

Sistemas Operativos I

Módulo Introductorio

CONCEPTOS BÁSICOS NECESARIOS

1

Temas del Módulo Introductorio

- **Definiciones y convenciones de notación.**
 - Unidades de cantidad de información.
 - Tiempo y Velocidad de transmisión de datos.
 - Frecuencia y Período.
- **Fundamentos de hardware de una computadora.**
 - CPU.
 - Memoria principal.
 - Subsistema de Entrada/Salida.
 - Buses.
- **Jerarquía de memoria.**
 - Diseño y componentes.
 - Características: tiempo de acceso y capacidad de almacenamiento.
 - Principio de cercanía de referencias.
- **Comunicación entre componentes de una computadora.**
 - Entrada/Salida programada. Polling.
 - Interrupciones.
 - DMA (Direct Memory Access).

2

Definiciones y convenciones de notación.

Unidades de cantidad de información:

Bit (b): **B**inary **D**igit. Es la unidad elemental de información.

Byte (B): conjunto de bits, dependiendo de la arquitectura, pueden ser de 4 a 12 bits.

Actualmente se lo considera de **8 bits**. También llamado **octeto** para evitar ambigüedades.

<https://en.wikipedia.org/wiki/Byte>
<https://es.wikipedia.org/wiki/Byte>

Múltiplos de las unidades de información: (IEC 80000-13)

Kilobit (Kb): $1.000 \text{ b} - 10^3 \text{ b}$ Kilobyte (KB): $1000 \text{ B} - 10^3 \text{ B}$

Megabit (Mb): $1.000.000 \text{ b} - 10^6 \text{ b}$ Megabyte (MB): 10^6 B

Gigabit (Gb): 10^9 b Gigabyte (GB): 10^9 B

Terabit (Tb): 10^{12} b Terabyte (TB): 10^{12} B

3

Definiciones y convenciones de notación.

Unidades de cantidad de información: (cont.)

Múltiplos de las unidades de información: (IEC 80000-13)

Kibibit (Kib): $1024 \text{ b} - 2^{10} \text{ b}$ Kibibyte (KiB): $1024 \text{ B} - 2^{10} \text{ B}$

Mebibit (Mib): $1024^2 \text{ b} - 2^{20} \text{ b}$ Mebibyte (MiB): $1024 \text{ KiB} - 2^{20} \text{ B}$

Gibibit (Gib): $1024^3 \text{ b} - 2^{30} \text{ b}$ Gibibyte (GiB): $1024 \text{ MiB} - 2^{30} \text{ B}$

Tebibit (Tib): $1024^4 \text{ b} - 2^{40} \text{ b}$ Tebibyte (TiB): $1024 \text{ GiB} - 2^{40} \text{ B}$

4

Definiciones y convenciones de notación.

Unidades de cantidad de información: (*cont.*)

Palabra (wd) (*word*, en inglés): cadena de bits tratados en conjunto.

Normalmente, los conjuntos son grupos de Bytes, y según la arquitectura de hardware, en números pares y potencia de 2.

En las arquitecturas modernas, la palabra tiene 64 bits.

[https://en.wikipedia.org/wiki/Word_\(computer_architecture\)](https://en.wikipedia.org/wiki/Word_(computer_architecture))
[https://es.wikipedia.org/wiki/Palabra_\(informática\)](https://es.wikipedia.org/wiki/Palabra_(informática))

5

Definiciones y convenciones de notación.

Unidad de tiempo (t):

Segundo (s): duración de 9.192.631.770 oscilaciones de la radiación emitida en la transición entre los dos niveles hiperfinos del estado fundamental del isótopo 133 del átomo de cesio (¹³³Cs), a una temperatura de 0°K.

Submúltiplos de la unidad de tiempo:

Milisegundo (ms): 10^{-3} s.

Microsegundo (μs): 10^{-6} s.

Nanosegundo (ns): 10^{-9} s.

Picosegundo (ps): 10^{-12} s.

