud del sonido: con 8 bits podremos medir 256 niveles

1. Introducción a los sistemas informáticos

en cada muestra y con 16 bits, 65535. La calidad CD es un estándar que indica que ese sonido está grabado a 16 bits y 44,1 KHz.

SECCION 1.4

Ejercicios

- El número 87 en base octal es el equivalente al:
 - 1100001 en binario
 - 61 en hexadecimal
 - 141 en octal
 - son correctas a y b
 - Todas son falsas
- 2048 cuartetos son:
 - 1024 octetos
 - 4096 octetos
 - 8 kilobytes
 - Son correctas a y c
 - Son correctas b y c
- ¿Qué es un Sistema Operativo?
- Representar en notación hexadecimal:
 - $0110101011110010_{(2)}$
 - $111010000101010100010111_{(2)}$
 - $01001000_{(2)}$
 - ¿Qué ventajas encuentras en la notación hexadecimal?
- Representar en binario:
 - $5FD97_{(16)}$
 - $610A_{(16)}$
 - $ABCD_{(16)}$
 - $0100_{(16)}$
 - $45A0_{(16)}$
 - $HF_{(16)}$
 - $AAB2_{(16)}$
 - $3020_{(16)}$
- ¿Qué dice el siguiente mensaje ASCII?
1101100110111111100110100000110001111011111011011110000
1110101111010011000011100100110111111001011001011110011

1. Introducción a los sistemas informáticos

7. En el conjunto de caracteres ASCII, ¿qué relación matemática hay entre los códigos correspondientes en mayúsculas? ¿y entre los códigos de los dígitos decimales y sus valores reales? Establece una fórmula que permita calcular el valor numérico de un dígito a partir de su código de carácter para cualquier conjunto de caracteres (no sólo el ASCII) que tenga consecutivos los diez caracteres numéricos.
8. Codificar en ASCII la frase: *¡Me gusta el fútbol!* ¿Notas alguna deficiencia en este código?
9. Convierte al sistema decimal
- a) 0101_2 b) 1001_2 c) 1011_2 d) 0110_2
e) 10010_2
10. Convierte a binario los números decimales:
- a) 11,11 b) 7618,255 c) 123,45 d) 0,234
e) 555,34 f) 123432,79
11. ¿Cuál es el valor numérico más grande que puede representarse con tres bytes si cada dígito se codifica empleando un patrón ASCII por byte? ¿Y si se usara la notación binaria con el mismo número de bytes?
12. Expresar en binario los siguientes valores expresados como números mixtos (parte entera + parte decimal):
- a) $4\frac{1}{2}$ b) $2\frac{3}{4}$ c) $1\frac{1}{8}$ d) $5\frac{5}{16}$
e) $5\frac{5}{8}$
13. ¿Cuál es el número positivo que se puede almacenar en binario puro en una posición de memoria de un ordenador con una longitud de palabra de 32 bits?
14. Supongamos que la pantalla de un monitor presenta 24 líneas, cada una de las cuales contiene 80 caracteres de texto. Si la imagen de la pantalla se almacenara en la memoria representando cada carácter con su código ASCII (modo texto), ¿cuántos bytes de la memoria de la máquina se requieren para guardar la imagen completa?

