

12 SISTEMAS DE E/S

- 1 GENERALIDADES
- 2 HARDWARE DE E/S
- 3 INTERFAZ DE APLICACIÓN DE E/S
- 4 SUBSISTEMA DE KERNEL DE E/S
- 5 TRANSFORMACIÓN DE REQUERIMIENTOS DE E/S EN OPERACIONES DE HARDWARE
- 6 RENDIMIENTO

12.1 GENERALIDADES

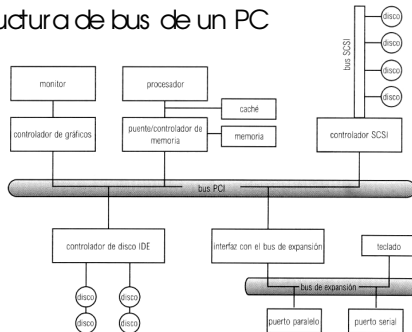
- Control de Dispositivos: preocupación importante en el diseño de un S.O.
- Alta heterogeneidad en funcionalidad y velocidad $\Rightarrow$ diversidad de métodos de control
- Conjunto de métodos constituye el Subsistema de E/S del núcleo, el cual aísla al resto del núcleo de la complejidad del manejo de E/S.
- Conflicto estandarización $\leftrightarrow$ variedad impulsa el desarrollo conjunto de técnicas de hardware - software.

12.2 HARDWARE DE E/S

- Gran variedad de dispositivos de E/S
- Conceptos comunes
 - Puerto
 - Bus (cadena o acceso directo compartido)
 - Controlador (adaptador de host o anfitrión)
- Dispositivos de control de instrucciones de E/S
- Los dispositivos tienen direcciones, usadas por
 - Instrucciones de E/S directa por ej: transferencia de un byte a una dirección de puerto.
 - E/S mapeada en memoria: correspondencia entre registros de control del disp. y el espacio de direcciones del procesador.

12.2 HARDWARE DE E/S

Estructura de bus de un PC



12.2 HARDWARE DE E/S

Puertas de E/S en un PC

Intervalo de Direcciones	Dispositivo
000 - 00F H	Controlador DMA
020 - 021 H	Controlador de Interrupciones
040 - 043 H	Timer
200 - 20F H	Controlador de Juegos
2F8 - 2FF H	Puerta Serial (2ª)
320 - 32F H	Controladora de Disco Duro
378 - 37F H	Puerta Paralela
3D0 - 3DF H	Controladora Gráfica
3F0 - 3F7 H	Controladora de Disquetera
3F8 - 3FF H	Puerta Serial (1ª)

12.2.1 Escrutinio (Polling)

- Determina estado del dispositivo
 - Lista para un comando
 - Ocupada
 - Error
- Cdo ocupada - en espera activa por la E/S de un dispositivo
- La CPU se encarga de escuchar repetidamente para detectar una finalización de E/S.
- ¿qué pasa si encuentra pocas veces un dispositivo listo para ser atendido y tiene tareas de procesamiento pendientes?
- Controlador notifica a la CPU ¿cómo?

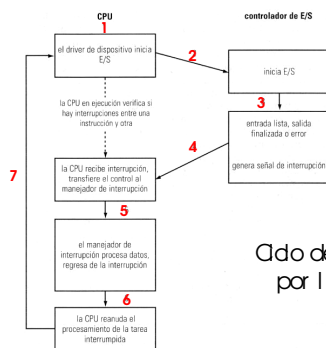
12.2.2 Interrupciones

- La interrupción a la CPU hace un requerimiento a la línea de solicitud de interrupción (IRQ), gatillada por el dispositivo de E/S
- El manejador de interrupciones recibe las interrupciones
- Se enmascara para ignorar o retrasar alguna interrupción
- Se vectorizan las interrupciones para despacharlas al manejador correcto
 - Basado en prioridad
 - algunas son no enmascarables
- El mecanismo de interrupciones también es usado para excepciones

