

Parte I - El sistema informático

1. Introducción	2
2. Componentes de un sistema informático	2
3. Organización del ordenador	3
4. La unidad central de proceso (CPU)	4
La unidad aritmético-lógica	4
La unidad de control	5
5. La memoria	7
La memoria principal	7
Tipos de memoria principal	8
Esquema básico de una unidad de memoria	9
Secuencia de pasos para leer/escribir en una posición de memoria	10
Operación de lectura	10
Operación de escritura	10
Jerarquía de memoria	10
Memoria virtual	10
Memoria caché	11
Jerarquía de memoria	11
Direccionamiento (acceso a una dirección de memoria)	12
6. Los buses	12
7. Los periféricos	13
8. Seguridad de la información	14
Seguridad física	14
Seguridad lógica	14
Seguridad de los datos	15

1. Introducción

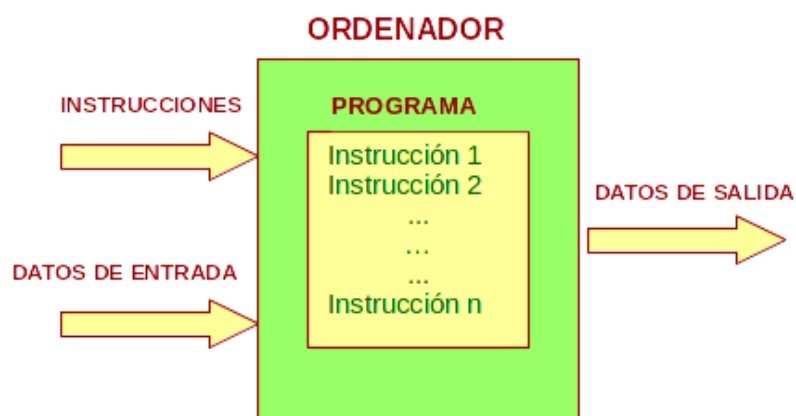
El ordenador es la herramienta que nos permite el **tratamiento automático de la información**, entendiendo por tal su **organización, tratamiento, transmisión y almacenamiento**. Es un instrumento capaz de efectuar cálculos y elaborar datos con extraordinaria rapidez y fiabilidad, pero incapaz de generar información nueva si previamente no se le ha indicado como conseguirla.

Un ordenador solo puede hacer **dos cosas: ejecutar secuencias de operaciones, y optar entre secuencias alternativas** cuando esta posibilidad ha sido prevista.

Un **sistema informático**, en mayor o menor medida, es precisamente esto, un conjunto de elementos de **hardware** y **software** interconectados para el tratamiento de la información. Un ordenador que ejecuta un programa de contabilidad conforma en sí mismo un sistema informático, pero también se puede formar un sistema informático formado por cientos de ordenadores conectados en red que cumplen una determinada función.

Un sistema informático se puede definir como una serie de elementos físicos (hardware) capaz de realizar muchas tareas a gran velocidad y con gran precisión. Para que estos elementos físicos realicen un proceso determinado, es necesario que en él se ejecuten un conjunto de órdenes o instrucciones (software), componentes no físicos que pongan en funcionamiento todos esos componentes físicos. Estas instrucciones ordenadas y agrupadas de forma adecuada, constituyen un programa. El conjunto de varios programas se denomina aplicación informática.

Entre los programas y el hardware se encuentra una aplicación informática especial, que hace de intermediario entre ambos. Ese software se denomina sistema operativo.



Desde una visión externa y muy simplificada, podemos considerar a un ordenador como una especie de caja negra que acepta unas **entradas** (datos e instrucciones) y devuelve unas **salidas** (datos de salida o resultados).

2. Componentes de un sistema informático

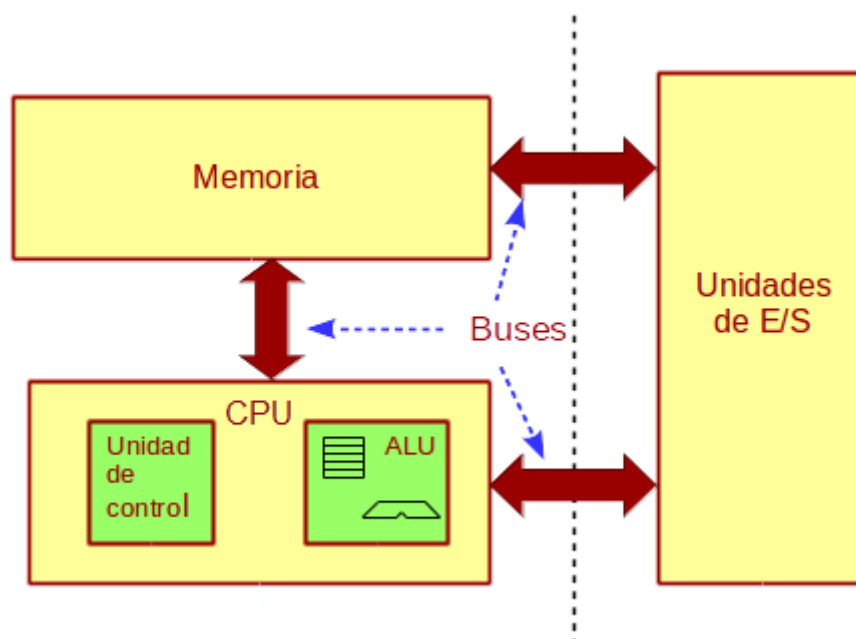
A continuación se indican los elementos que constituyen un sistema informático y que, por consiguiente, hacen posible el desarrollo y aprovechamiento de aplicaciones informáticas.

- ✓ **Hardware:** es el conjunto de materiales **físicos** que componen el sistema informático, es decir, todo el material físico (los elementos tangibles) relacionado con el ordenador (conexiones, cables, etc.) y con cualquier dispositivo conectado al mismo.