1. Introducción a los sistemas informáticos

15. Una imagen de una pantalla se representa mediante una matriz rectangular de 1024 columnas por 768 filas de pequeños puntos o pixels (picture elements). Si se requieren ocho bits para codificar el color y la intensidad de cada pixel (modo gráfico), ¿cuántas celdas de memoria de un byte se requerirán para contener toda la gráfica?
16. Convertir a binario los siguientes números en base decimal: 123,45;0,234;555,34 y 123432,79.
17. Convertir a binario los siguientes números en base hexadecimal: 45A0; HF; AAB2 y 3020.
18. Convertir a hexadecimal los siguientes números binarios: 110001000 y 100010110.
19. Sumar, restar, multiplicar y dividir los siguientes números binarios: 1101 y 1111;110011 y 010101; 0011, 11 y 0101, 01.
20. Sumar, restar, multiplicar y dividir los siguientes números: 1100 1010 1111 0011 + 7BAFh
21. Realizar las sumas anteriores pasando primero los números a decimal y mostrando el resultado en binario.
22. Sean las siguientes cantidades expresadas en base decimal: $-12 + 17 - 25$. Expresarlas utilizando los convenios de representación que se especifican, indicando el tipo de error que se produce
 - a) Binario en signo magnitud con 8 dígitos binarios
 - b) Binario en complemento a 1 con una longitud de 9 dígitos binarios
 - c) Binario en complemento a 2 con 8 dígitos binarios
23. Interpretar las siguientes combinaciones 1110010011 como
 - a) Datos representados en binario natural
 - b) Datos representados en signo magnitud
 - c) Datos representados en Complemento a 1
 - d) Datos representados en Complemento a 2
24. Convierte los siguientes números a los distintos sistemas de representación que aparecen en la tabla.

1. Introducción a los sistemas informáticos

Número	Octal	Hexadecimal	BCD
32			
23			
55			
328			
269			
243			
128			
255			
99			

25. Convierte los siguientes números a base decimal:

Número	Sistema decimal
00101100, 11 ₍₂₎	
10101010, 10 ₍₂₎	
532, 9 ₍₈₎	
333, 55 ₍₈₎	
AB38 ₍₁₆₎	
FC6 ₍₁₆₎	
12 ₍₈₎	

26. Convierte los siguientes números a base decimal

1. Introducción a los sistemas informáticos

Número	Formato	Sistema decimal
00101100	Binario	
10101010	Signo y magnitud	
11111111	Complemento a 2	
11010110	Binario puro	
01000101	Signo y magnitud	
10001011	Complemento a 2	
11010111	Signo y magnitud	
00111110	Complemento a 2	
10001011	Binario	
11110110	Signo y magnitud	
10101110	Binario puro	
11001100	Complemento a 2	
00110011	Signo y magnitud	
10101111	Complemento a 2	
11110000	Binario	

27. Realizar las siguientes operaciones en complemento a 2

	Resultado
$8 + 26$	
$8 - 26$	
$10 + 50$	
$25 - 50$	
$125 - 8$	
$125 - 255$	

28. Dado el siguiente patrón de bits

11000000000000000000000000000000

Qué representa si tenemos en cuenta que es...

- a) Un número entero en complemento a dos.
- b) Un entero sin signo.
- c) Un número en coma flotante con formato IEEE simple precisión.

29. Dado el siguiente patrón de bits

1. Introducción a los sistemas informáticos

11000111111000000000000000000000

Qué representa si tenemos en cuenta que es

- a) Un entero sin signo
- b) Un número en coma flotante con formato IEEE simple precisión
- c) Un número entero en complemento a dos

30. Indique cuál de las siguientes afirmaciones es falsa:

- a) Un sistema de numeración es una forma de representar cualquier cantidad numérica.
- b) Cada posición del dígito tiene un valor intrínseco que aumenta de derecha a izquierda según potencias sucesivas de la base del sistema de numeración empleado.
- c) El sistema de *base 2* utiliza sólo dos símbolos, $\{1,2\}$, denominados bits.
- d) El número de símbolos distintos que se pueden usar en un determinado sistema de numeración constituye su *base*, es decir, base 10 los números que podemos representar son $\{0,1,2,3,4,5,6,7,8,9\}$.

31. Indique cuál de las siguientes afirmaciones es verdadera:

- a) Todo sistema informático está compuesto por dos elementos básicos (hardware y software).
- b) Hay dos tipos de software (de sistema y de aplicación)
- c) El software de sistema es un conjunto de programas necesarios para que el ordenador tenga capacidad de trabajar.
- d) Todas las anteriores son correctas.

