

Administración de E/S

El sistema operativo debe:

- Emitir comandos para los dispositivos
- Captar interrupciones
- Manejar errores
- Proporcionar una interfaz "*simple y fácil de usar*" entre los dispositivos y el resto del sistema.
 - Si es posible, que sea igual para todos los dispositivos. *Independencia de dispositivos*

El código de E/S representa una fracción considerable del sistema operativo total.

Dispositivos de E/S

Se pueden dividir en dos categorías:

Dispositivos de bloque

Almacena información en bloques de tamaño fijo, cada uno con su propia dirección.

Los tamaños de bloque comunes varían desde 512 bytes hasta 32,768 bytes.

Todas las transferencias se realizan en unidades de uno o más bloques completos **consecutivos**.

Es posible leer o escribir cada bloque de manera independiente de los demás.

Es direccionable y permite búsquedas.

Discos duros, CD-ROMs y memorias USBs son dispositivos de bloque comunes.

Dispositivos de carácter

Envía o acepta un flujo de caracteres, sin importar la estructura del bloque.

No es direccionable y no tiene ninguna operación de búsqueda.

Impresoras, interfaces de red, ratones y cualquier dispositivo que no sea parecido a un disco se puede considerar como un dispositivo de carácter.

Important

Este sistema de clasificación **no es perfecto**. Existen excepciones como los relojes o las pantallas por asignación de memoria

Otra agrupación que se puede hacer, a grandes rasgos, en **3 categorías**:

Legibles para el usuario: Adecuados para la comunicación con el usuario del computador.

impresoras, terminales gráficas

Legibles para la máquina: Adecuados para la comunicación con equipamiento electrónico.

unidades de disco y cintas, sensores, controladores, activadores

Comunicación: Adecuados para la comunicación con dispositivos remotos. controladores de una línea digital, módems

Diferencias entre categorías

Velocidad de transferencia de datos

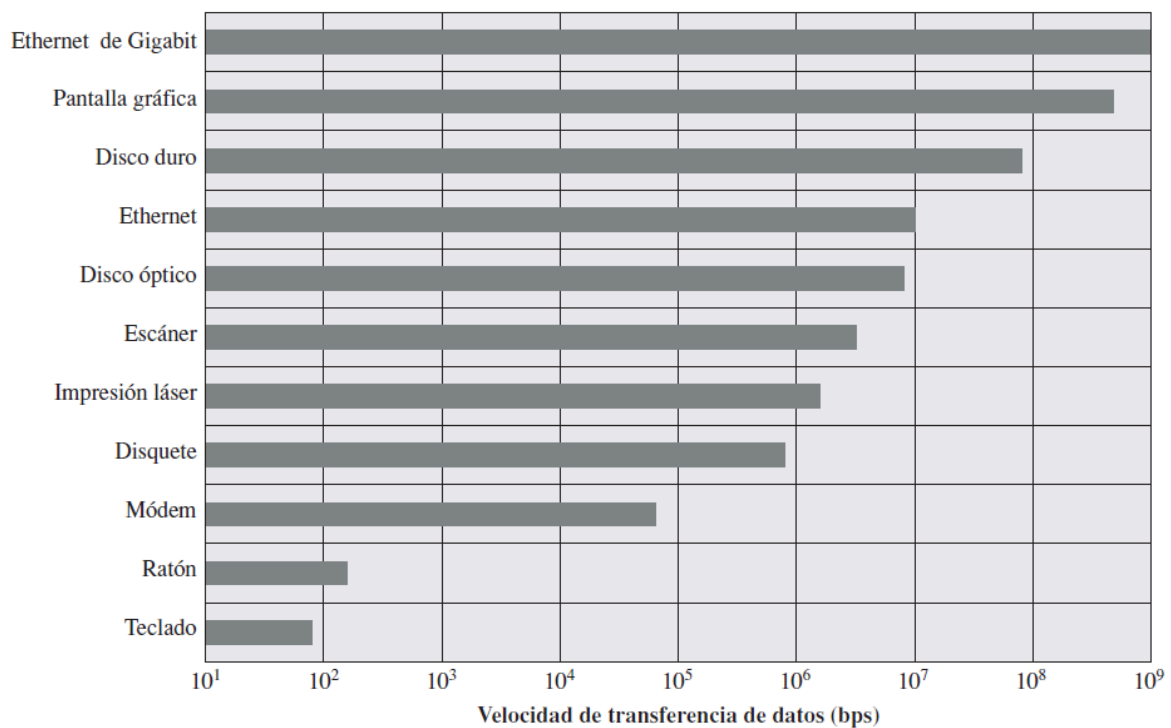


Figura 11.1. Velocidad típica de transferencia de datos de E/S.