6

Definiciones y convenciones de notación.

Velocidad de transmisión de datos:

Cantidad de unidades de información por unidad de tiempo.

Unidad de velocidad de transmisión de datos:

Bit por segundo (bps – b/s) ¿Y el B/s?

Múltiplos de la unidad de velocidad de transmisión:

Kilobit por segundo (Kbps): 10^3 bps - 1.000 bps

Megabit por segundo (Mbps): 10^6 bps - 1.000.000 bps

Gigabit por segundo (Gbps): 10^9 bps - 1.000.000.000 bps

Terabit por segundo (Tbps): 10^{12} bps

7

Definiciones y convenciones de notación.

Frecuencia (f):

Magnitud que mide el número de repeticiones por unidad de tiempo de cualquier fenómeno o suceso periódico.

Unidad de frecuencia:

Hertz (Hz): ciclos por segundo.

Múltiplos de la unidad de frecuencia:

Kilohertz (KHz): 10^3 Hz.

Megahertz (MHz): 10^6 Hz.

Gigahertz (GHz): 10^9 Hz.

Terahertz (THz): 10^{12} Hz.

		Longitud de onda	Frecuencia	Energía
Radio	Muy Baja frecuencia	≥ 10 km	≤ 30 kHz	$\leq 1.99 \cdot 10^{-29}$ J
	Onda larga	≤ 10 km	≥ 30 kHz	$\geq 1.99 \cdot 10^{-29}$ J
	Onda media	≤ 650 m	≥ 650 kHz	$\geq 4.31 \cdot 10^{-26}$ J
	Onda corta	≤ 180 m	≥ 1.7 MHz	$\geq 1.13 \cdot 10^{-27}$ J
	Muy alta frecuencia	≤ 10 m	≥ 30 MHz	$\geq 2.05 \cdot 10^{-26}$ J
Microondas	Ultra alta frecuencia	≤ 1 m	≥ 300 MHz	$\geq 1.99 \cdot 10^{-25}$ J
		≤ 30 cm	≥ 1.0 GHz	$\geq 1.99 \cdot 10^{-24}$ J
Infrarrojo	Lejano / submilimétrico	≤ 1 mm	≥ 300 GHz	$\geq 199 \cdot 10^{-24}$ J
	Medio	≤ 50 μ m	≥ 6.0 THz	$\geq 3.98 \cdot 10^{-21}$ J
	Cercano	≤ 2.5 μ m	≥ 120 THz	$\geq 79.5 \cdot 10^{-21}$ J
Luz Visible		≤ 780 nm	≥ 384 THz	$\geq 255 \cdot 10^{-21}$ J
		≤ 380 nm	≥ 789 THz	$\geq 523 \cdot 10^{-21}$ J
Ultravioleta	Extremo	≤ 200 nm	≥ 1.5 PHz	$\geq 993 \cdot 10^{-21}$ J
Rayo X		≤ 10 nm	≥ 30.0 PHz	$\geq 19.9 \cdot 10^{-18}$ J
Rayos Gamma		≤ 10 pm	≥ 30.0 EHz	$\geq 19.9 \cdot 10^{-15}$ J

8

Definiciones y convenciones de notación.

Período (T): duración de un ciclo de reloj.