12.2.2 Interrupciones

- Se utiliza el mecanismo de interrupciones para la paginación de memoria virtual. Un fallo es una excepción.
- Llamados al sistema: verifica argumentos dados por la aplicación, construye una estructura de datos para comunicar los argumentos al núcleo y luego ejecuta una instrucción llamada interrupción, se commuta a modo supervisor y se ejecuta lo solicitado.
- En resumen: Actualmente se usan interrupciones para manejar sucesos asincrónicos y transferir el control al modo supervisor. Tareas urgentes: sistema de prioridades de interrupciones

12.2.2 Interrupciones



Ciclo de E/S controlado por Interrupciones

12.2.2 Interrupciones

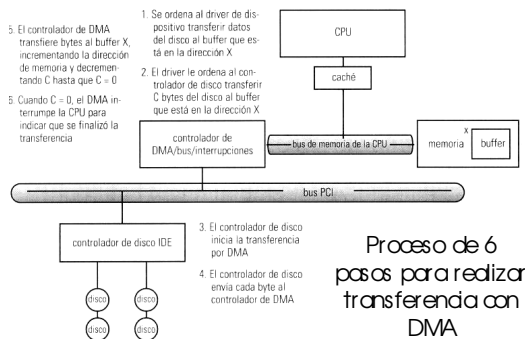
Códigos de Interrupción del Pentium

número de vector	descripción
0	error de división
1	excepción de depuración
2	interrupción nula
3	punto de ruptura
4	desbordamiento detectado por INTO
5	excepción de intervalo acortado
6	código de operación no válido
7	dispositivo no disponible
8	doble fallo
9	rebalse del segmento de coprocesador (reservado)
10	segmento de estado de tarea no válido
11	segmento ausente
12	fallo de pila
13	protección general
14	fallo de página
15	(reservado por Intel, no usar)
16	error de punto flotante
17	verificación de alineación
18	verificación de máquina
19-31	(reservado por Intel, no usar)
32-255	interrupciones enmascarables

12.2.3 Acceso Directo a Memoria

- Usado para evitar que se emplee la E/S programada (ver bits de estado, alimentar datos a un registro de controlador byte por byte) en el movimiento de gran cantidad de datos
- Requiere controlador DMA
- Se solicita a la CPU para transferir datos directamente entre el dispositivo de E/S y la memoria
- Se escribe un bloque de órdenes de DMA en la memoria que contiene:
 - Un puntero al origen de la transferencia
 - Un puntero al destino de la transferencia
 - Un contador del número de bytes a transferir

12.2.3 Acceso Directo a Memoria



12.2.3 Acceso Directo a Memoria

- Conceptos principales:
 - Bus
 - Controlador
 - Puerto de E/S y sus registros
 - La relación entre el host y un controlador de dispositivo
 - La ejecución de la comunicación anterior en un ciclo de escrutinio o mediante interrupciones
 - Delegar trabajo al DMA cuando las transferencias son muy grandes
- ¿cómo se pueden conectar nuevos dispositivos sin tener que reescribir el S.O.?
- ¿se puede presentar una interfaz de E/S uniforme?

12.3 INTERFAZ DE APLICACIÓN DE E/S

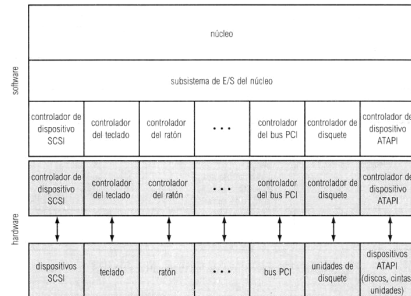
- Se soluciona con abstracción, encapsulamiento y estructuración del sw en capas.
- Por abstracción se pueden eliminar las diferencias específicas identificando unas cuantas clases generales. Se accede a través de una interfaz.
- Las diferencias reales se encapsulan en módulos del S.O. denominados controladores de dispositivos
- Se estructuran entonces en capas

12.3 INTERFAZ DE APLICACIÓN DE E/S

- Por lo tanto, las llamadas al sistema para E/S encapsulan el comportamiento del dispositivo en clases genéricas
- La capa del manejador del dispositivo (driver) esconde al kernel las diferencias entre controladores de E/S
- Dispositivos varían en base a muchas características, por ende un dispositivo dado existe para MS-DOS, Windows 95, NT y Solaris.

