

Sistemas Operativos

Estructura de los sistemas de computación

Curso 2024

Facultad de Ingeniería, UDELAR

Agenda

1. Componentes de un sistema
2. Protección de hardware
3. Red

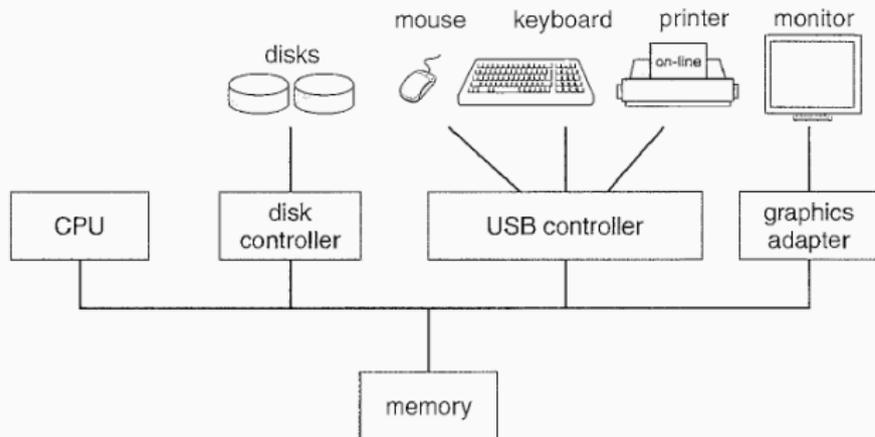
Componentes de un sistema

Componentes de un sistema

- CPU (procesador)
 - Unidad central de procesamiento. Permite ejecutar un conjunto de instrucciones. Su velocidad es varios órdenes mayor con respecto al acceso a la memoria.
- Memoria
 - Permite mantener la información disponible. Existen una jerarquía de memoria: registros, caches, memoria física de tipo RAM (**Random Access Memory**), dispositivos magnéticos, ópticos.
- Dispositivos de Entrada/Salida (I/O)
 - Permiten interactuar con el sistema. Algunos dispositivos más comunes: impresoras, teclados, ratón, video, disco, red, etc.

Componentes de un sistema

- Esquema gráfico:



CPU (procesador)

- La unidad central de procesamiento es la que ejecuta los programas. En un sistema puede haber más de una.
- El ciclo básico consiste en tomar la instrucción apuntada por el PC (**program counter**) (**fetching**), decodificarla para determinar su tipo y operandos (**decoding**), ejecutarla (**executing**), y luego continuar con la siguiente instrucción.
- Arquitecturas modernas aumentan la performance ejecutando las operaciones en paralelo (fetching, decoding, executing). Esta técnica es conocida como **pipelining**.
- Existen varias arquitecturas de procesador que se clasifican en **RISC** (Reduced Instruction Set Computer) o **CISC** (Complex Instruction Set Computer). Algunas arquitecturas: SPARC, POWER, x86, Itanium.

CPU (procesador)

- La velocidad del procesador es varios órdenes de magnitud mayor que la velocidad de acceso a información que está en la memoria volátil (RAM).
- Esto implicó la creación de registros a nivel del procesador y finalmente una cache de memoria (caches de 1er. Nivel, 2do. Nivel y hasta 3er. Nivel).
- Los registros son la memoria más rápida que accede un procesador y están integrados al chip.
- En los últimos años han surgido procesadores que en un mismo chip contienen varios núcleos de ejecución. Esto ha llevado a una nueva terminología: single-core, dual-core, quad-core, etc.

- Dentro del mismo chip del procesador se incluyen registros de rápido acceso:
 - Registros punto fijo y punto flotante.
 - Registros de direccionamiento ES, SS, DS, CS, etc..
 - Registro de Estado. Incluye PC y banderas con zero, carry.
 - Caches:
 - 1^{er} Nivel (del orden de 20 Kb).
 - 2^{do} Nivel (del orden de 512Kb a 2Mb).
 - 3^{er} Nivel (del orden de 8Mb).