- ✓ **Software:** parte **lógica** del sistema informático que dota al equipo físico de la capacidad para realizar cualquier tipo de tareas. El software está formado por el conjunto de programas ejecutables sobre el hardware junto con los documentos y datos asociados a los mismos. Nos permiten operar con el ordenador, así como controlar y coordinar los distintos elementos hardware. En definitiva, es la parte intangible del ordenador, que sabemos que se encuentra en él, pero que solo podemos acceder a ella a través de estos componentes del sistema.
- ✓ **Firmware:** conjunto de instrucciones que las computadoras llevan **pregrabadas de fábrica** en su propia circuitería. Se trata de un software muy integrado con la electrónica de los dispositivos que es capaz de interactuar directamente con el hardware. En función de la técnica y el soporte utilizado para contenerlo, el firmware puede ser actualizado posteriormente.
- ✓ **Personal informático** (o **humanware**): conjunto de **personas** que desempeñan las distintas funciones relacionadas con la utilización y explotación de los ordenadores en una determinada empresa u organización.

3. Organización del ordenador

La **arquitectura Von Neumann** es una estructura, introducida por **John Von Neumann** en 1945 y, aunque con algunas diferencias, su planteamiento continúa utilizándose en la actualidad.



- ✓ La **memoria principal** contiene las instrucciones de los programas y sus datos. Para que un programa pueda ser ejecutado, antes debe cargarse en memoria principal desde alguna unidad de memoria externa.
- ✓ La **unidad central de procesamiento (CPU)**, también se denomina procesador, y que se comercializa como un único circuito integrado es el componente del computador que interpreta las instrucciones contenidas en los programas y procesa los datos. La CPU proporciona la característica fundamental de la computadora digital (la programabilidad) y está formada por dos unidades:
 1. La **unidad de control (UC)**. Es la encargada de ejecutar las instrucciones almacenadas en memoria y de controlar y coordinar el funcionamiento de todo el sistema.
 2. La **unidad aritmético-lógica (ALU)**. Está formada por circuitos electrónicos capaces de realizar operaciones elementales de dos tipos: aritméticas (sumas, restas, multiplicaciones y divisiones) y

lógicas (comparaciones, operaciones or, operaciones and, ...). Cuando la instrucción a ejecutar consiste en una operación de alguno de los tipos indicados, la unidad de control genera las órdenes de control necesarias para que la misma sea llevada a cabo por la ALU.

3. Tanto la **ALU** como la **unidad de control**, disponen de elementos de almacenamiento de información de muy baja capacidad (unos pocos bytes) y de alta velocidad. Estos elementos, denominados **registros** sirven para almacenar temporalmente pequeñas cantidades de información necesarias en el **procesamiento de cada instrucción**.
- ✓ Finalmente, las unidades se encuentran **interconectadas** por unos elementos denominados **buses** que pueden ser de varios tipos y tienen determinados propósitos que se verán más adelante.
 - ✓ **Unidad de E/S (entrada/salida)** comunica el procesador con el resto de componentes internos del ordenador, con los periféricos de entrada y salida y los dispositivos de almacenamiento externo.

4. La unidad central de proceso (CPU)

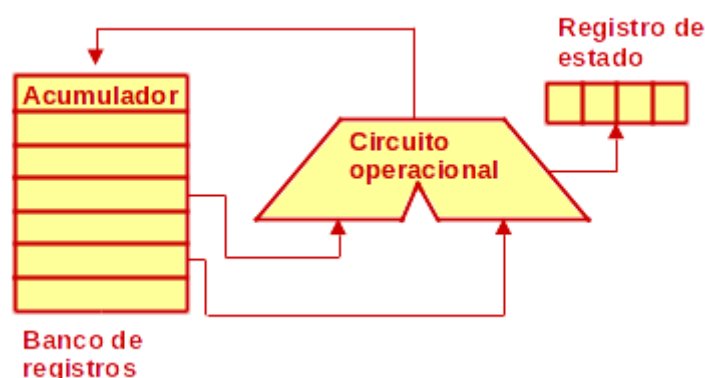
Ya hemos visto que la CPU está formada por la **ALU** y **por la unidad de control**. Ahora procedemos al estudio de cada una de estas unidades.

La unidad aritmético-lógica

La unidad aritmético-lógica o ALU se encarga de realizar las **operaciones elementales de tipo aritmético y lógico**. Los datos con los que opera deben estar previamente almacenados en sus registros. Un dato puede proceder del resultado de otra operación o bien, de una de las otras dos unidades. En el segundo caso, éste llega al registro correspondiente a través del **bus de datos**. Del mismo modo, el bus de datos se utiliza para llevar hasta la memoria (para almacenarlo) un dato contenido en sus registros.

Tanto la selección y ejecución de la operación a realizar, como los movimientos de datos mencionados, son controlados por la **unidad de control** mediante las correspondientes **señales** de control.

La siguiente figura muestra el esquema básico de la ALU.



- ✓ **Circuito operacional.** Formado por los circuitos necesarios para la realización de las operaciones con los datos procedentes del banco de registros. Hasta él llegan las líneas de control para seleccionar el tipo de operación que debe realizar.
- ✓ **Banco de registros.** El elemento indicado en la figura como banco de registros se corresponde con los registros de propósito general. Éstos sirven para que el usuario almacene temporalmente datos y resultados intermedios. El número de registros viene determinado por la arquitectura del ordenador y suele oscilar entre unos pocos (de 8 a 16) hasta varios centenares.

Habitualmente uno de estos registros, llamado **acumulador**, recibe un trato privilegiado, puesto que se utiliza como depositario del resultado de la operación y, sobre su contenido, se realizan muchas operaciones que no se pueden hacer sobre otros registros.