$$T = \frac{1}{f}$$

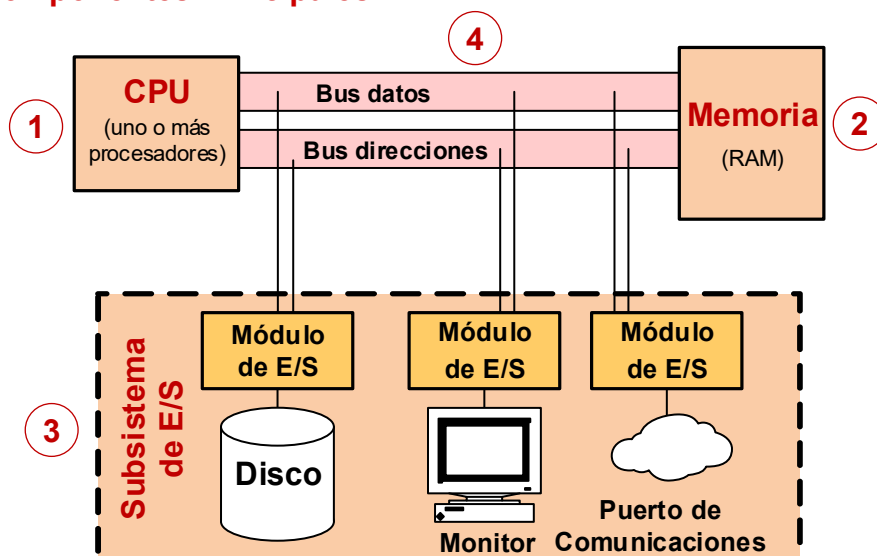
Equivalencia entre frecuencia y velocidad:

$$1\text{Hz} \equiv 1\text{bps}$$
$$1\text{KHz} \equiv 1\text{Kbps}$$

9

Fundamentos de HW de una computadora.

Componentes Principales:



10

Fundamentos de HW de una computadora.

1. CPU (Unidad Central de Procesamiento).

Puede estar conformado por uno o más procesadores. Es el dispositivo que **controla la operación del computador** y **procesa los datos**.

Contiene registros internos que se utilizan para almacenar temporalmente instrucciones u operandos.

Estos registros se pueden clasificar en:

- **Registros Visibles al Usuario:** están disponibles para ser usados por los **programadores** para diversas tareas. Ej.: cargar datos o instrucciones para minimizar las referencias a memoria principal y, así, reducir tiempos de procesamiento.
- **Registros de Control y Registros de Estado:** **No pueden ser accedidos por lo usuarios**. Sólo pueden ser utilizados por la lógica interna del **procesador** (para control de las operaciones que realiza), o por **subrutinas privilegiadas del SO** para controlar la ejecución de los programas.

11

Fundamentos de HW de una computadora.

1. CPU (Unidad Central de Procesamiento). (cont.)

Registros Visibles al Usuario.

Se pueden acceder con instrucciones de alto y bajo nivel. Se dividen en:

- **Registros de Datos:** se usan para **almacenar datos en general**. Algunos registros pueden sólo usarse para datos especiales como, por ejemplo, para operaciones con coma flotante.
- **Registros de Dirección:** pueden ser de uso general o para un uso específico de direccionamiento. Para uso específico los más comunes son:
 - **Registro Índice (IR):** para direccionamiento indexado.
 - **Puntero de Segmento:** se usa cuando la memoria se divide en segmentos. Indica un desplazamiento de un segmento de memoria.
 - **Puntero de Pila:** apunta al comienzo de una pila.
- **Registros de Código de Condición:** El código de condición es un conjunto de bits indicadores, banderas, que **indican alguna característica del resultado de una operación**. Ej.: resultado positivo, negativo, cero, desbordamiento, etc. Normalmente estos registros de código de condición sí **pueden ser leídos** por el programador **pero no pueden ser modificados**.

12

Fundamentos de HW de una computadora.

1. CPU (Unidad Central de Procesamiento). (cont.)

Registros de Control y Registros de Estado.

Se emplean para controlar operaciones del procesador. No pueden ser accedidos por los usuarios y, en la mayoría de los casos, no son visibles. Sólo pueden ser accedidos por las subrutinas del SO. Ej.: en cambios de procesos. Los registros más típicos son:

- **Program Counter (PC):** Contiene la dirección de la próxima instrucción a ser referenciada y leída por el procesador.
- **Instruction Register (IR):** Contiene el código de la última instrucción leída.
- **Program Status Word (PSW):** Contiene información del estado actual del procesamiento. Los campos o bits más típicos son los indicadores de:
 - Signo.
 - Cero.
 - Acarreo.
 - Igualdad.
 - Overflow.
 - Habilitar/inhabilitar interrupciones, etc.