12.3 INTERFAZ DE APLICACIÓN DE E/S

Estructura de E/S del núcleo



12.3 INTERFAZ DE APLICACIÓN DE E/S

Características de Dispositivos de E/S

ASPECTO	VARIACIÓN	EJEMPLO
Modo de Transferencia de Datos	Carácter Bloque	Terminal Disco
Método de Acceso	Secuencial Directo	Modem CD-ROM
Itinerario de Transferencia	Sincrónico Asincrónico	Anta Teclado
Compartición	Dedicado Compartido	Anta Teclado
Velocidad de Dispositivo	Latencia Búsqueda (seek) Tasa de Transferencia Retardo entre Operaciones	
Dirección de la E/S	Sólo lectura Sólo escritura Lectura/Escritura	CD-ROM Controlador Gráficos Disco

12.3.1 Dispositivos de Bloque y Carácter

- Los dispositivos de bloque comprenden a los discos
 - Los comandos incluyen read, write, seek
 - Hacen E/S "bruta" (acceder a un dispositivo por bloques como un arreglo lineal) o en base al sistema de archivos
 - Es posible el acceso a archivos mapeados en memoria: una interfaz con mapa en memoria, proporciona acceso al disco a través de un arreglo de bytes en la mem. Ppal., se devuelve la dirección de mem. Virtul que contiene una copia del archivo.

12.3.1 Dispositivos de Bloque y Caracter

- Los dispositivos de carácter incluyen teclados, ratones y puertos seriales
 - Comandos incluyen get (obtener), put (colocar) un carácter
 - Bibliotecas en capas superiores permiten edición de una línea a la vez, por ej; se tecldea un retroceso, se elimina el carácter anterior.

12.3.2 Dispositivos en Red

- Unix y Windows/NT incluyen interfaz de tipo socket
 - Separa el protocolo de red de la operación de la red
 - Incluye funcionalidad de tipo select: devuelve información sobre cuáles socket tienen un paquete que espera ser recibido, cuáles sockets tienen espacio para aceptar un paquete que se enviará
- Los tipos varían ampliamente (pipes, FIFOs, streams, queues, mailboxes)

12.3.3 Relojes y Timers

- Proveen el tiempo actual, tiempo transcurrido y el valor del timer para iniciar una operación Y en un tiempo T.
- Si se utilizan intervalos de tiempo programables para timing, se programa para que espere cierto tiempo y luego ocurra una interrupción o interrupciones periódicas.
- IOCTL (Temporizador Programable de Intervalos, en UNIX) comprende aspectos de la E/S tales como relojes y timers

12.3.4 E/S Bloqueada y No bloqueada

- Bloqueado: el proceso se suspende hasta que la E/S se completa, de la cola de ejecución pasa a la cola de espera y luego se coloca otra vez en la cola de ejecución
 - Es fácil de usar y entender
 - Es insuficiente para algunas necesidades
- No bloqueado: la llamada retorna todo lo disponible, sólo algunos o ninguno
 - Interfaz de usuario, copia de datos (E/S con buffer)
 - Implementada via multi-hebra
 - Retorna rápidamente con el contador de bytes leídos o escritos

12.3.4 E/S Bloqueada y No bloqueada

- Asíncrono: el proceso se ejecuta mientras la E/S procede
 - Difícil de usar
 - El subsistema de E/S envía señales al proceso cuando se completa la E/S

12.4 SUBSISTEMA DE E/S DEL KERNEL

- 1 Iteración
- 2 Uso de Buffers
- 3 Uso de Cache
- 4 Uso de SPOOL
- 5 Manejo de Errores
- 6 Estructura de Datos del Kernel

