

TEMA 5: Gestión de Entrada/Salida

- Introducción
- Software de E/S
- Discos
- Relojes
- Terminales

1. Introducción

- Función del sistema operativo

Controlar la Entrada/Salida

- Enviar comandos
- Recibir interrupciones
- Tratar los errores
- Presentar una interfaz

Categorías de dispositivos

- Dispositivos de bloques

- » Bloques de tamaño fijo
- » L/E de cada bloque de forma independiente
- » Se puede direccionar cada bloque

DISCOS

- Dispositivos de caracteres

- » Sin estructura de bloques
- » L/E de flujos de caracteres

TERMINALES

IMPRESORAS

Manejadores y Controladores

- Manejador de dispositivos

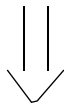
Programa de más bajo nivel que se encarga de los detalles que son dependientes del dispositivo

- Controlador de dispositivo o adaptador

Elemento electrónico de la unidad de E/S
(tarjeta de circuitos impresos)

Registros

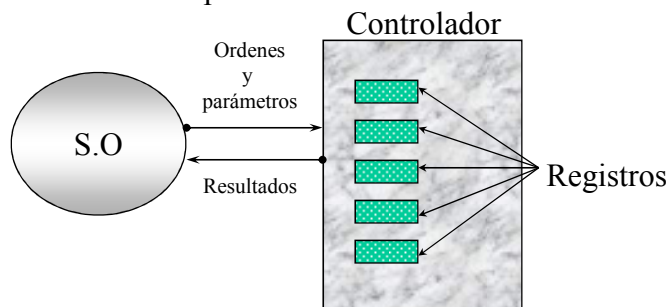
- Forman parte del espacio normal de direcciones de la memoria
- Espacio de direcciones especial para E/S



Puertos de E/S

Operaciones de E/S

- El S.O. escribe órdenes y parámetros en los registros.
- El controlador acepta la orden y comienza a trabajar.
- El controlador genera una interrupción.
- El S.O. mira si la operación se ha realizado correctamente.



2. Software de E/S

- Objetivos de la programación de la E/S
 - » Independencia del dispositivo
 - » Uniformidad de los nombres
 - » Manejo de errores
 - » Presentar al programador transferencias síncronas
 - Síncronas: bloquean al programa.
 - Asíncronas: no bloquean al programa.
 - » Gestión de los dispositivos
 - compartidos
 - dedicados

Niveles del software de E/S

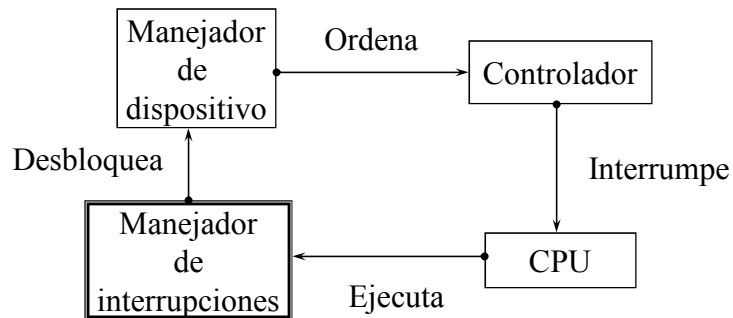
Programas de usuario

Programas del S.O
independientes del
dispositivo

Manejadores de
dispositivos

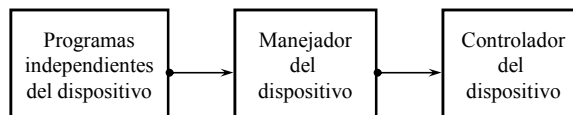
Manejadores de
interrupciones

Manejadores de interrupciones



Manejadores de dispositivo

- Código dependiente del dispositivo
- Función:



- » Enviar órdenes a los dispositivos
- » Comprobar que se ha realizado correctamente
- La petición de los programas independientes del dispositivo se llevan a una cola
- Única parte del S.O que conoce los registros del controlador