13

Fundamentos de HW de una computadora.

2. Memoria Principal.

Almacena **datos y programas**. Es normalmente volátil. También se la denomina **memoria principal**, **memoria física**, **memoria real**, **RAM (memoria de acceso aleatorio)**.

La **memoria** es el dispositivo donde se carga **parte** o **todo** de:

- Sistema Operativo (en adelante, SO).
- Programas de usuarios.
- Datos que va a utilizar el procesador en la ejecución de un programa.

14

Fundamentos de HW de una computadora.

3. Subsistema de Entrada/Salida¹.

Se divide en dos subsistemas:

- Subsistema de almacenamiento².
- Subsistema de comunicaciones.

Ambos subsistemas están compuestos por un conjunto de:

- Módulos de E/S o Controladores.
- Dispositivos periféricos.

Módulo de E/S:

- Transporta datos entre la memoria del computador y los dispositivos periféricos.

Dispositivos periféricos:

- Discos.
- Equipos de comunicación.
- Monitor, teclado, etc.

¹ En adelante, E/S.

² También llamado "de almacenamiento secundario".

15

Fundamentos de HW de una computadora.

3. Subsistema de E/S. (cont.)

Cada módulo de E/S realiza las siguientes tareas:

- "Dialoga" e intercambia datos con el procesador y con la memoria a través del bus del sistema.
- Maneja y controla el dispositivo en el cual lee o escribe datos.

Cada uno de ellos contiene:

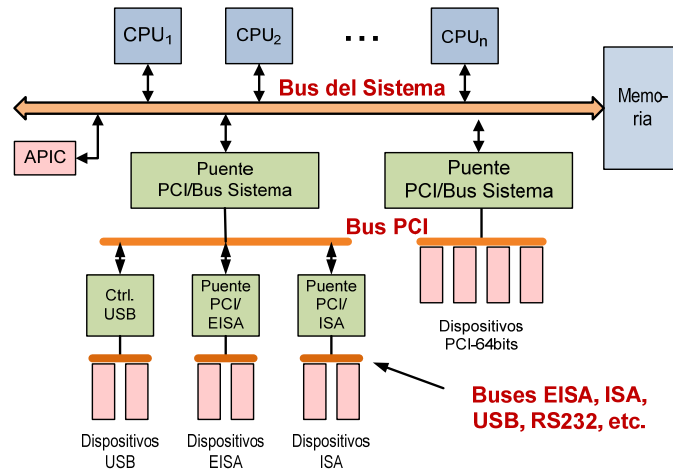
- Un procesador.
- Memoria ROM con código almacenado.
- Registro de estado.
- Buffer.
- Electrónica para el manejo del dispositivo que controla.

16

Fundamentos de HW de una computadora.

4. Buses.

Son la vía de conexión entre procesadores, memoria y módulos de E/S. También entre los módulos de E/S y los periféricos.



17

Jerarquía de memoria.

Diseño de la jerarquía de memoria.

- Al ser la memoria el dispositivo donde se almacena el SO, los programas y sus datos, lo ideal es contar con una gran cantidad, con alta velocidad de acceso y de bajo costo.
- La **cantidad** de memoria está relacionada con el espacio que ocupará el SO, las aplicaciones que van a ejecutar y sus datos.
- La **velocidad** de acceso a la memoria debe seguir el ritmo del procesador, para que éste no tenga que esperar la llegada de los datos almacenados en ella.
- El **costo** de la memoria debe ser razonable, en relación a otros componentes del computador.
- Sin embargo, la realidad es que la **cantidad**, la **velocidad** y **costo compiten entre sí**.

18

Jerarquía de memoria.

Diseño de la jerarquía de memoria. (cont.)