Aplicación

El uso al que está destinado un dispositivo tiene influencia en el software y en las políticas del sistema operativo y de las herramientas que le dan soporte. `terminal` que puede utilizar un usuario normal o un administrador del sistema implica diferentes niveles de privilegio y diferentes prioridades

Complejidad de control

Una impresora requiere una interfaz de control sencilla. Un disco es mucho más complejo.

El efecto que tienen estas diferencias en el sistema operativo se filtra hasta cierto punto por el módulo de E/S que controla el dispositivo.

Unidad de transferencia

Los datos pueden transferirse como:

- Flujo de bytes o caracteres `E/S de un terminal`
- Bloques de mayor tamaño `E/S de un disco`

Representación de datos

Los dispositivos utilizan diferentes esquemas de codificación de datos, incluyendo diferencias en el código del carácter y en las convenciones sobre la paridad.

Condiciones de error

La naturaleza de los errores, el modo en que se notifican, sus consecuencias y el rango disponible de respuestas difieren considerablemente de un dispositivo a otro.

Controladores de dispositivos

Por lo general, las unidades de E/S consisten en un componente mecánico y un componente electrónico.

- **Componente electrónico**
 - Se le llama **controlador de dispositivo** o **adaptador**.
 - En computadoras personales, comúnmente tiene la forma de un chip en la tarjeta principal o una tarjeta de circuito integrado.
- **Componente mecánico**
 - Es el dispositivo en sí.

Si la interfaz entre el controlador y el dispositivo es estándar (*ya sea ANSI, IEEE, ISO o un estándar de facto*), entonces las empresas pueden fabricar controladores o dispositivos que se adapten a esa interfaz.

Esta interfaz es a menudo de muy bajo nivel.

El trabajo del controlador es convertir el flujo de bits serial en un bloque de bytes y realizar cualquier corrección de errores necesaria.

Muchas empresas fabrican unidades de discos con interfaz IDE, SATA, USB.

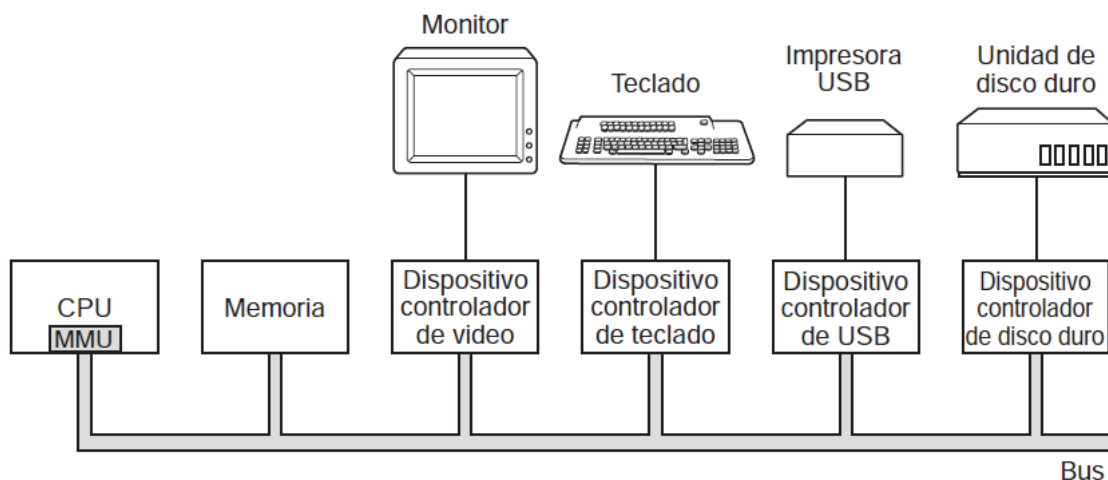


Figura 1-6. Algunos de los componentes de una computadora personal simple.