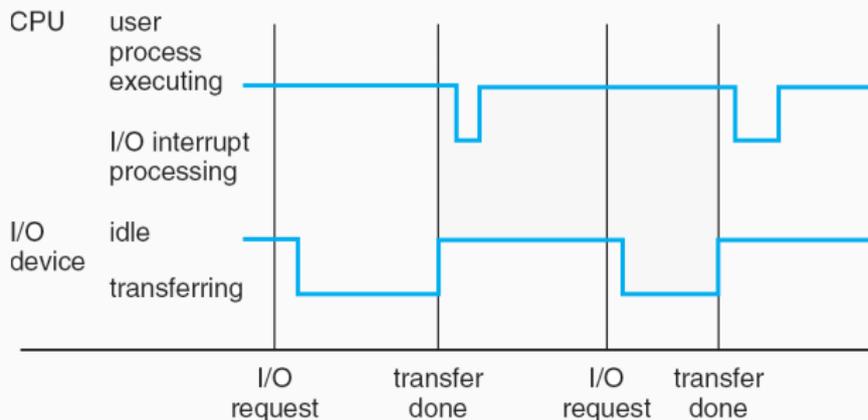
- Instrucciones
 - Operador Operandos...
- Los operandos pueden ser inmediatos, registros, relativos, de memoria DS:[SI] según diferentes técnicas. (vistos en Arquitectura de computadores).
- Las familias de instrucciones incluyen aritméticas, lógicas, transferencia control (Jump, Call, Loop, etc), de memoria, de stack, de sincronización (**Lock:XChg** ax, bx) y de entrada salida.
- Las instrucciones de sincronización sirven para resolver problemas de concurrencia

CPU: Instrucciones privilegiadas

- Se establecen niveles de ejecución y un conjunto de instrucciones para cada nivel.
- Un protocolo seguro para aumentar el nivel de ejecución que se basa en siempre transferir el control a código autenticado (**trusted**) para aumentar el nivel de ejecución.
- Por ejemplo:
 - Detener el procesador
 - Cambiar el vector de interrupciones
 - Cambiar las tablas de páginas

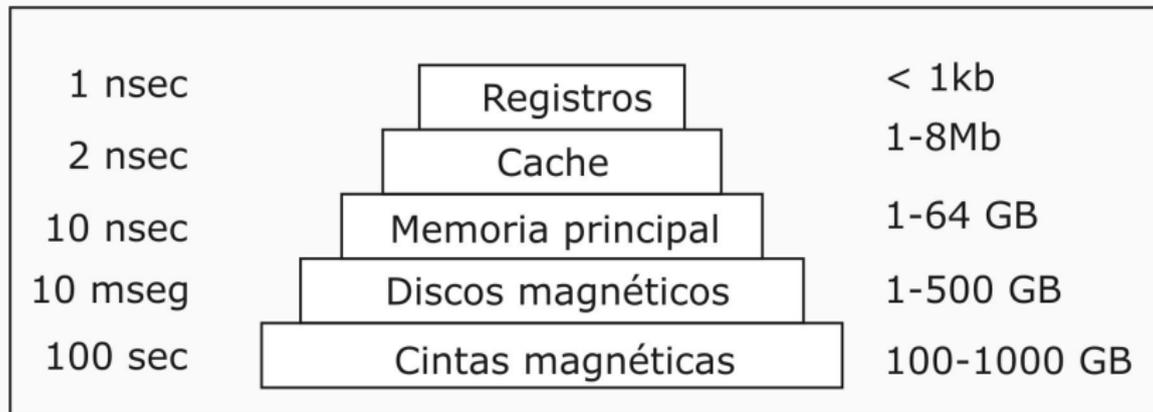
CPU: Interrupciones

- Interrumpen el flujo normal de un programa.
- Es la forma principal de comunicarse con el sistema operativo
- El sistema operativo preserva el estado actual (previo a la interrupción) del procesador (registros, etc.)
- Se determina que tipo de interrupción ocurrió.
- Se ejecuta la rutina de atención correspondiente.