- ✓ **Registro de estado.** Engloba un conjunto de biestables (un **biestable** es un circuito que almacena un bit) en los que se deja constancia de condiciones que se dieron en la última operación realizada. Al registro de estado también se le conoce con el nombre de **palabra de estado**. Habitualmente, como mínimo, siempre figuran los siguientes biestables de estado, en los que se deja constancia de algunas **condiciones** que se dieron en la **última operación** realizada.
 - **Cero.** Se pone a 1 si el resultado de la operación es cero.
 - **Negativo.** Se pone a 1 si el resultado de la operación es negativo.
 - **Acarreo.** Se pone a 1 si el resultado de la operación tiene acarreo.
 - **Desbordamiento.** Se pone a 1 si el resultado de la operación tiene un acarreo en el bit **más significativo** y, en consecuencia, no se puede representar.

Entre otras cosas, el contenido del registro de estado es tenido cuenta por la unidad de control en la ejecución de instrucciones denominadas instrucciones de **salto condicional**.

La unidad de control

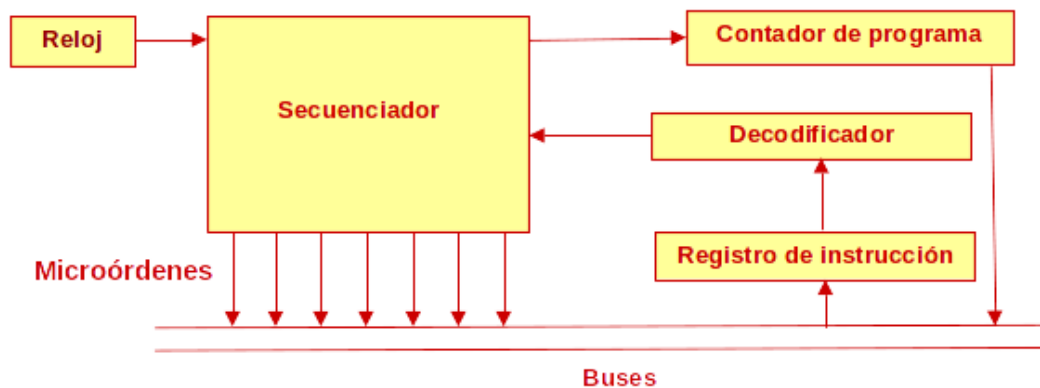
La unidad de control se encarga de controlar y coordinar el resto de unidades del sistema. Su tarea principal consiste en interpretar y ejecutar instrucciones, controlando la secuencia en que éstas se ejecutan. Esto implica:

1. **Controlar las transferencias a/desde la memoria**, ya sea para extraer la siguiente instrucción a ejecutar, para obtener los datos con los que se ha de operar o, en sentido contrario, para almacenar en memoria el contenido de algún registro.
2. Proporcionar las **temporizaciones** para que todo ocurra de forma sincronizada (en el momento debido).

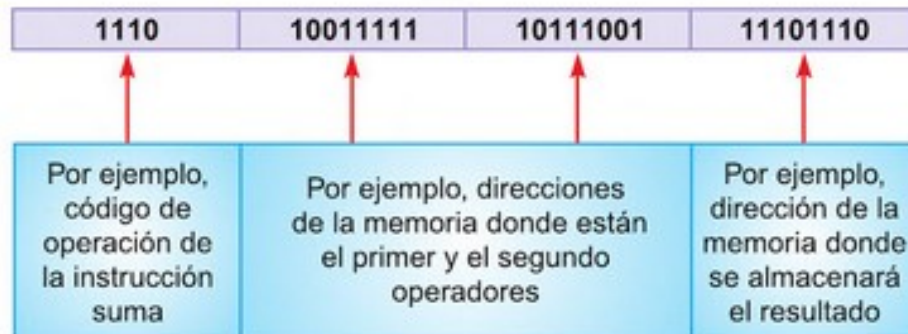
El desarrollo de su cometido se lleva a cabo mediante el envío de las oportunas señales de control a cada unidad y la recepción de señales para conocer el estado de la misma. Por ejemplo:

- ✓ Indicar a la unidad de memoria el **sentido de la transferencia**, es decir si el acceso que se hace es para **lectura** o para **escritura**.
- ✓ Si la instrucción consiste en una **operación aritmética o lógica**, las señales de control para seleccionar la operación a realizar por la **ALU**.
- ✓ **Leer** el valor del **registro de estado** tras la ejecución de una operación aritmética o lógica.

El esquema elemental de la unidad de control se muestra en la siguiente figura.



- ✓ **Contador de programa (CP).** Permite controlar la **secuencia de ejecución** de instrucciones. Contiene en cada momento la dirección de memoria donde se encuentra la instrucción siguiente a ejecutar. Al iniciar la ejecución de un programa en este registro se carga la dirección de la primera instrucción. **Su valor se incrementa una unidad de forma automática cada vez que obtiene una instrucción de memoria.** Cuando la instrucción a ejecutar sea de salto (de ruptura de secuencia de ejecución), pasará a contener la dirección de la instrucción a la que hay que saltar.
- ✓ **Registro de instrucción (RI).** Almacena temporalmente la instrucción que se ejecuta. Se transfiere a este registro desde la memoria a través del bus de datos. Las instrucciones se codifican mediante una secuencia de bits. Una parte de ese código, denominada **código de operación** identifica a la instrucción, mientras que el resto del código determina los operandos, bien directamente o, indirectamente especificando registros o dirección de memoria.