Objetivos del software E/S

Independencia de dispositivos

Debe ser posible escribir programas que puedan acceder a cualquier dispositivo de E/S sin tener que especificar el dispositivo por adelantado.

Depende del sistema operativo encargarse de los problemas producidos por el hecho de que son dispositivos diferentes.

```
sort <entrada> salida
cat /dev/input/mice
```

Denominación uniforme ~ Independencia de denominación

El nombre de un archivo o dispositivo simplemente debe ser una cadena o un entero sin depender del dispositivo de ninguna forma.

En **UNIX**, todos los archivos y dispositivos se direccionan mediante el nombre de una ruta.

```
mount /usr/ast/respaldo
```

Manejo de errores

En general, los errores se deben manejar lo más cerca del hardware que sea posible.

Si el controlador descubre un error de lectura, debe tratar de corregir el error por sí mismo. Si no puede, entonces el software controlador del dispositivo debe manejarlo.

Muchos errores son transitorios y comúnmente desaparecen si se repite la operación.

Note

Sólo si los niveles inferiores no pueden lidiar con el problema, los niveles superiores deben saber acerca de ello.

En muchos casos, la recuperación de errores se puede hacer de manera transparente a nivel bajo, sin que los niveles superiores se enteren.

Transferencias síncronas o asíncronas

Síncronas: De bloqueo.

Asíncronas: Controladas por interrupciones.

La mayoría de operaciones de E/S son asíncronas. La CPU inicia la transferencia y se va a hacer algo más hasta que llega la interrupción

Los programas de usuario son muchos más fáciles de escribir si las operaciones de E/S son de bloqueo (**síncronas**). Después de una llamada `read()`, el programa se suspende de manera automática hasta que haya datos disponibles en el búfer

Note

Depende del sistema operativo hacer que las operaciones que en realidad son controladas por interrupciones (**asíncronas**) parezcan de bloqueo (**síncronas**) para los programas de usuario.

Utilización de buffers

A menudo, los datos que provienen de un dispositivo no se pueden almacenar directamente en su destino final.

Algunos dispositivos tienen severas restricciones en tiempo real, por lo que los datos se deben colocar en un búfer de salida por adelantado para desacoplar la velocidad a la que se llena el búfer, de la velocidad que se vacía, evitando sub-desbordamientos de búfer. **dispositivos de audio digital**

El uso del búfer involucra una cantidad considerable de copiado y a menudo tiene un importante impacto en el rendimiento de la E/S.

E.x ~ Placa de red

Cuando un paquete llega de la red, el sistema operativo no sabe donde colocarlo hasta que se almacena en alguna parte y lo examina.

Dispositivos compartidos vs dedicados

Compartidos

Pueden ser utilizados por muchos usuarios a la vez.

No se producen problemas debido a que varios usuarios tengan archivos abiertos en el mismo disco al mismo tiempo.

Discos

Dedicados

Tienen que estar dedicados a un solo usuario hasta que éste termine.

Si dos o más usuarios escriben bloques entremezclados al azar en la misma cinta, no va a funcionar.

Introducen **interbloqueos**.

Unidades de cinta

Organización del sistema de E/S

Existen **3** técnicas para llevar a cabo la E/S:

En la mayoría de computadoras, el DMA es la forma de transferencia predominante, a la que el sistema operativo debe dar soporte.

E/S programada

El procesador envía un mandato de E/S, a petición de un proceso, a un módulo de E/S.

Ese proceso realiza una espera activa hasta que se complete la operación antes de continuar.

E/S dirigida por interrupciones

El procesador emite un mandato de E/S a petición de un proceso y continúa ejecutando las instrucciones siguientes. Es interrumpido por el módulo de E/S cuando éste ha completado su trabajo.

Las **instrucciones siguientes** pueden ser o no del mismo proceso.

Acceso directo de memoria (DMA)

Un módulo de DMA controla el intercambio de datos entre la memoria principal y un módulo de E/S.