Memoria

- El sistema de memoria es construido en base a una jerarquía, que permite mejorar la utilización del procesador:



Memoria: Memoria principal (RAM)

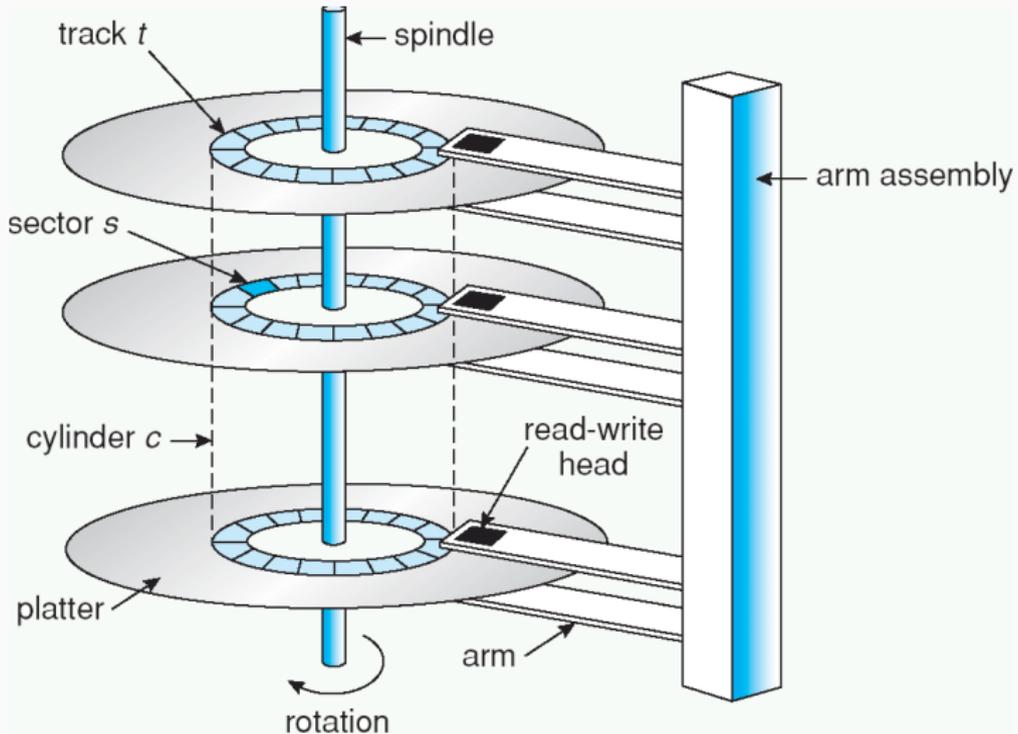
- Memoria de tipo volátil, con direcciones de palabra o byte.
- Palabra de 32, 48, 64 bits
- Transferencia en un ciclo del bus y acceso en paralelo (**interleaving**) a más de un módulo de memoria.
- Existen instrucciones que toman como argumentos direcciones de memoria.
- Es útil también para hacer transferencias con controladoras de dispositivos. Las controladoras tienen su propio **buffer** de memoria, y existen instrucciones de E/S que permiten la transferencia directa desde el buffer a memoria principal.

Memoria: Discos magnéticos (hard disk)

- Dispositivos de velocidad de acceso mucho menor que la memoria principal, pero de mayor capacidad.
- Tiene componentes mecánicas a diferencia de la memoria principal, cache y registros. Consta de:
 - Platos de metal que giran a alta velocidad (entre 6.000 y 10.000 rpm)
 - Un brazo mecánico que contiene las cabezas de lectura/escritura para cada plato
- La superficie de los platos se divide en secciones:
 - **Pistas** (tracks): La superficie de los platos es dividida lógicamente en pistas circulares.
 - **Sectores** (sectors): Cada pista es dividida en un conjunto de sectores.
 - **Cilindros** (cylinders): El conjunto de pistas (de todos los platos) que están en una posición del brazo mecánico forman un cilindro.