12.4.1 Itineración de E/S

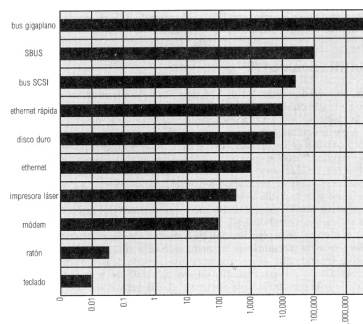
- Algunos requerimientos de E/S se ordenan vía colas por dispositivos
- Algunos Sistemas Operativos usan "imparcialidad", es decir trata de ser eficiente con todas las solicitudes.
- Por ej: brazo de disco cerca del principio del disco
 - Aplicación 1 solicita un bloque cerca del final
 - Aplicación 2 solicita un bloque cerca del principio
 - Aplicación 3 solicita un bloque cerca de la mitad
 El orden sería 231 y no en el orden 123. El planificador lo reordena para mejorar la eficiencia global del S.O.

12.4.2 Uso de Buffers

- Mediante buffers, se almacenan datos en memoria mientras se lleva a cabo la transferencia entre los dispositivos:
 - Para tratar con la disparidad de velocidad entre los dispositivos (el que envía y el que recibe)
 - Para tratar con la disparidad de tamaño de la transferencia entre los dispositivos
 - Para mantener la "semántica de copiado": Se garantiza que la versión de los datos que se escribió en el disco es la versión que existía en el momento en que la aplicación hizo la llamada al sistema

12.4.2 Uso de Buffers

Tasa de Transferencia de Dispositivos
Sun Enterprise 6000
(logarítmicas)



12.4.3 Uso de Caché

- Se usa una memoria escondida rápida que mantiene copia de los datos (caché)
 - Siempre es sólo una copia
 - El acceso a la copia en caché es más eficiente que el acceso original
 - Es clave para el rendimiento
- La diferencia entre un buffer y un caché
 - Buffer: contiene la única copia existente de un elemento de información
 - Caché: contiene una copia en almacenamiento más rápido de un elemento que existe en otro almacenamiento

12.4.4 Uso de SPOOL y Reserva

- Mediante SPOOLING, se mantiene la salida para un dispositivo
 - El dispositivo puede servir sólo un requerimiento a la vez (puede ser compartido)
 - Ej., impresora (se provee al usuario ver la cola de trabajos, eliminar, pausar, continuar etc)
- Reserva de dispositivo (POOL); provee acceso exclusivo a un dispositivo
 - Llamadas al sistema para asignación y desasignación
 - Potencialmente podría ocurrir deadlock

12.4.5 Manejo de Errores

- El Sistema Operativo se puede recuperar de lectura de disco, dispositivo no disponible y fallas transitorias de escritura
- La mayoría retorna un número o código de error cuando falla un requerimiento de E/S
- El sistema registra los errores y los reporta

12.4.6 Estructuras de datos del Núcleo

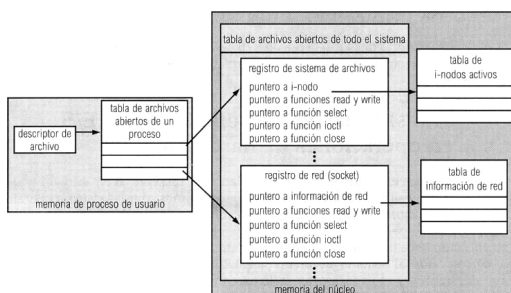
- El kernel mantiene información de estado para los componentes de E/S, incluyendo tablas de archivos abiertos, conexiones de red, estado de dispositivos de caracteres
- Hay muchas estructuras de datos complejas para registrar la actividad de los buffers, asignación de memoria, bloques "sucios"
- Algunas usan métodos orientados a objeto (unix) y traspaso de mensajes (windows NT) para implementar la E/S

12.4.6 Estructuras de datos del Núcleo

- El subsistema de E/S coordina una amplia colección de servicios. Supervisa:
 - Administración del espacio de nombres para archivos y dispositivos
 - Control de acceso a archivos y dispositivos
 - Control de operaciones
 - Asignación de espacio para el sistema de archivos
 - Asignación de dispositivos
 - Uso de buffers, cachés y spool
 - Planificación de E/S
 - Vigilancia de los estados de dispositivos, manejo de errores y recuperación después de fallos
 - Configuración e iniciación de drivers de dispositivos