- Las tecnologías de fabricación, además, imponen ciertas limitaciones:
- A menor tiempo de acceso, mayor costo por bit.
- A mayor cantidad, menor costo por bit.
- A mayor cantidad, mayor tiempo de acceso.
- La salida del dilema anterior consiste en no depender de un único componente o de una única tecnología de memoria, sino emplear una **jerarquía de la memoria**.
- Esto significa, combinar distintos dispositivos de memoria, cada uno con características distintas de cantidad, velocidad y costo.

19

Jerarquía de memoria.

Componentes de la jerarquía de memoria.

- La cantidad total de memoria de una computadora es la suma de los tamaños de:
 - Registros el procesador.
 - Cache.
 - Memoria principal.
 - Almacenamiento secundario.
- La **jerarquía de memoria es la manera en que se organizan estos elementos**, según sus capacidades de almacenamiento, su velocidad de acceso y su costo relativo.
- El objetivo de esta organización es **optimizar los accesos a memoria por parte del procesador**, en cuanto **minimizar los tiempos de acceso totales**, lograr **la máxima cantidad posible** y al **costo más razonable**.

20

Jerarquía de memoria.

Características de los componentes.

Tiempo de acceso típico

Capacidad típica

1 nseg	Registros	< 1 KB
2 nseg	Cache	4 MB
10 nseg	Memoria principal	1 – 16 GB
10 mseg	Disco magnético	500 GB – 4 TB
100 seg	Cinta magnética	2 GB – 4 TB

- En esta figura, a medida que descendemos por la pirámide:
 - Disminuye el costo por bit.
 - Aumenta la capacidad.
 - Aumenta el tiempo de acceso.
 - Disminuye la frecuencia de acceso a la memoria.

21

Jerarquía de memoria.

Principio de cercanía de referencias.

La clave está en utilizar adecuadamente la disminución de la frecuencia de acceso a memoria.

Con una frecuencia de acceso baja, se puede obtener un buen rendimiento utilizando 2 o más niveles de memoria:

- Nivel 1: memoria de baja capacidad, alto costo por bit y rápida.
- Nivel 2: memoria de gran capacidad, bajo costo por bit y lenta.

Si la palabra a acceder está en el nivel 1, entonces es leída directamente por el procesador. A esto le llamaremos **acierto**.

Si la palabra **no está** en el nivel 1 y **está en el nivel 2**, entonces el procesador debe primero traer la palabra al nivel 1 y recién después puede accederla. A esto le llamaremos **fallo**. En este caso, el tiempo de acceso será la suma de ambos niveles.

22

Jerarquía de memoria.

Principio de cercanía de referencias. (cont.)

Para tratar de eliminar o amortiguar el efecto negativo de la lentitud es necesario disminuir la frecuencia de acceso a las memorias lentas. Para eso, es necesario que en cada acceso a una memoria lenta se traigan muchos datos a memorias más rápidas.

Entonces, surge una pregunta: ¿Qué datos deben traerse?

Principio de cercanía de referencias.

Durante la ejecución de un programa, las referencias a memoria (tanto para instrucciones como para datos) están generalmente cercanas. Esto sucede porque los programas contienen lazos y subrutinas iterativas que hacen que se produzcan referencias repetidas a un pequeño conjunto de instrucciones y datos que está cercanos entre sí.

23

Jerarquía de memoria.

Principio de cercanía de referencias. (cont.)

Para mejorar el rendimiento del procesador, es decir, disminuir el tiempo de ejecución conviene, por ejemplo, traer varias instrucciones y/o datos cada vez que se accede a memoria y almacenarlos en una memoria cache que es más rápida.

El principio de cercanía de referencias puede aplicarse en cualquier nivel de la jerarquía:

- De memoria principal a cache.
- De cache a registros del procesador.
- De disco a memoria principal (se usa parte de la memoria como cache de disco).

24

Comunicación entre componentes de una PC.

Hemos visto que el CPU es el dispositivo que controla las operaciones que se realizan en la computadora y procesa los datos almacenados.

Para ello, el CPU establece comunicaciones con la memoria y con los dispositivos de E/S a través de los Buses.