El procesador manda una petición de transferencia de un bloque de datos al módulo de DMA y resulta interrumpido sólo cuando se haya transferido el bloque completo.

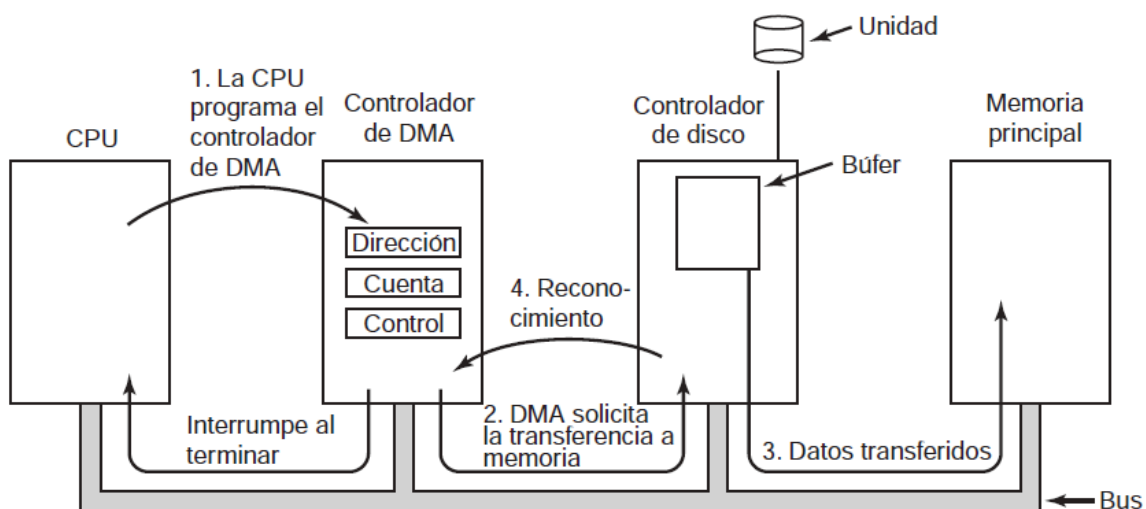


Figura 5-4. Operación de una transferencia de DMA.

Tabla 11.1. Técnicas de E/S.

	Sin interrupciones	Con interrupciones
Transferencia de E/S a memoria a través del procesador	E/S programada	E/S dirigida por interrupciones
Transferencia directa de E/S a memoria		Acceso directo a memoria (DMA)

Evolución del sistema de E/S

1. El procesador controla directamente un dispositivo periférico. dispositivos simples controlados por un microprocesador

2. Se añade un controlador o módulo de E/S. El procesador usa **E/S programada sin interrupciones**. procesador se independiza de detalles específicos de interfaces de dispositivos externos
3. Se utiliza lo mismo, pero **con interrupciones**. procesador no gasta tiempo esperando que se realice la operación. más eficiente
4. Al módulo de E/S se le da control directo de la memoria mediante **DMA**. puede mover un bloque de datos a la memoria sin involucrar al procesador. EXCEPTO AL PRINCIPIO Y EL FINAL DE LA TRANSFERENCIA
5. Se mejora el módulo de E/S, convirtiéndose en un procesador independiente con instrucciones especializadas a la E/S. procesador es interrumpido sólo cuando se termine la secuencia completa
6. El módulo de E/S tiene su propia memoria local. Es un computador por derecho propio. **Se pueden controlar un gran conjunto de dispositivos E/S con una intervención mínima del procesador**. utilizado para controlar la comunicación con terminales interactivos