Programas del S.O independientes del dispositivo

10

- Funciones (I)
 - » Presentar una interfaz uniforme a los programas de usuario
 - » Asignar nombres y establecer la correspondencia entre nombre y manejadores
 - En UNIX los dispositivos se tratan como ficheros
 - N° de dispositivo principal: Para localizar el manejador
 - N° de dispositivo secundario: Para especificar la unidad referenciada.
 - » Proteger a los dispositivos de accesos no permitidos

Programas del S.O independientes del dispositivo

11

- Funciones (II)
 - » Ocultar las diferencias en las unidades de información tanto en los dispositivos de bloques como de caracteres
 - » Almacenar temporalmente los datos de los dispositivos (buffering)
 - completar bloques, sincronizar, entrada de teclado...
 - » Asignar espacio libre en los dispositivos de bloques
 - » Gestionar el uso de los dispositivos dedicados
 - Aceptar o rechazar las peticiones (ej: OPEN)
 - » Tratar los errores

Programas de usuario

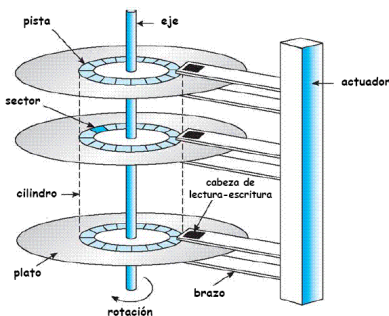
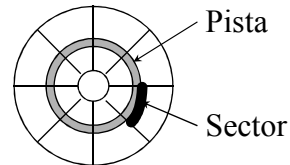
- Funciones de biblioteca
 - » Realizar las llamadas al sistema:
`count=read(file,buf,nbytes);`
 - » Realizar además otras funciones:
`printf(“ %d ”, i);`
- Programas completos fuera del núcleo
 - » Sistema de spooling
 - Ej: Impresora
 - Demonio de impresión, directorio de spooling

3. Discos

- Ventaja del disco sobre la memoria para el almacenamiento
 - » Mayor capacidad de almacenamiento
 - » Precio por bit más bajo
 - » La información no se pierde

Hardware del disco

- Pistas: círculos concéntricos
- Se dividen en sectores
 - » Mismo n° de bytes
- Organizados en cilindros
 - N° de pistas del cilindro =
cabezas apiladas verticalmente
- Unidad con n platos
 - » 2n cabezas
 - » 2n pistas por cilindro



Característica del dispositivo

- Búsquedas solapadas
 - » Mejoran el tiempo de acceso
 - » Búsquedas simultáneas en 2 o más unidades
 - » El controlador puede iniciar una búsqueda mientras que espera que se complete una búsqueda en otra unidad
 - » No se pueden realizar 2 lecturas o escrituras a la vez

Programación del disco

- Tiempo de lectura o escritura:

tiempo de búsqueda + latencia de rotación + tiempo de transferencia

- El tiempo de búsqueda es mayor que los demás tiempos
- Se intenta mejorar =>

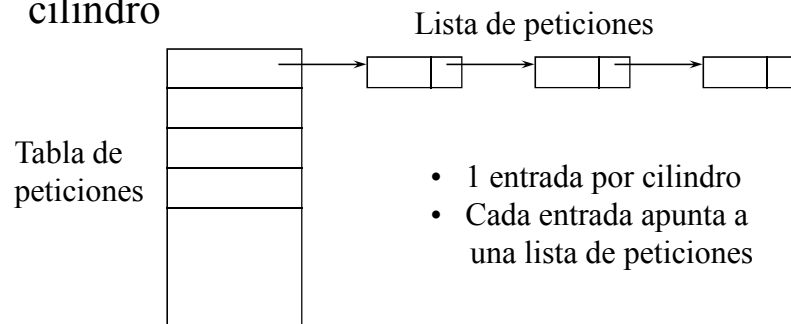
Algoritmos de planificación del brazo del disco

FCFS

- Primero en llegar, primero en servirse
- El manejador acepta las peticiones de 1 en 1
- Sirve las peticiones en el orden de llegada
- No se puede mejorar el tiempo de búsqueda