Debido a que la memoria principal es básicamente una sola, la velocidad de comunicación entre CPU y memoria es constante; no así con los dispositivos de E/S, que tienen un amplio rango de velocidades.

Como éstos últimos tienen tiempos de acceso varios órdenes de magnitud más grandes que los de memoria, se utilizan técnicas diferentes para la comunicación con el CPU, a fin de optimizar la ejecución de los programas.

Estas técnicas se diferencian según el grado de intervención del procesador en la operación de E/S.

25

Comunicación entre componentes de una PC.

E/S Programada.

La técnica se denomina **programada** porque las tareas son realizadas, en su mayoría, por rutinas de software ejecutadas por el CPU. Éstas rutinas son:

- **Rutina de E/S (driver):** programa del SO cuya función es llevar a cabo la operación de E/S. Hay una rutina de E/S específica para cada dispositivo de E/S.
- **Código del módulo de E/S:** está almacenado en la ROM del módulo y realiza el transporte de datos entre buffer y el dispositivo, controlando los circuitos electrónicos y elementos mecánicos (si hubiere).

El procesador interviene en todas las tareas de la operación de E/S de los datos entre memoria y dispositivo.

26

Comunicación entre componentes de una PC.

E/S Programada. (cont.)

Los tipos de operaciones de E/S en esta técnica son:

1. Transferencia de datos de memoria a dispositivo de E/S.
2. Transferencia de datos de dispositivo de E/S a memoria.

Por ejemplo, en **1**, éstas son las tareas que se realizan:

- Un programa que ejecuta en el procesador genera datos y llena con ellos un bloque de memoria.
- Se transfieren esos datos al buffer de un módulo de E/S, mediante un conjunto específico de instrucciones del procesador.
- El procesador envía los comandos específicos al módulo para realizar alguna tarea de procesamiento de los datos enviados.
- Mientras los datos son procesados por el módulo de E/S, **el procesador se encuentra desocupado**, esperando que finalice el procesamiento.
- Al finalizar el procesamiento, el módulo de E/S notifica al procesador.
- El procesador retoma la ejecución del programa para generar nuevos datos.

27

Comunicación entre componentes de una PC.

E/S Programada. (cont.)

En las tareas descritas anteriormente, el **procesador se mantiene desocupado** mientras de el dispositivo de E/S realiza el procesamiento de los datos enviados desde memoria.

Esto implica que se detiene la ejecución del programa, por lo que **se estaría perdiendo la oportunidad de**, o bien **ejecutar otro programa** mientras el primero espera al dispositivo de E/S, o **seguir con la ejecución del mismo programa** realizando operaciones que no dependan de la E/S.

Para **aprovechar el tiempo de procesador desocupado** se plantea la técnica de **polling**.

28

Comunicación entre componentes de una PC.

Polling.

Consiste volver a utilizar el procesador para ejecutar un programa, ya sea el que estaba en ejecución u otro distinto, una vez que éste ha enviado la orden al dispositivo de E/S para que procese los datos enviados.

Luego, periódicamente, el procesador chequea el estado del dispositivo de E/S, a fin de verificar si han concluido las tareas de procesamiento que éste último debe realizar:

- Si aún no terminaron, retoma la ejecución del programa en curso.
- Si terminaron, invoca las siguientes rutinas de SO para concluir con la operación de E/S, y luego retoma la ejecución del programa en curso, o bien vuelve a ejecutar el programa que esperaba el resultado de la operación de E/S.

29

Comunicación entre componentes de una PC.

Polling. (cont.)

De esta manera, se aprovecha mejor el tiempo de uso de procesador, a pesar de que la rutina de polling para verificar el estado del dispositivo de E/S consuma parte de ese tiempo.

Observaciones:

Cuando se usa **polling** es **muy importante** determinar **con qué frecuencia se debe realizar la consulta**:

- Si la frecuencia es **alta**, hay **ineficiencia del procesador**.
- Si la frecuencia **es muy baja**, el dispositivo puede estar mucho **tiempo desocupado** hasta que el procesador detecte su estado.