Capas del software de E/S

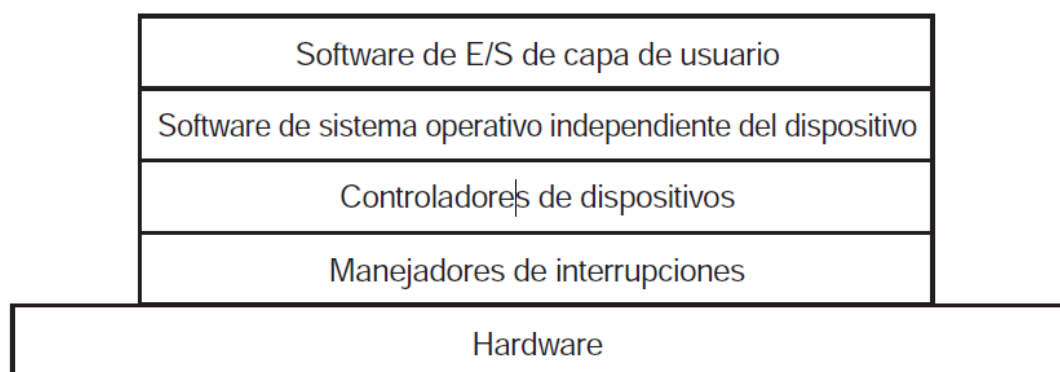


Figura 5-11. Capas del sistema de software de E/S.

Controladores de dispositivos

Cada dispositivo de E/S conectado a una computadora necesita cierto código específico para controlarlo. Este código es el **driver** y es escrito por el fabricante del dispositivo.

Pueden ser contruidos para ejecutarse en el espacio de usuario y aíslan problemas. **Por lo general**, se ejecutan en el espacio del kernel.

Un controlador de dispositivo tiene varias funciones:

- Aceptar peticiones abstractas de lectura y escritura del software independiente del dispositivo que está por encima de él, y ver que se lleven a cabo.
- Puede tener que administrar sus propios requerimientos y eventos del registro.
- Puede tener que inicializar el dispositivo.