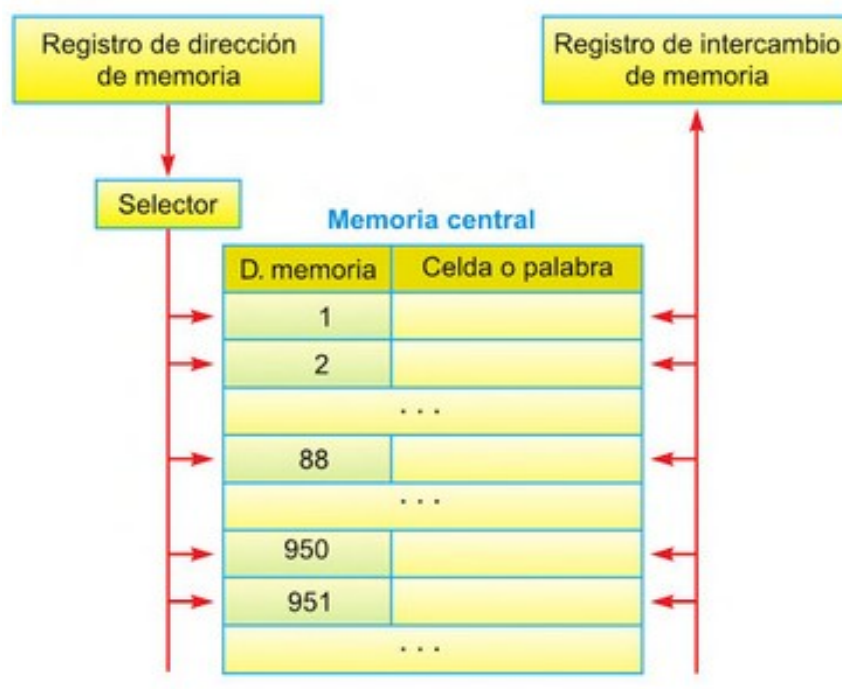
- ✓ **Decodificador de instrucciones (D).** Se encarga de interpretar la instrucción que hay almacenada en el registro de instrucción. La interpretación consiste en extraer el código de operación de la instrucción, analizarlo y generar las señales de control necesarias a través del secuenciador para su ejecución.
- ✓ **Reloj (R).** Genera una serie de **pulsos eléctricos a intervalos constantes** que marcan los instantes en que han de comenzar los distintos pasos de que consta cada instrucción.
- ✓ **Secuenciador (S).** Este elemento genera órdenes muy elementales (**microórdenes**) sincronizadas con el reloj. Se trata de las señales que hacen posible el control de las unidades implicadas en la ejecución de la instrucción.

5. La memoria

La memoria principal

Tal como hemos visto, según la estructura Von Neumann, para poder ejecutar un programa, tanto sus **instrucciones** como los **datos** a procesar deben estar **almacenados** en la memoria principal. Esta memoria está formada por multitud de celdas o posiciones (**palabras de memoria**), de un determinado número de bits, que normalmente coincide con el ancho de palabra de la arquitectura

Las celdas están numeradas de manera consecutiva, de tal forma que cada celda se identifica mediante su número, conocido como **dirección de memoria**. La dirección permite acceder de forma directa a cualquiera de las celdas con independencia de su posición. Por esta razón se dice que la memoria principal es una memoria de acceso directo o memoria accesible por dirección.



Además, la memoria principal tiene asociados dos registros para la realización de operaciones de lectura o escritura, y un dispositivo encargado de seleccionar una celda de memoria en cada operación de acceso sobre la misma:

- ✓ **Registro de dirección de memoria (RDM)**. Contiene la dirección de memoria donde se encuentran o va a ser almacenada la información (instrucción o dato), tanto si se trata de una lectura como de una escritura de o en memoria central, respectivamente.
- ✓ **Registro de intercambio de memoria (RIM)**. Si se trata de una operación de lectura, el RIM es quien recibe el dato de la memoria señalado por el RDM, para su posterior envío a uno de los registros de la ALU. Si se trata de una operación de escritura, la información a grabar tiene que estar en el RIM, para que desde él se transfiera a la posición de memoria indicada por el RDM.
- ✓ **Selector de memoria (SM)**. Es el dispositivo que, tras una orden de lectura o escritura, conecta la celda de memoria cuya dirección figure en el RDM con el RIM, posibilitando la transferencia de los datos en un sentido o en otro.

Tipos de memoria principal

En la memoria principal podemos encontrar dos tipos de memoria:

1. **Memoria RAM (*Random Access Memory* - Memoria de acceso aleatorio).** En ella se puede leer (extraer su contenido) y escribir (establecer su contenido). Esta memoria es **volátil**, puesto que al interrumpirse la alimentación eléctrica la información almacenada en ella desaparece. A su vez, en función de su tecnología de construcción, existen dos tipos de memoria RAM:
 - **SRAM o RAM estática.** Es una memoria de semiconductor basada en puntos de memoria de tipo biestable (el punto de memoria, que es el elemento tecnológico para almacenar un bit, se implementa con 6 transistores). La ventaja de esta tecnología radica en que retiene los datos mientras no se interrumpa la alimentación.
 - **DRAM o RAM dinámica.** Implementa los puntos de memoria mediante condensadores (cada punto de memoria se construye con un condensador y un transistor). Estos elementos tienen el inconveniente de que tienden a descargarse y, como consecuencia, los datos desaparecen lentamente aunque la memoria esté alimentada eléctricamente.

Para resolver el problema se hace necesario efectuar de forma periódica una operación denominada **refresco**, consistente en leer el contenido de las celdas y volver a escribirlos en la misma posición. Todas las posiciones de memoria se van refrescando de forma cíclica mientras el ordenador permanezca encendido.

La diferencia tecnológica permite que las memorias SRAM sean mucho más rápidas, pero son de menor capacidad y mayor coste económico que las DRAM. A continuación se describen brevemente los **diferentes tipos de memorias DRAM** fruto de su evolución a lo largo del tiempo hasta llegar a la actualidad:

- **SDRAM (DRAM síncrona).** A diferencia de la DRAM convencional, la memoria SDRAM se caracteriza por estar sincronizada con el reloj del sistema.
- **DDR SDRAM (DRAM síncrona de doble tasa de transferencia).** En relación a la SDRAM, esta memoria es capaz de trabajar con el doble de información en cada ciclo de reloj. Posteriormente han aparecido sucesivas versiones DDR2, DDR3 y DDR4, caracterizándose por duplicar en cada versión la tasa de información manejada.