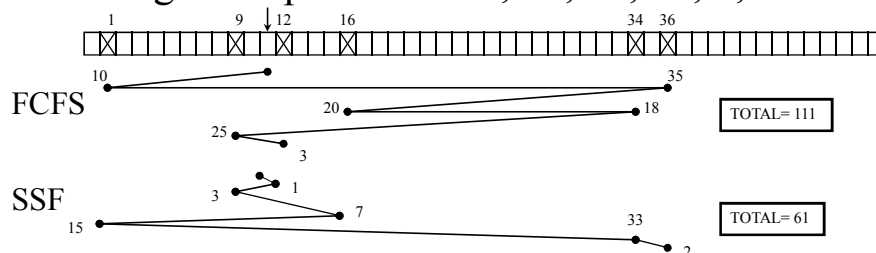
SSF (I)

- Prioridad a la búsqueda más corta
- tabla de peticiones indexada por nº de cilindro



SSF (II)

- Se mejora el tiempo de búsqueda tomando la solicitud al cilindro más cercano.
- Cilindro inicial => 11
- Llegan las peticiones 1, 36, 16, 34, 9, 12

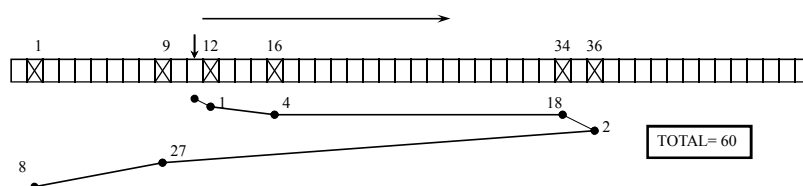


Problemas de SSF

- El brazo tiende a quedarse en el centro para discos sobrecargados
- Los extremos tienen que esperar a que no haya peticiones en el centro
- No se hace un servicio equitativo de las peticiones

Algoritmo del ascensor

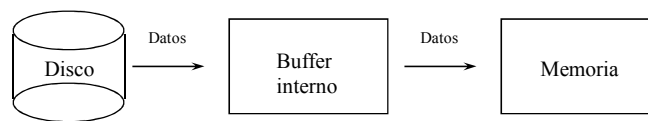
- Bit SUBIENDO/BAJANDO



- Límite superior: doble del nº de cilindros
- Variación: siempre sentido ascendente

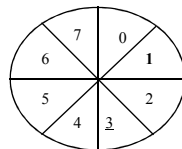
Entrelazado (I)

- Técnica que se utiliza para dar tiempo a transferir los datos del buffer interno a memoria

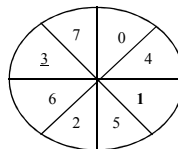


entrelazado (II)

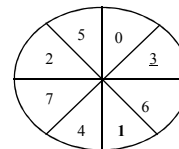
Factor de entrelazado:



Entrelazado simple



Entrelazado doble



Entrelazado triple

Otras optimizaciones

- Optimización en la latencia de rotación
 - » Enviar la petición del sector que pase antes bajo la cabeza de L/E
- Copia de pistas completas
 - » Se almacena la pista completa en un buffer intermedio del manejador
 - » Complica el software
 - » No permite DMA para la información del buffer
 - » Algunos controladores tienen una memoria interna
- Búsquedas solapadas

Tratamiento de errores

- Errores de programación
- Errores transitorios en el código de control
- Errores permanentes en el código de control
- Errores de búsqueda
- Errores en el controlador

Errores de programación

- El controlador recibe órdenes del manejador
 - » mover el brazo a un cilindro
 - » leer un sector
 - » escribir los datos de una posición de memoria
- Los parámetros pueden no ser coherentes
- Se debe a que el programa falla
- El controlador debe devolver un código de error

Errores transitorios

- Los datos no son correctos
- Por ejemplo debido a la existencia de polvo en las cabezas
- Se puede eliminar el error repitiendo la operación

Errores permanentes

- Marcar bloques como inservibles
 - » Elaboración de un fichero con la lista de bloques inservibles
 - El asignador nunca utilizará estos bloques
 - Las copias de seguridad se deben hacer fichero a fichero, evitando leer el fichero de bloques defectuosos
 - » Reserva de pistas
 - Se localizan los bloques inservibles al formatear
 - Se sustituyen las pistas de reserva por las dañadas
 - Se utiliza una tabla para hacer la correspondencia