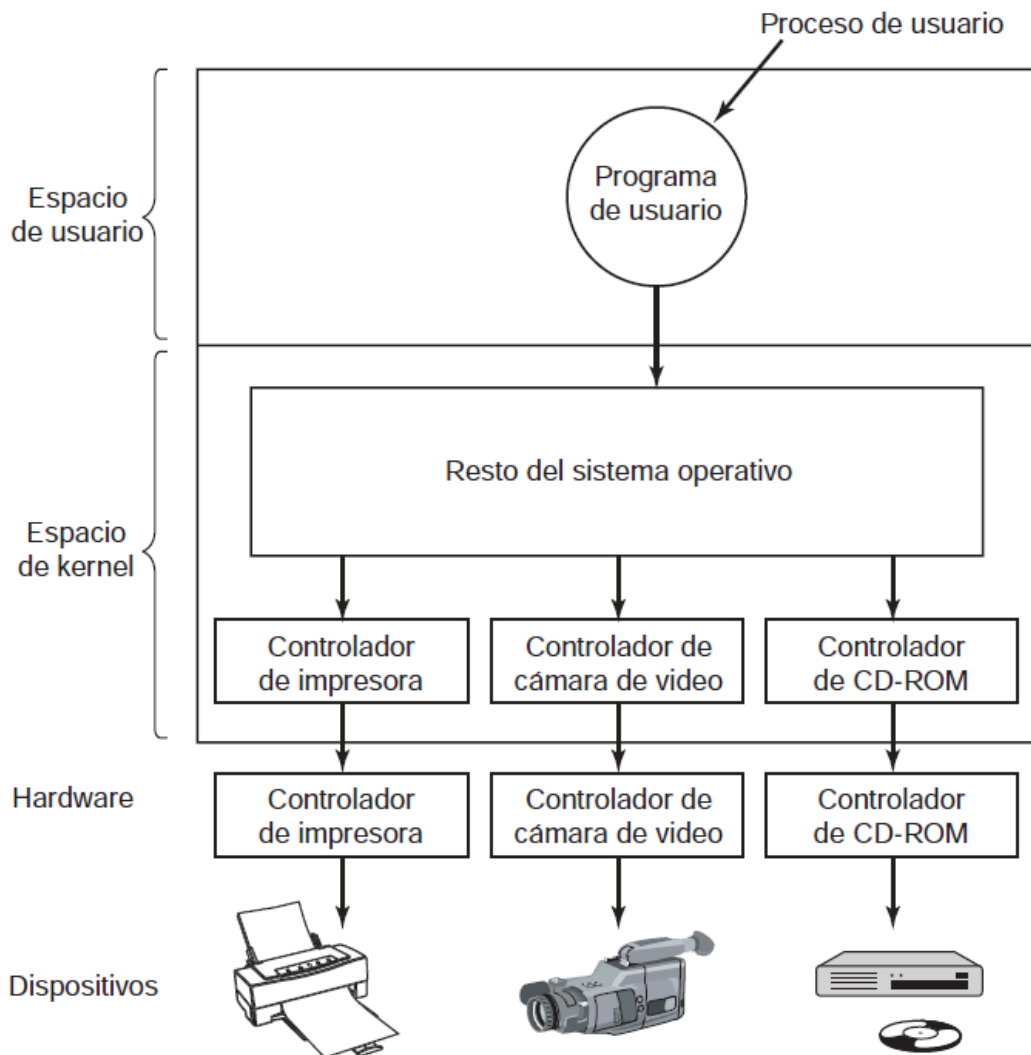


Figura 5-12. Posicionamiento lógico del software controlador de dispositivos. En realidad toda la comunicación entre el software controlador y los controladores de dispositivos pasa a través del bus.

Software de E/S independiente del dispositivo

Interfaz uniforme para controladores de dispositivos
Uso de búfer
Reporte de errores
Asignar y liberar dispositivos dedicados
Proporcionar un tamaño de bloque independiente del dispositivo

Figura 5-13. Funciones del software de E/S independiente del dispositivo.

La función básica del software independiente del dispositivo es realizar las funciones de E/S comunes para todos los dispositivos y proveer una interfaz uniforme para el software a nivel de usuario.

Interfaz uniforme para controladores de dispositivos

Hacer que todos los dispositivos de E/S y sus controladores se vean más o menos iguales

No es conveniente tener que modificar el sistema operativo para cada nuevo dispositivo.

Important

En la práctica, no todos los dispositivos son absolutamente idénticos, pero por lo general hay sólo un pequeño número de tipos de dispositivos, e incluso éstos en general son casi iguales.

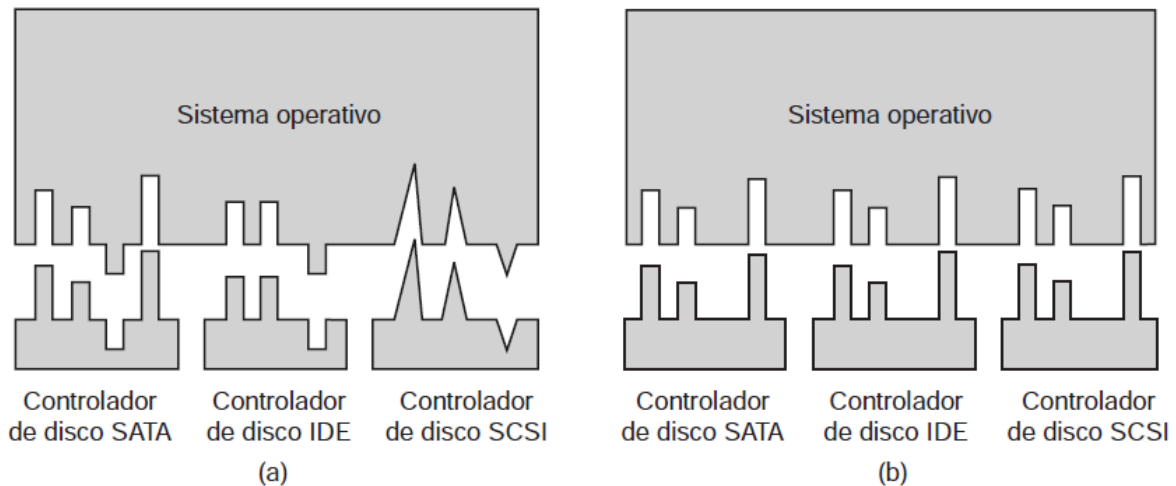


Figura 5-14. (a) Sin una interfaz de controlador estándar. (b) Con una interfaz de controlador estándar.

Otro aspecto de tener una interfaz uniforme es la forma en que se nombran los dispositivos. El software independiente del dispositivo se hace cargo de asignar nombres simbólicos.

```
/dev/disk0 ~ EJEMPLO EN UNIX
```