Además de mejorar la tasa de transferencia (la cantidad de información ofrecida por cada ciclo de reloj), la evolución de la memoria DRAM ha permitido una mayor frecuencia (velocidad), y un menor voltaje de trabajo. Esto último permite moderar el incremento de calor generado con el aumento de la frecuencia de trabajo. La evolución tecnológica de la memoria DRAM ha permitido también mayores capacidades de memoria, sin embargo, este es un factor que incrementa la latencia. De tal forma que, por ejemplo, la latencia de una memoria DDR4 es considerablemente mayor que la de una memoria DDR3.

Se denominan **latencias** de una memoria RAM a los diferentes retardos producidos en el **acceso a los distintos componentes** de esta última. Estos retardos influyen en el tiempo de acceso de la memoria por parte de la CPU

La **latencia** hace referencia a tiempos necesarios (**retardos**) para acceder a los diferentes componentes de la memoria RAM. Comercialmente se suele tomar como referencia la **latencia CAS (CL)** que hace **referencia a los ciclos de reloj que transcurren desde que se da la orden de lectura hasta que se obtienen los datos**.

2. **Memoria ROM (*Read Only Memory* - Memoria de sólo lectura).** Se trata de una memoria permanente (no volátil) que no permite realizar operaciones de escritura. Por la forma de almacenar

la información es más lenta que la memoria RAM, aunque esta diferencia puede variar según el tipo de memoria ROM. A continuación se describen brevemente los tipos de memoria ROM que se han ido utilizando a lo largo del tiempo:

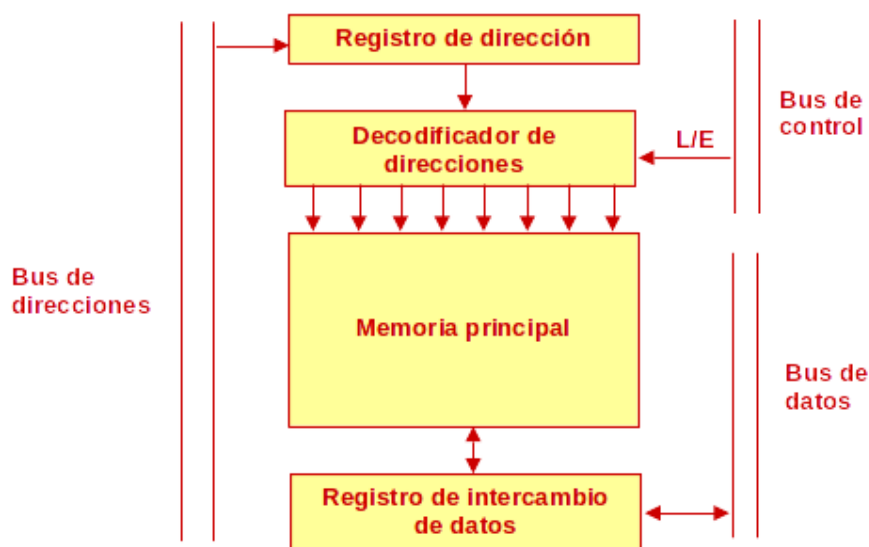
- **ROM**. El contenido de los chips de memoria ROM es grabado en el momento de su fabricación y no puede modificarse.
- **PROM (*Programmable ROM*)**. Los chips de memoria PROM salen de fábrica sin contenidos y pueden grabarse una única vez con un dispositivo adecuado.
- **EPROM (*Erasable Programmable ROM*)**. Pueden ser borradas y grabadas más de una vez utilizando un dispositivo adecuado, basado en luz ultravioleta.
- **EEPROM (*Electrically Erasable Programmable ROM*)**. Puede ser borrada y reprogramada eléctricamente mediante un proceso controlado por software que no requiere de dispositivos especiales. Este tipo de ROM más flexible que las anteriores se utiliza actualmente para guardar la BIOS de las placas base.

La BIOS es realmente parte del **firmware** del ordenador. Entre otras cosas se encarga de controlar el ordenador desde el momento en que arranca hasta que el sistema operativo toma el control. En las primeras fases del proceso de arranque se encarga de hacer un **chequeo** del hardware, de tal forma que si detecta alguna anomalía, informa al usuario mediante un **código de pitidos**, que puede variar de una BIOS a otra.

La configuración de la BIOS que establece el usuario se guarda en una memoria denominada **CMOS**. Como la CMOS es **volátil**, se utiliza una **pila o batería**, que le suministra la alimentación eléctrica necesaria para mantener la información que contiene.

Esquema básico de una unidad de memoria

Una unidad de memoria implementa las funciones de almacenamiento (escritura) y recuperación (lectura) del contenido de las celdas. El esquema básico de una unidad de memoria se muestra en la figura siguiente.



- ✓ **Registro de dirección de memoria.** Contiene la dirección de la celda que va a intervenir en la operación a realizar (lectura o escritura). La dirección llega a este registro a través del bus de direcciones.
- ✓ **Decodificador de dirección o selector de memoria.** Es responsable de establecer la conexión entre la celda de memoria cuya dirección figura en el registro de dirección de memoria con el registro de intercambio de datos, a fin de que la transferencia tenga lugar en un sentido u otro. Se activa con la señal de control L/E procedente de la unidad de control. Esta señal determina el sentido de la transferencia.
- ✓ **Registro de intercambio de datos.** En él se almacena el dato (o instrucción) que se ha leído de memoria o el dato que se va a escribir en ella.
- ✓ **Líneas de control.** Permiten transmitir órdenes procedentes de la unidad de control (p. e. señal de lectura/escritura).