Errores de búsqueda

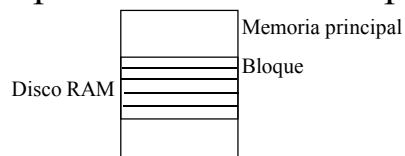
- Errores mecánicos
- El brazo se posiciona en un cilindro que no se pidió
- RECALIBRATE: Desplaza el brazo hasta el final y considera el cilindro 0
- Llamar al técnico

Errores en el controlador

- Variables, buffers, órdenes
- Secuencia de eventos que produzcan un error en el controlador
- Ej: el controlador no admite órdenes
- El manejador debe dar una orden para iniciar el controlador

Discos RAM

- Dispositivo de bloques
- Se utiliza una parte de la memoria principal



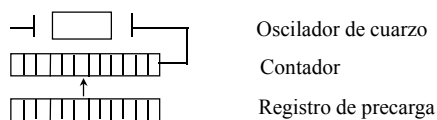
- La memoria se divide en bloques
- El manejador calcula la posición del bloque
- Acceso instantáneo

Relojes

- = Temporizadores
- Mantienen la fecha y la hora
- Sirve para controlar el tiempo que los procesos pueden estar ejecutandose

Hardware del reloj

- 2 tipos
 - Se conectan a la red de potencia y generan una interrupción con cada ciclo de red
 - Programables: se construyen con 3 elementos



- Para un reloj de 1MHz ($1/10^{-6}$ Hz, 1 pulso por microsegundo) y registro de precarga de 16 bits, se pueden generar interrupciones desde cada microsegundo hasta 65535 microsegundos

Programación del reloj

- El hardware genera interrupciones a intervalos regulares y conocidos
- El manejador debe realizar todas las funciones


Actualizar fecha y hora (I)

- En ordenadores antiguos no había reloj de tiempo real
- La fecha se da al arrancar. Se traduce a un nº de pulsos desde el 1 de Enero de 1970.
- Con cada pulso de reloj se incrementa un contador
- Problema: se puede desbordar

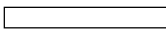
Actualizar fecha y hora (II)

- Solución:

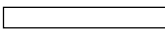
- » Utilizar un contador más grande (Ej: 64 bits)

 64 bits

- » Se guarda en segundos con un contador auxiliar para contar el nº de pulsos del segundo actual



Hora y fecha
en segundos



Nº de pulsos del
segundo actual

2^{32} = más de 136 años

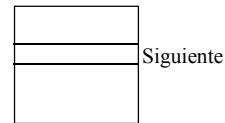
- » Se guarda en segundos el instante de arranque del sistema y un contador para los pulsos desde el momento de arranque.

Controlar el tiempo de ejecución

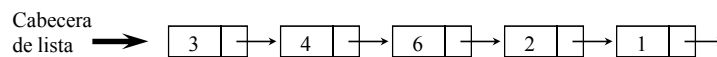
- Al cargar un proceso
 - » Se actualiza un contador con los pulsos del cuanto asignado a un proceso
- Con cada interrupción de reloj
 - » Se decrementa este contador
- Cuando llega a 0
 - » Se llama al planificador

Temporizadores

- Para avisar a los procesos (señal, mensaje...)
- Ej: Aplicación que retransmite paquetes transcurrido un cierto intervalo de tiempo si no se ha confirmado
- Tabla con tiempos de aviso
- Lista enlazada



3 Siguiendo señal



Temporizadores de guarda

- Temporizadores del propio sistema
- Se suelen utilizar en los manejadores de dispositivos
 - » Esperar 250 mls después de arrancar el motor de una unidad
 - » En vez de generar una señal, el manejador llama al procedimiento indicado