Memoria: Discos magnéticos (hard disk)

- Esquema de discos magnéticos



Memoria: Discos magnéticos (hard disk)

- La velocidad del disco tiene dos componentes:
 - Tasa de transferencia (**transfer rate**): Es la tasa con la cual los datos van entre el disco y la computadora.
 - Tiempo de posicionamiento (**positioning time**): Es el tiempo que se tarda en ubicar el brazo en el cilindro adecuado (**seek time**), mas el tiempo de rotar el plato al sector adecuado (**rotational latency**).
- La unidad de transferencia es el bloque. Ocasionalmente los bloques pueden estar con **interleaving**.
- Existen distintos tipos de buses de conexión:
 - **IDE** (Integrated drive electronics)
 - **ATA** (Advanced Technology Attachment)
 - **SATA** (Serial Advanced Technology Attachment)
 - **SCSI** (Small Computer-Systems Interface)
 - **SAS** (Serial Attached SCSI)

Discos de estado sólido (SSD)

- Están contruidos con memoria flash no volátil.
- Son más confiables que los discos magnéticos porque no tienen partes móviles.
- Son más rápidos que los discos magnéticos porque no tienen **seek time** o **rotational latency**.
- Son más caros y generalmente más chicos que los discos magnéticos.
- Los datos no se pueden sobrescribir, hay que borrarlos si se quieren cambiar. La memoria se deteriora con cada ciclo de borrado.
- Su vida útil no se mide en años sino en escrituras por día (**Drive Writes Per Day**).

Memoria: Cache

- El cache es un principio muy importante, es utilizado a varios niveles en el sistema de computación (hardware, sistema operativo, software).
- El concepto es mantener una copia de la memoria que está siendo utilizada en un medio temporal de mayor velocidad de acceso.
- El medio de memoria cache es mucho menor en capacidad, pero más veloz que el dispositivo principal. Esto genera que el manejo de cache es un problema de diseño importante.
- El tamaño del cache y sus políticas de reemplazo tienen un alto impacto en la mejora real de la performance.

Memoria: Coherencia de la cache

- Un problema que introduce la memoria cache en ambientes de multiprocesadores, es la coherencia y consistencia de los datos que están replicados.
- Caches en multiprocesadores:
 - Mayor rendimiento, no se satura el bus del sistema (cuello de botella).
 - Aún en un monoprocesador, hay que contemplar a los controladores de dispositivos.
 - Problemas de coherencia entre caches, ya que una palabra puede estar replicada en diferentes caches de los procesadores. El problema de coherencia se torna mucho más complicado.
 - Surgen técnicas como **write-through** y **write-back**.

Dispositivos de entrada/salida (I/O)

- Los dispositivos, por lo general, se componen de una controladora y el dispositivo en sí.
- La controladora es un chip que controla físicamente al dispositivo. Acepta comandos del sistema operativo y los ejecuta (genera las correspondientes señales sobre el dispositivo para realizar la tarea).
- La interfaz que le presenta la controladora al sistema operativo es bastante más simple que la provista por el dispositivo.
- En un sistema existen distintas controladoras (de discos, red, etc.), por eso es necesario distintos componentes de software para manejar cada uno.

Dispositivos de entrada/salida: Device drivers

- Al software que se comunica con la controladora se le denomina **device driver**.
- Para cada controladora se debe proveer el device driver adecuado. Estos son incorporados al sistema operativo dado que son la vía de comunicación con los dispositivos.
- Los device drivers son cargados de diferentes formas:
 - Ensamblados estáticamente al núcleo del sistema.
 - Cuando se carga el sistema se lee un archivo de configuración que menciona cuales device drivers cargar.
 - Cargados dinámicamente a demanda.

- Las controladoras contienen un conjunto de registros que sirven para comunicarse con ella y ejecutar comandos.
 - Ej.: la controladora de un disco podría tener registros para especificar la dirección en disco, la dirección en memoria principal, el número de sectores y el sentido (lectura y escritura).
- Acceso a los registros de la controladora:
 - **Memory mapped I/O**: Los registros son “mapeados” a direcciones de memoria principal.
 - **Direct I/O instructions**: A los registros se le asigna una dirección de puerto (I/O port address)

Dispositivos de entrada/salida: Memory mapped I/O

- Para facilitar el acceso a registros de los dispositivos, se reserva un espacio de la memoria principal que mapea a los registros del dispositivo.
- Leer o escribir en los registros de los dispositivos se traduce en leer o escribir sobre las direcciones de memoria. Al operar sobre estas direcciones de memoria se genera la transferencia a los registros del dispositivos en forma transparente.
- Las direcciones de memoria deben ser puesta fuera del alcance de los procesos del usuario.
- Ej.: La pantalla es mapeada a un lugar de memoria. Para desplegar un carácter en pantalla solo basta con escribir sobre el lugar correcto de la memoria principal.