Secuencia de pasos para leer/escribir en una posición de memoria

Operación de lectura

1. La dirección de la celda a leer **llega al registro de dirección** de memoria por el bus de direcciones.
2. Se **decodifica la dirección**. Es decir, el decodificador establece un camino entre la celda correspondiente y el registro de intercambio de datos.
3. Después de un tiempo determinado (**tiempo de acceso a memoria**), el contenido de la celda pasa al **registro de intercambio de datos** y desde éste es enviado a su destino (un registro de la CPU) por el bus de datos.

Operación de escritura

1. La dirección en la que se va a escribir se transfiere al **registro de dirección de memoria** a través del bus de direcciones.
2. El dato (o instrucción) a escribir se transfiere el dato al registro de intercambio de datos a través del bus de datos.
3. Se **decodifica la dirección**.
4. El contenido del registro de intercambio de datos pasa a la celda correspondiente.

Jerarquía de memoria

Memoria virtual

Para que un programa pueda ser ejecutado, debe encontrarse en memoria principal. Sin embargo, hay situaciones en que la cantidad de programas que se están ejecutando es tal, que su tamaño excede el tamaño de la memoria disponible. En estos casos se utiliza la técnica denominada **memoria virtual**, que consiste en guardar el programa y sus datos, en memoria masiva (externa o auxiliar) y mantener en memoria principal únicamente la parte de ellos que está implicada en ese momento en la ejecución. Pero los dispositivos de memoria externa son mucho más lentos que la memoria principal.

Memoria caché

Por otra parte, la CPU capta instrucciones y datos de la memoria principal, almacenando en ella los resultados de las operaciones. Sin embargo, con los años, la evolución de la tecnología ha dado lugar a una diferencia de velocidades de operación cada vez mayor entre ambas unidades, de tal forma que la memoria principal se ha convertido en un elemento ralentizador del rendimiento de la CPU. La solución a este problema de cuello de botella ha consistido en interponer entre la memoria principal y los registros del procesador otro tipo de memoria denominada **memoria caché**.

La caché se estructura en varios niveles, normalmente tres, denominados L1, L2, L3. L1 está más cerca de la CPU, es más rápida y de menor capacidad. L2 es más lenta que L1 pero de mayor capacidad y, finalmente, L3 es más lenta que L2 y a su vez, de mayor capacidad. En la actualidad, el primer nivel de caché está formado por dos memorias, una para datos y otra para instrucciones.

En sus primeros momentos de desarrollo tecnológico, la memoria caché, de un único nivel se colocaba en la placa base, fuera del procesador. Con el tiempo se usaban **dos cachés** (L1 y L2), de tal forma que una estaba **en la placa y otra en el procesador**. La tendencia es que los tres niveles que se usan actualmente se incluyan en el procesador.

Jerarquía de memoria

Lo ideal sería tener siempre una memoria homogénea que sea muy rápida, de gran capacidad y a ser posible barata (mínimo coste por bit), pero estos parámetros son contrapuestos. Entre estas tres características se dan las siguientes relaciones:

- A menor tiempo de acceso → mayor coste por bit.
- A mayor capacidad → mayor tiempo de acceso.
- A mayor capacidad → menor coste por bit.

La solución pasa por **estructurar la memoria** en varios niveles. Se crea una jerarquía con un **nivel superior muy rápido, de pequeña capacidad**, y niveles sucesivos, más alejados de la CPU, de **menor velocidad, pero con mayor capacidad**. La información se ubica en uno de los niveles, de acuerdo a su probabilidad de uso, de tal forma que, la menos usada, se encuentra en niveles inferiores y de mayor capacidad. La información que necesita el procesador en un momento determinado se transfiere a los niveles superiores.

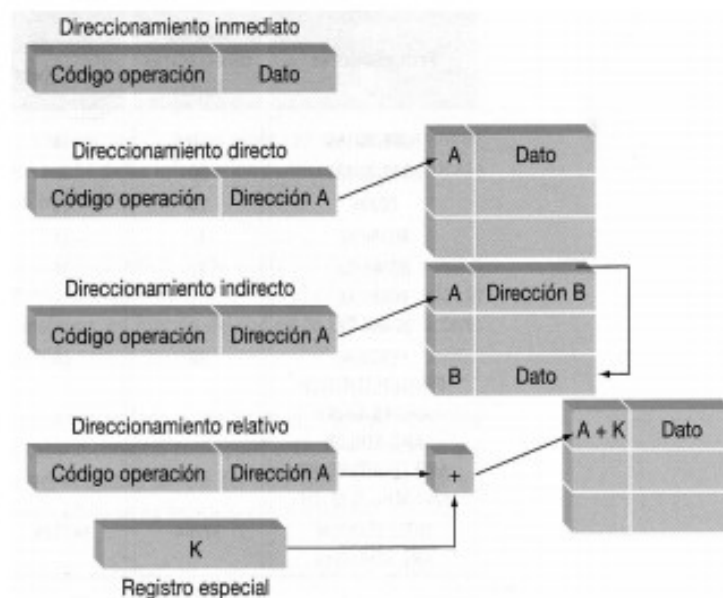
JERARQUÍA DE MEMORIA	
MEMORIA INTERNA	Registros
	Caché L1
	Caché L2
	Caché L3
	Memoria principal
MEMORIA EXTERNA	Discos duros / SSD (almacenamiento masivo interno)
	Almacenamiento masivo externo

La **memoria virtual** se implementa sobre disco duro, con lo cual, los **dispositivos de los dos niveles más bajos no intervienen para nada en la ejecución de un programa**, salvo en el momento de su carga en memoria para su ejecución. Se han incluido en la jerarquía puesto que su finalidad es almacenar información de manera permanente, como medios no volátiles que son.