Otras funciones

- Medir tiempos de ejecución
- Realizar estadísticas

5. Terminales

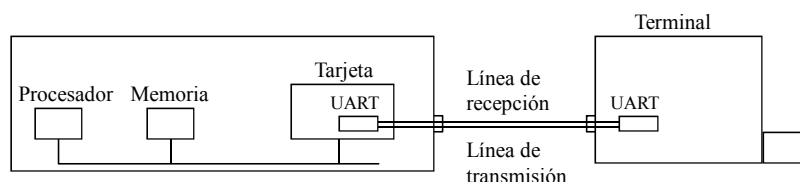
- Muchos tipos diferentes de terminales
- El manejador se encarga de ocultar las diferencias
- De esta forma, la parte del S.O. independiente del terminal sirve para cualquier terminal

Hardware del terminal

- Desde el punto de vista del S.O. hay dos tipos diferentes de terminales
 - » Terminales con interfaz RS-232
 - » Terminales con interfaz por memoria (mapped-memory terminals)

Terminales con interfaz RS-232

- Utilizan una línea serie para comunicarse con el ordenador
- UART : Conversor de paralelo a serie y de serie a paralelo



Terminales con interfaz RS-232

44

- El manejador escribe el carácter en la interfaz y se bloquea
- Es almacenado temporalmente por la UART para luego mandarlo bit a bit
- El manejador se desbloquea cuando llega la interrupción de la interfaz

Ventaja: cualquier ordenador trae una interfaz RS-232
Desventaja: son lentos (ej: 9600 bits/s)

Categorías de terminales RS-232

45

- Teletipos-impresoras
 - » Imprimen los caracteres enviados en papel
- Teletipos de cristal (tty)
 - » Mandan el carácter a un CRT
- Terminales inteligentes
 - » Tienen procesador y memoria
 - » Entienden ciertas teclas de control
 - » Se puede hacer que el cursor se mueva por la pantalla, escribir texto en mitad de la pantalla ...

Terminales con interfaz por memoria

46

- Son parte del mismo ordenador
- La comunicación se hace a través de la RAM de vídeo que es parte del espacio de direcciones
- El controlador lee bytes de la memoria RAM y envía las señales al monitor para que imprima los caracteres
- Tipos
- El teclado es independiente del monitor
- El hardware proporciona un n° de tecla, el manejador hace la correspondencia

Terminales de despliegue de caracteres
Terminales de mapas de bits

Terminales de despliegue de caracteres

47

- El controlador almacena en ROM los patrones de bits de cada carácter
- El carácter que el procesador escribe en la memoria RAM de vídeo aparece en la pantalla transcurrido un cierto tiempo
- Un byte asociado a cada carácter
- Son rápidos
- Imagen de 25x80 => 4000 bytes

Terminales de mapas de bits

- Considera la pantalla como una matriz de elementos de imagen llamados **pixels**
800x600 1204x768 1152x864
- Cada pixel controlado por un bit de la RAM
- Permite crear varios patrones para las letras
- Planos de bits para los colores
(n planos $\Rightarrow 2^n$ colores)
- Requieren mucha memoria

1024x1024 \Rightarrow 128 Kbytes

con 4 bits por pixel \Rightarrow 0,5 Mbytes

Programación del terminal

- Software de entrada
- Software de salida

Software de entrada (I)

50

- El manejador obtiene los caracteres de teclado
- Dos modos de funcionamiento
 - » Modo puro: secuencia ASCII
 - » Modo elaborado: edición dentro de la línea
- Tablas de correspondencia
- Buffer para guardar líneas
- ioctl : llamada al sistema para controlar los parámetros del terminal

Software de entrada (II)

51

- Funciones
 - » Eco en pantalla
 - » Truncar la línea
 - » Tabuladores
 - » Conversión de intro
 - » Borrado de caracteres
 - » Borrado de líneas
 - » Secuencias de escape
 - » CTRL-S CTRL-Q
 - » CTRL-C CTRL-D

Software de salida

52

- En terminales de interfaz RS-232
 - » El manejador copia un carácter del buffer de salida y se bloquea hasta que llega la interrupción
- En terminales de interfaz por memoria
 - » los caracteres se copian en la RAM de vídeo
 - » posición en la RAM de vídeo
 - » scroll de pantalla
 - » Posicionamiento del cursor
 - » Inserción de texto