Conclusión.

Este método es **eficiente en el caso de dispositivos lentos (teclado)**, pero **ineficiente en los dispositivos rápidos (discos)**.

30

Comunicación entre componentes de una PC.

Interrupciones.

El objetivo de esta técnica es permitir al procesador ejecutar un programa mientras se lleva a la cabo la operación de E/S.

Delega al controlador del dispositivo la responsabilidad de notificar la finalización de la operación de E/S. De esta manera, el procesador no realiza el polling y por lo tanto su participación en la operación es mucho menor.

Por esta razón, todas las plataformas de hardware de las computadoras actuales utilizan esta técnica.

Existen distintos tipos de interrupciones:

- **De programa:** se genera por un resultado específico obtenido luego de la ejecución de una instrucción.
- **De reloj:** un temporizador interno del CPU interrumpe la ejecución en intervalos regulares de tiempo.
- **De operación de E/S:** el módulo envía una interrupción (en adelante IRQ) al procesador para indicar algún evento específico.

31

Comunicación entre componentes de una PC.

Interrupciones. (cont.)

El funcionamiento de esta técnica es el siguiente:

- Cuando el programa en ejecución requiere de una operación de E/S, el procesador envía los comandos específicos al módulo controlador.
- Una vez enviados estos comandos, el procesador retoma la ejecución del programa o bien ejecuta otro, si es que la ejecución del primero depende de la operación de E/S.
- El módulo controlador del dispositivo realiza la operación que solicita el programa.
- Al finalizar la operación, envía al procesador una señal de interrupción (IRQ), para notificar esta situación.
- El procesador interrumpe la ejecución del programa en curso y ejecuta una rutina especial, llamada "**Interrupt Handler**" (IH), que se encarga de gestionar el pedido del dispositivo.
- Una vez finalizada la rutina de interrupción, se continúa con la ejecución del programa que corresponda.

32

Comunicación entre componentes de una PC.

Interrupciones. (cont.)

Las tareas que realiza el **Interrupt Handler** son las siguientes:

1. Guarda el estado de ejecución del programa que estaba en curso. Esto consiste en guardar los valores de los registros de control y de estado del procesador.
2. Chequea la **prioridad**, el **origen** y **tipo de tarea** requerida por la interrupción. En base a esto, **evalúa** si corresponde atender o no la interrupción:
 - Si IH decide **no atender** la interrupción, entonces IH restituye el uso del procesador al programa que venía ejecutando antes de producirse la interrupción (restituye en los registros del procesador la información que existía en el momento en que dejó de ejecutar el programa anterior para que pueda continuar su ejecución).
 - Si IH decide **atender** la interrupción, y la misma está relacionada con una operación de E/S, el IH llama a un **módulo específico** - Rutina de E/S - que realizará la tarea correspondiente.

33

Comunicación entre componentes de una PC.

Interrupciones. (cont.)

Observaciones:

- La ejecución del **Interrupt Handler** produce una **sobrecarga a la tarea de procesamiento**.
- No obstante, **esta sobrecarga, en tiempo, es mucho menor que la operación de lectura/escritura de un dispositivo**.
- De esta forma, **si bien hay una sobrecarga en la tarea del procesador (overhead), el uso de las interrupciones mejora la eficiencia global del procesamiento** y, por este motivo, **las IRQ son usadas por todos los procesadores actuales**.

34

Comunicación entre componentes de una PC.

Interrupciones. (cont.)

Interrupciones múltiples.

Cabe la posibilidad de que cuando un programa en ejecución se interrumpe, mientras se atiende la interrupción, llegue otra solicitud de interrupción. Y mientras ésta última se atiende, otra más puede llegar.

En ese caso, se deberá guardar el estado de ejecución de todos los programas que fueron interrumpidos. Esta situación se denomina **interrupciones múltiples**.

Se hace necesario, entonces, algún mecanismo que permita gestionar eficientemente estas sucesivas interrupciones, a fin de garantizar la correcta continuidad en la ejecución de los programas interrumpidos.