Direccionamiento (acceso a una dirección de memoria)

El direccionamiento es una operación que se realiza cuando el procesador ejecuta o interpreta una instrucción. El modo de direccionamiento utilizado afecta directamente a la rapidez de ejecución de un programa. Para acceder a una dirección de memoria se pueden utilizar diferentes modos de direccionamiento:

- ✓ **Direccionamiento inmediato.** Se produce cuando las instrucciones contienen dentro sus propios datos, de modo que no se necesita acceder a la memoria para leerlo.
- ✓ **Direccionamiento directo.** Se produce cuando expresa la dirección real del objeto. Así la dirección de memoria 12000 se corresponde con la posición 12000 de memoria.
- ✓ **Direccionamiento indirecto.** Se produce cuando la dirección obtenida no es el objeto deseado, sino su dirección. Por tanto, para obtener el objeto deseado se requiere un acceso adicional a la memoria.
- ✓ **Direccionamiento relativo.** El direccionamiento se llama relativo cuando la dirección del dato que interviene en la instrucción se obtiene sumando a la dirección de la propia instrucción una cantidad fija, que normalmente está contenida en un registro de tipo especial.



6. Los buses

Los **elementos de conexión de unidades funcionales** reciben el nombre de buses. Un bus no es más que un **conjunto de líneas o caminos conductores de electricidad**. Su propósito es transmitir bits codificados de alguna manera con el uso de magnitudes eléctricas.

Los buses se clasifican en:

- ✓ **Bus de datos.** Transporta los datos que se transfieren entre las unidades. Aunque no siempre ha sido así, el número de líneas de este bus coincide con la cantidad de bits de los operandos de las operaciones aritméticas y lógicas, es decir, con el ancho de palabra de la arquitectura. Es un bus bidireccional, puesto que los mismos hilos se utilizan para transmitir información hacia adentro o hacia afuera de una unidad, pero siempre en instantes diferentes.

- ✓ **Bus de direcciones.** Conecta la unidad de control con la unidad de memoria principal. Este bus unidireccional permite que la unidad de control indique la posición (dirección) de la celda de memoria cuyo contenido (dato o instrucción) desea obtener (leer), o en la que va a escribir un dato. El tamaño de este bus define la cantidad máxima de celdas (posiciones) de memoria RAM que la CPU puede gestionar. Por ejemplo:
 - Con 10 bits → 2^{10} posiciones → 1K posiciones → 1.024 posiciones.
 - Con 16 bits → 2^{16} posiciones → 64K posiciones → 65.536 posiciones.
 - Con 32 bits → 2^{32} posiciones → 4G posiciones → 4.294.967.296 posiciones.
- ✓ **Bus de control.** Contiene líneas que transportan las señales de control y las señales de estado entre la unidad de control y el resto de unidades. Algunas de ellas se mencionan en los siguientes apartados.

7. Los periféricos

En informática, se denomina periféricos a los **aparatos o dispositivos auxiliares e independientes conectados a la unidad central de procesamiento** de una computadora. Se consideran periféricos tanto a las unidades o dispositivos a través de los cuales la computadora se comunica con el mundo exterior, como a los sistemas que almacenan o archivan la información, sirviendo de memoria auxiliar.

A pesar de que el término periférico suele implicar el concepto de “adicional pero no esencial”, muchos de ellos son elementos fundamentales para un sistema informático. El teclado y el monitor son posiblemente los periféricos más comunes. El ratón es posiblemente el ejemplo más claro de este aspecto, ya que hace menos de 20 años no todos los ordenadores personales incluían este dispositivo. El sistema operativo MS-DOS, el más común en esa época, tenía una interfaz de línea de comandos para la que no era necesaria el empleo de un ratón, todo se hacía mediante comandos de texto.

Los periféricos pueden clasificarse en cinco categorías principales:

- ✓ **Periféricos de entrada:** captan y digitalizan los datos de ser necesario, introducidos por el usuario o por otro dispositivo y los envían al ordenador para ser procesados.
- ✓ **Periféricos de salida:** son dispositivos que muestran o proyectan información hacia el exterior del ordenador. La mayoría son para informar, alertar, comunicar, proyectar o dar al usuario cierta información, de la misma forma se encargan de convertir los impulsos eléctricos en información legible para el usuario.
- ✓ **Periféricos de entrada/salida (E/S):** sirven básicamente para la comunicación de la computadora con el medio externo. Los periféricos de entrada/salida son los que utiliza el ordenador tanto para mandar como para recibir información. Su función es la de almacenar o guardar, de forma permanente o virtual, todo aquello que hagamos con el ordenador para que pueda ser utilizado por los usuarios u otros sistemas.
- ✓ **Periféricos de almacenamiento:** son los dispositivos que almacenan datos e información por bastante tiempo. La memoria de acceso aleatorio no puede ser considerada un periférico de almacenamiento, ya que su memoria es volátil y temporal.
- ✓ **Periféricos de comunicación:** son los periféricos que se encargan de comunicarse con otras máquinas o computadoras, ya sea para trabajar en conjunto, o para enviar y recibir información.

8. Seguridad de la información

Una de las principales funciones de un administrador de sistemas es mantener la información del sistema **accesible** y **segura**. La información debe llegar fácilmente a algunos usuarios, y sin embargo debe ser inaccesible para otros. Esto implica que debemos tomar medidas para que la información no se pierda o corrompa. La seguridad además se puede centrar en varios aspectos.

Seguridad física

La seguridad física suministra protección ante **accesos no autorizados**, **daños** e **interferencias a las instalaciones de la organización y a la información**.