12.4.6 Estructuras de datos del Núcleo

Estructura del Núcleo de E/S de Unix

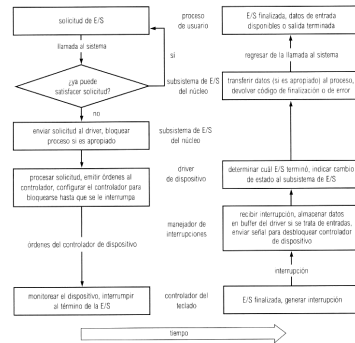


12.5 TRANSFORMACIÓN DE REQUERIM. DE E/S A OPERACIONES DE HW

- Considerar que un proceso está leyendo un archivo en disco
 - Determinar en qué dispositivo está el archivo
 - Traducir el nombre a la representación del dispositivo
 - Físicamente, leer los datos del disco al buffer
 - Hacer que los datos queden disponibles para el proceso que los requiere
 - Retornar el control al proceso

12.5 TRANSFORMACIÓN DE REQUERIM. DE E/S A OPERACIONES DE HW

Ciclo de Vida de un Requerimiento de E/S

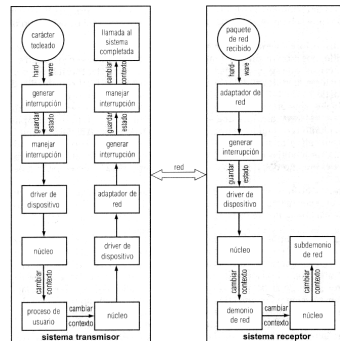


12.6 RENDIMIENTO

- E/S un factor principal en el rendimiento del sistema
 - Demanda por usar la CPU para ejecutar driver del dispositivo, código de E/S del kernel
 - Hace cambio de contexto debido a interrupciones: ejecuta el manejador de interrupciones y luego restaura el estado
 - Copia de datos: se somete a una carga al bus de memoria entre los controladores y la memoria física
 - El tráfico en la red también presenta una frecuencia de conmutaciones de contexto elevadas

12.6 RENDIMIENTO

Comunicación
entre
Computadores



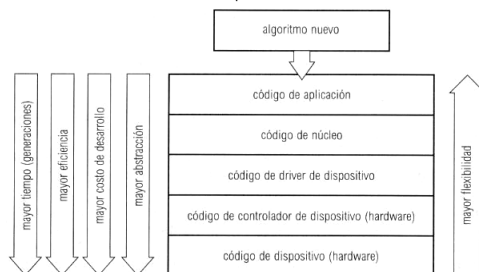
12.6 RENDIMIENTO

- Mejora del Rendimiento
 - Reducir el número de cambios de contexto
 - Reducir la copia de datos en la memoria durante la transferencia entre el dispositivo y la aplicación
 - Reducir la frecuencia de interrupciones usando grandes transferencias, controladores inteligentes, polling
 - Uso de DMA
 - Se balancean la CPU, memoria, bus y rendimiento de la E/S para throughput mayor

12.6 RENDIMIENTO

¿Dónde debe implementarse la funcionalidad de E/S, en el HW del dispositivo, en el driver del dispositivo o en el SW de aplicación?

Secuencia de Funcionalidad de Dispositivos:



12.6 RENDIMIENTO

- Nivel de aplicación:
 - Flexible y poco probable que induzca a caídas del sistema
 - No se requiere volver a cargar los dispositivos o reiniciar después de cada cambio
 - Existe gasto extra en las conmutaciones de contexto
 - No puede utilizar las ventajas de las estructuras de datos internas y funcionalidad del S.O.
- Nivel de núcleo:
 - Mejora el desempeño con respecto al anterior
 - Desarrollo es más complejo
 - Depuración de código es exhaustiva
- Nivel de HW:
 - Desempeño más alto logrado
 - Dificultad y costo para realizar mejoras adicionales o corregir errores
 - Tiempo de desarrollo es mucho mayor
 - Menor flexibilidad