Dispositivos de entrada/salida: I/O port address

- A cada registro se le asigna una dirección de puerto.
- El sistema cuenta con instrucciones privilegiadas **IN** y **OUT** que permiten a los device drivers leer o escribir en los registros de la controladora.
- La instrucción genera señales en el bus del sistema para seleccionar el dispositivo adecuado.

I/O address range (hexadecimal)	device
000-00F	DMA controller
020-021	interrupt controller
040-043	timer
200-20F	game controller
2F8-2FF	serial port (secondary)
320-32F	hard-disk controller
378-37F	parallel port
3D0-3DF	graphics controller
3F0-3F7	diskette-drive controller
3F8-3FF	serial port (primary)

- Memory-mapped I/O:
 - No necesita de instrucciones especiales: simplifica la CPU
 - Tiene el problema de que hay que excluir esas direcciones de los procesos de usuarios
- Direct I/O instructions:
 - No consume memoria principal.
 - Las instrucciones de I/O deben ser privilegiadas

Dispositivos de entrada/salida: Interacción con la controladora

Métodos para efectuar una operación de entrada-salida:

- **Espera activa** (Polling): El procesador le comunica un pedido a la controladora del dispositivo y queda en un busy waiting consultando a la controladora si está listo el pedido.
- **Interupciones** (Interrupts): El procesador le comunica el pedido a la controladora y se libera para realizar otras tareas. Al culminar el pedido el dispositivo, la controladora genera una interrupción al procesador.
- **Acceso directo a memoria** (DMA – Direct Memory Access): Se utiliza un chip especial que permite transferir datos desde alguna controladora a memoria sin que el procesador tenga que intervenir en forma continua.

Dispositivos de entrada/salida: Espera activa

- El sistema queda en **busy waiting** consultando un registro del controlador para saber si está listo.
- Ej.: Imprimir un buffer en una impresora.

```
p = copy_from_user(buffer, k_buffer, count);
for (i = 0; i < count; i++) {
    while (*printer_status_reg != READY);
    *printer_data.register = p[i];
}
return_to_user();
```

Dispositivos de entrada/salida: Interrupciones

- El sistema se independiza del controlador, que genera una interrupción cuando finaliza el pedido.
- Es necesario tener un vector de rutinas de atención de interrupciones (**interrupt vector**), que es cargado cuando se inicia el sistema operativo.
- Ej.: Imprimir un buffer en una impresora.

```
p = copy_from_user(buffer, k_buffer, count);  
while (*printer_status_reg != READY);  
i = 0;  
*printer_data.register = p[i];  
scheduler();
```

- Ej.: Rutina de atención de la interrupción.

```
if (i == count)
    unblock_user();
else {
    i++;
    *printer_data.register = p[i];
}

return_from_interrupt();
```

Dispositivos de entrada/salida: DMA

- Se dispone de un dispositivo especializado que permite realizar transferencias desde ciertos dispositivos a memoria. La transferencia se hace en paralelo mientras el procesador realiza otras tareas.
- El procesador carga ciertos registros en el controlador DMA para realizar el pedido. El controlador DMA se encarga de la tarea de transferencia, interrumpiendo al procesador cuando finalizó.

Ej.: Imprimir un buffer en una impresora.

```
p = copy_from_user(buffer, k_buffer, count);  
set_up_DMA_controller();  
scheduler();
```

Ej.: Rutina de atención de la interrupción de DMA.

```
unblock_user();  
return_from_interrupt();
```

Protección de hardware

Protección de hardware

- Con la introducción de sistemas multiprogramados y multiusuarios se empezaron a generar problemas en el uso de los recursos debido a procesos “mal programados” o “mal intencionados”.
- Fue necesario la introducción de protección entre los distintos procesos que ejecutaban en un sistema.
- El hardware fue suministrando a los sistemas operativos de mecanismos para la protección:
 - **Modo Dual**: Se provee de al menos dos modos de operación.
 - **Protección de E/S**: Todas las instrucciones de Entrada/Salida son privilegiadas.
 - **Protección de Memoria**: Evaluación de las direcciones de memoria a través de la MMU.
 - **Protección de CPU**: Introducción de un timer que permite limitar el uso de CPU.