Los requisitos sobre seguridad física varían considerablemente según las organizaciones y dependen de la escala y de la organización de los sistemas de información. Pero son aplicables a nivel general los conceptos de asegurar áreas, controlar perímetros, controlar las entradas físicas e implantar equipamientos de seguridad. Las líneas de actuación recomendadas son:

- ✓ **Adecuación de locales donde se sitúan los servidores:** definir de forma proporcionada las medidas que garanticen la seguridad de las áreas a proteger en relación con los requisitos de seguridad de la información que se almacene o procese. Hay que proteger el acceso no autorizado de personal, tomar medidas contra **amenazas potenciales** como fuego, agua, temperaturas extremas, etc.
- ✓ **Adecuación de las líneas de telecomunicaciones:** considerar medidas para **proteger los cables** de líneas de datos contra escuchas no autorizadas o contra daños (por ejemplo, evitando rutas a través de áreas públicas o fácilmente accesibles). Esto es muy importante en casos de líneas de telecomunicación inalámbricas o Wifi.
- ✓ **Adecuación de las líneas eléctricas:** se debe instalar al menos en la **ubicación de los servidores**, un sistema de alimentación ininterrumpida (SAI) de modo que el sistema pueda permanecer en funcionamiento en caso de avería eléctrica, al menos el tiempo necesario para sacar una copia de seguridad de emergencia y echar abajo el sistema de forma controlada.

En los últimos años se ha creado un nuevo riesgo que está afectando gravemente a los sistemas informáticos. Normalmente el administrador, o al menos un buen administrador, va a blindar su sistema del exterior, con **cortafuegos**, **antivirus**, etc. Eso ha hecho que muchos ataques no se produzcan desde fuera, sino desde dentro, para ello el atacante se limita a intentar controlar un ordenador portátil de cualquier empleado cuando lo use en su casa. Una vez que dicho usuario vuelva a la empresa y conecte su portátil en red, tendremos al atacante dentro de nuestro sistema, donde las medidas de seguridad suelen ser mucho menores.

Seguridad lógica

La seguridad lógica se ocupa de los **riesgos que sufre la información del sistema**, no promovida por fallos de hardware o robo de material. Las líneas de actuación recomendadas son:

- ✓ **Protección antivirus.** En la actualidad, existen cientos de tipos de **virus** y **gusanos** que atacan por la red. Los virus y gusanos son programas que tienen la función de propagarse, infectando equipos indiscriminadamente usando deficiencias del software para instalarse sin el conocimiento del usuario. Es vital hoy en día instalar en el sistema un **software** antivirus y mantenerlo **actualizado**, y en el caso de sistemas informáticos controlados por servidor, es conveniente instalar un producto que proteja de virus toda la red, de forma corporativa. También es conveniente instalar entre nuestra red local interna y la red Internet, un cortafuego que sirva para impedir la entrada de la mayoría de gusanos y virus.

- ✓ **Protección contra el *malware*.** Dentro del término *malware* incluimos varios programas, muy parecidos a los virus y gusanos pero que tienen la diferencia de ser legales. Esto implica que incluido en un programa legal que instalemos en el sistema, puede venir incluido un software de este tipo, lo que hace inútiles la mayoría de antivirus. Existen programas que se encargan específicamente de buscar en nuestro sistema software de este tipo y eliminarlo. Este software debe ser actualizado aun con mayor frecuencia que el antivirus.
- ✓ **Protección contra errores de software.** Es imposible crear un programa de software de cierta magnitud que no incluya fallos. No hablamos normalmente de errores que impliquen que el software no funcione (que también) sino que nos referimos a que bajo ciertas circunstancias, el software puede dejar de funcionar, dar un error, quedarse colgado, etc. Algunos de estos errores pueden ser aprovechados por otros programas, para tomar el control del ordenador, en este caso estos errores de software suelen ser denominados **vulnerabilidades**. Estas vulnerabilidades suelen ser corregidas por los desarrolladores de software mediante **parches**, que hay que instalar en el sistema.
- ✓ **Copias de respaldo y material redundante.** Es obligatorio la creación de una política de **copias de seguridad** usando varios soportes de forma periódica. También se debe instalar hardware redundante en los equipos servidores, de modo que la avería fortuita de un hardware no afecte al sistema.
 - **Discos duros redundantes.** Se instalan en el sistema varios discos duros funcionando al mismo tiempo, de modo que si uno se estropea, los otros son capaces de seguir funcionando sin pérdida de información ni tiempo. Estos sistemas de discos duros redundantes se conocen como RAID.
 - **Fuentes de alimentación redundantes.** Las cajas de ordenador para servidores suelen incluir varias fuentes de alimentación, de modo que si una se estropea, las otras mantienen el equipo en funcionamiento.
 - **Tarjetas de red redundantes.** Instalar varias tarjetas de red en un equipo permite que ante la avería de una de ellas, o del cableado de la misma, las otras tarjetas usen sus propios cables para seguir alimentando el sistema de datos.

Seguridad de los datos

Aquí tratamos de los riesgos de seguridad que nos podemos encontrar, que van a **afectar a los datos**, a la información de nuestro sistema, pero que no vienen promovidos ni por un error en un dispositivo físico, ni por un programa y son normalmente de los propios usuarios de nuestro sistema.

Cualquier usuario de nuestro sistema, si le damos el derecho a modificar datos, puede equivocarse y modificar los datos mal, puede borrar un fichero sin darse cuenta, cambiarle el nombre, perderlo por en el disco duro, y cosas peores. Las medidas de actuación para minimizar estos riesgos, entre otras muchas, son:

- ✓ **Formación del personal.** El personal debe conocer lo que está haciendo, al menos de una forma sucinta. El administrador debe preocuparse de que se oferten cursillos, de realizar manuales, etc.
- ✓ **Copias de seguridad.** Las vimos en la seguridad lógica, pero aquí son importantes no ya por si falla un disco duro, sino por que un usuario puede borrar sus propios datos o los de los compañeros.
- ✓ **Política de restricción de derechos.** Cada usuario debe acceder sólo a la parte de la información que necesita, y sólo con los permisos que necesita. Si solo debe leer los datos de la facturación semanal, no tiene sentido que dejemos que los modifique.
- ✓ **Ingeniería social.** Con este punto nos referimos a todo lo relacionado con el conocimiento del ser humano.