35

Comunicación entre componentes de una PC.

Interrupciones. (cont.)

Interrupciones múltiples. (cont.)

- Existen dos maneras básicas de gestionar las interrupciones múltiples:
 - **Inhabilitando las interrupciones.**
 - **Asignando prioridades.**
- **Inhabilitación de interrupciones:** cuando llega una interrupción, y el procesador la atiende, inmediatamente se inhabilitan las interrupciones posteriores. Esto implica que las que lleguen mientras se atiende una, quedan pendientes de atención, por lo que la atención de las mismas se realiza en forma secuencial, en el orden de llegada.
 - **Ventaja:** es una manera ordenada y simple de manejar las IRQ.
 - **Desventaja:** no tiene en cuenta las prioridades relativas de los programas, lo que puede llevar a situaciones críticas.

36

Comunicación entre componentes de una PC.

Interrupciones. (cont.)

Interrupciones múltiples. (cont.)

- **Asignación de prioridades:** de esta manera, cuando llega una interrupción de prioridad más alta que la que se está atendiendo, ésta interrumpe la atención de la de menos prioridad y pasa a ser atendida.
 - **Ventaja:** al tener en cuenta las prioridades relativas de los programas, es posible evitar situaciones críticas en la ejecución de programas de mayor prioridad.
 - **Desventaja:** la gestión de prioridades es más compleja que la anterior; sin embargo es la más eficiente.

37

Comunicación entre componentes de una PC.

DMA (Direct Memory Access).

Si bien en la técnica de interrupciones la participación del CPU en las operaciones de E/S es baja, sigue encargándose de la transferencia de datos.

Esto implica que la velocidad de transferencia de E/S está limitada por la velocidad con la que el procesador puede dar servicio a un dispositivo, lo cual representa una desventaja.

Por otro lado, resulta ineficiente:

- **En dispositivos rápidos.** Es el caso de una línea de comunicación de alta velocidad (Ej: LAN), la sobrecarga del procesador en el traspaso de estos datos puede significar una gran cantidad de ciclos del procesador, puesto que los servicios de éste son requeridos con alta frecuencia.
- **En alto volumen de datos a transferir.** En dispositivos medianamente rápidos y orientados a bloques de datos, como es el caso de un disco, los servicios del procesador serán requeridos varias o muchas veces en cada llenado del buffer del módulo de E/S, lo que también consume muchos ciclos del procesador.

38

Comunicación entre componentes de una PC.

DMA (Direct Memory Access). (cont.)

La alternativa para estos casos, entonces, es contar con un módulo de E/S con hardware específico para la transferencia de datos: el **módulo DMA**.

Con éste módulo, cuando se necesita transferir datos, el procesador emite una orden al módulo DMA enviándole información sobre:

- Tipo de operación a realizar.
- Dirección del dispositivo de E/S.
- Dirección inicial en memoria del bloque de datos a transferir.
- Cantidad de palabras a transferir.

Luego, el procesador se dedica a hacer otra tarea, y el módulo de DMA se encarga de toda la operación de E/S.

Cuando la transferencia de datos se completa, envía una interrupción al procesador.

39

Comunicación entre componentes de una PC.

DMA (Direct Memory Access). (cont.)

El módulo de DMA debe tomar el control del bus para transferir los datos a la memoria, por lo que puede existir **competencia con el procesador** por el uso del bus. El procesador debe esperar a que el bus se desocupe.

El **efecto global** que produce una **transferencia de DMA** es una ejecución del procesador **más lenta que la normal** durante la misma; puesto que, si bien, **el procesador está más libre para procesar otros programas**, debe disputar el bus con el módulo **DMA**, por ejemplo, cuando necesita traer un dato de memoria.

No obstante, en una **transferencia de E/S con alto volumen de datos**, **DMA es bastante más eficiente** que la E/S con interrupciones.

40

Fin del Módulo Introductorio