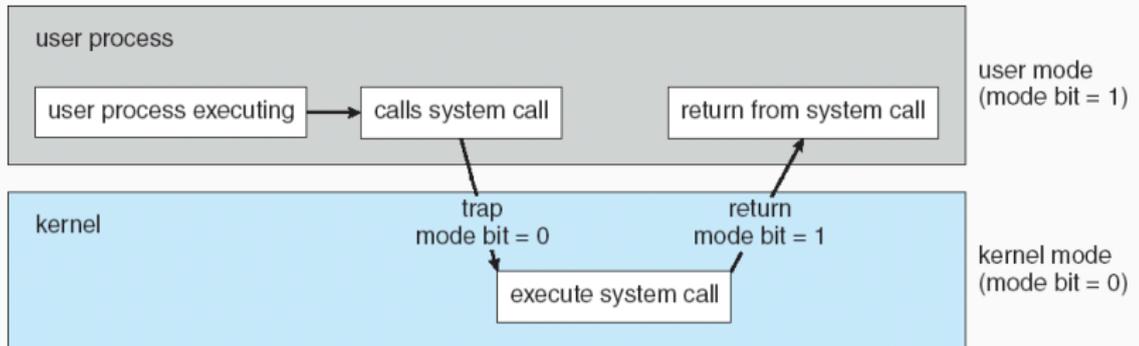
- El hardware provee al menos dos modos de ejecución:
 - **Modo usuario** (user mode): en este modo de ejecución se puede ejecutar un conjunto reducido de instrucciones de hardware. Los procesos a nivel de usuarios ejecutan en este modo.
 - **Modo monitor** (monitor mode): en este modo todas las instrucciones de hardware están disponibles. El sistema operativo es el único que debe ejecutar en este modo.
- Un bit, llamado **mode bit**, es agregado al hardware para indicar el modo actual.

Modo dual

- La ejecución de instrucciones privilegiadas en el modo monitor garantiza que los procesos, a nivel de usuario, no accedan directamente a los dispositivos de E/S.
- El acceso a un dispositivo se realiza a través de los servicios que brinda el sistema operativo (**syscall**).
- La solicitud de un servicio al sistema operativo es tratado como una interrupción a nivel de software (**trap**), y en ese momento el sistema pasa de modo usuario a modo monitor.
- En Intel la instrucción `int 0x80` genera el cambio de modo.
- Posteriormente, se ejecuta el **handler** de la excepción `0x80` (128 decimal).

Modo dual

- Esquema gráfico del cambio de modo:

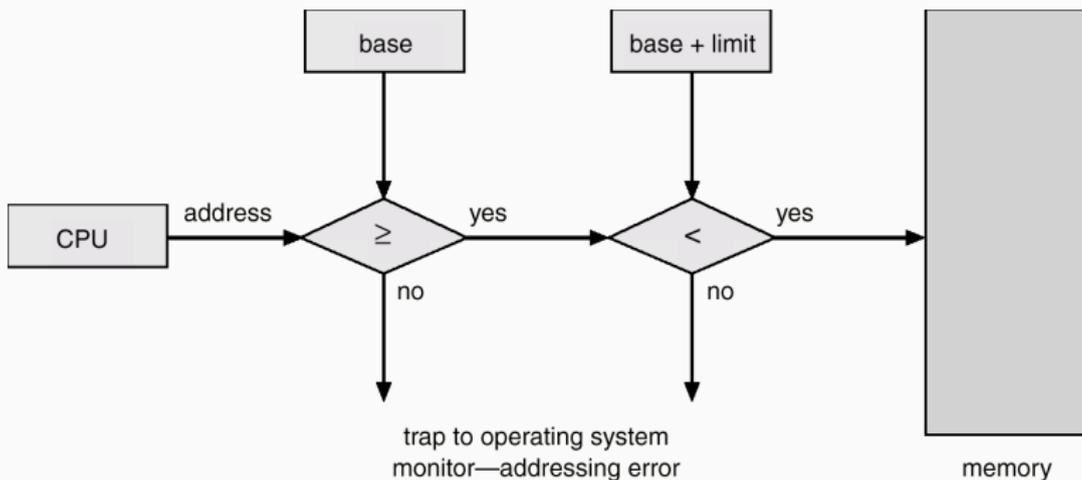


- Es necesario restringir que los procesos a nivel de usuario no accedan directamente a los dispositivos, sino que deban hacerlo a través del sistema operativo.
- Por eso, se define que **todas** las instrucciones de E/S son privilegiadas.

- Es necesario proteger la memoria del núcleo (p.ej.: el vector de interrupciones) y, a su vez, proteger el acceso de memoria entre los distintos procesos (un proceso no debería acceder a la memoria de otro).
- El sistema debe lograr saber si cada dirección generada por un proceso es válida.
- Una forma es utilizar dos registros:
 - **Base**: Contiene la dirección de memoria física más baja que puede acceder.
 - **Límite**: Contiene el tamaño del bloque de memoria a partir del registro base.

Protección de memoria

- Esquema gráfico de la protección a través de registro base y límite:



Protección de memoria

- Cada dirección física generada por la CPU es controlada para comprobar si es una dirección válida.
- En caso de una acceso inválido se genera un trap al sistema operativo.
- La unidad que convierte direcciones lógicas a físicas es la MMU (**Memory Management Unit**), y es la que controla el acceso a memoria. Esta es un dispositivo de hardware.
- La unidad MMU únicamente debe ser administrada en modo monitor. Por ejemplo cargar los registros base y límite.

- Una vez que a un proceso se le asigna un recurso procesador, puede entrar en una iteración infinita (infinite loop) y no retornar nunca más el control al sistema.
- Deben existir mecanismos de protección de uso del procesador.
- Una alternativa es la utilización de un **timer** que interrumpa el procesador cada cierto tiempo.

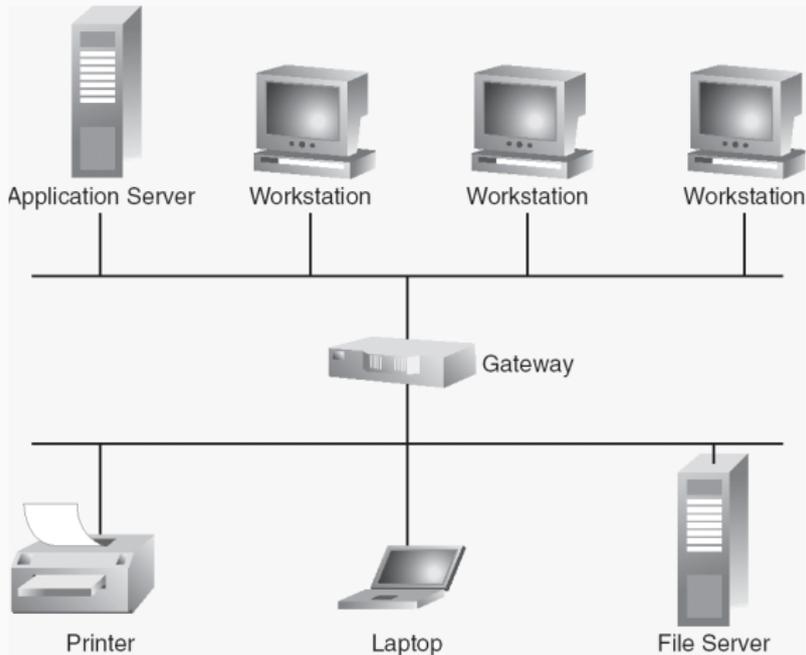
- El sistema operativo al asignar la CPU carga un contador. Cada vez que la interrupción de timer se genera se ejecuta la rutina de atención correspondiente.
- En la rutina de atención de la interrupción el contador es disminuido. Si alcanza al valor 0, se le quita el recurso procesador al proceso y se invoca al planificador para que seleccione otro.
- La instrucción que permite cargar el contador debe ser privilegiada.

Red

- Las redes se pueden clasificar, básicamente, en dos tipos:
 - **Red LAN** (Local Area Network):
 - Las redes LAN son pequeñas y su alcance está limitado por lo general a no más de un edificio.
 - Velocidades de 10, 100, 1000 Mbits/s, o más.
 - **Red WAN** (Wide Area Network):
 - Las redes WAN son redes distribuidas sobre una región grande.
 - 1,5 a 100 Mbits/s.
- La diferencia principal es como están geográficamente distribuidas.

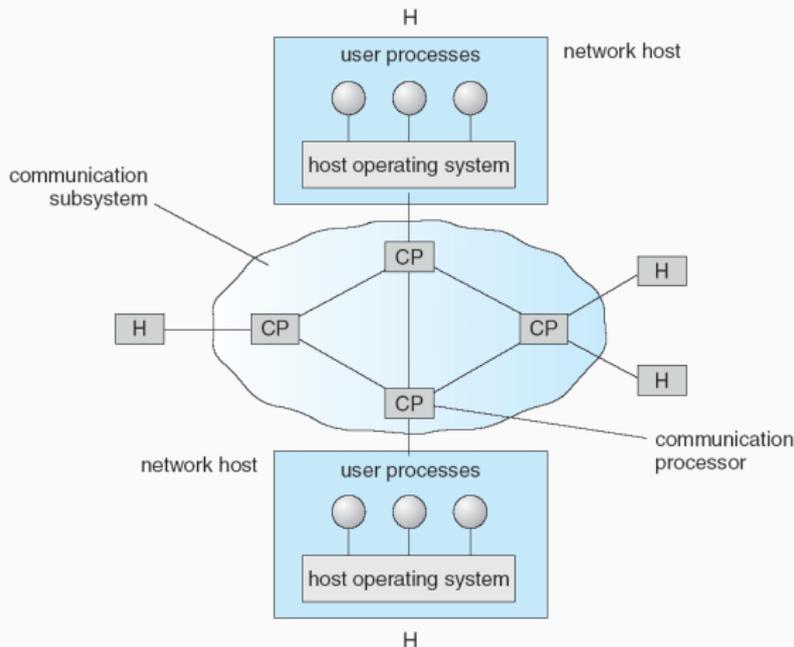
LAN: Local Area Networks

- Son redes que interconectan sistemas a corta distancia y se tiende a tener interconexiones de alta velocidad con baja tasa de error.



WAN: Wide Area Networks

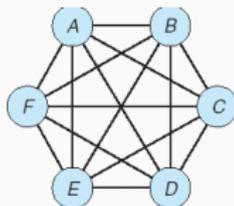
- Son redes que interconectan sistemas remotos.
- Los enlaces, por lo general, son provistos por empresas de telecomunicaciones.



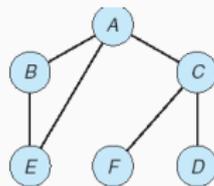
- Las redes pueden estar interconectadas de diferentes formas.
- Esto dependerá de:
 - Costos básicos: Qué costo a nivel monetario implica interconectar la red.
 - Costo a nivel de comunicación: Qué tiempo lleva enviar un mensaje desde un nodo a otro de la red.
 - Nivel de confianza: Qué tan resistente es la red ante eventuales fallos de componentes.
- Las topologías que se implementen dependerán de estas tres variables.

Topologías de red

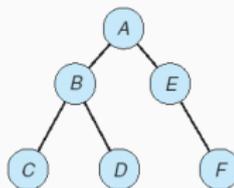
- Esquema gráfico de algunas topologías más comunes:



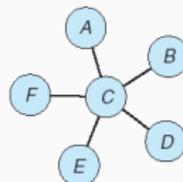
fully connected network



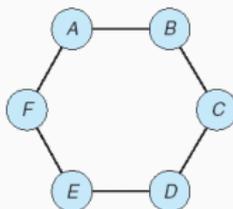
partially connected network



tree-structured network



star network



ring network