

## Tema 1. Arquitectura de E/S

- Aspectos básicos de la arquitectura de Entrada/Salida (E/S)
- Dispositivos e interfaces
- Buses

## Arquitectura básica de un computador

Un computador es una máquina que:

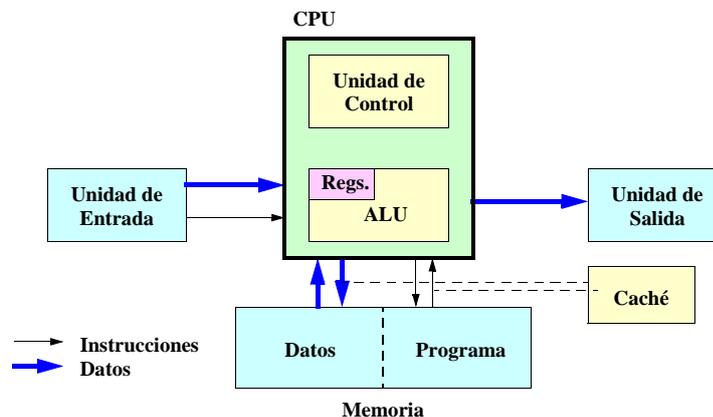
- acepta información de entrada
- la procesa ejecutando paso a paso una secuencia de instrucciones o programa
- y produce una información de salida.

El computador está por tanto compuesto por un equipo electrónico (hardware) y un conjunto de programas (software)

El computador puede realizar dos tipos de instrucciones:

- acciones
- decisiones

## Estructura de un Computador



## Rendimiento del computador



Las prestaciones de un computador no sólo dependen de la velocidad del procesador, que en muchos de ellos es el componente más rápido.

Los cuellos de botella están en:

- el intercambio de datos con la memoria:
  - se puede mejorar aumentando la anchura de los buses de datos, o mediante los sistemas de memoria jerárquicos (cachés dentro y fuera del procesador), o incluso mediante sistemas jerárquicos de buses
- el intercambio de datos con los dispositivos de entrada/salida:
  - puede llegar a ser importante en sistemas empujados en los que existe una fuerte interacción con el entorno y las operaciones de entrada/salida son muy frecuentes

## Plataformas de Tiempo Real: Dispositivos y Drivers



### Tema 1. Arquitectura de E/S

- Aspectos básicos de la arquitectura de Entrada/Salida (E/S)
- **Dispositivos e interfaces**
- Buses

## Entrada y salida de datos



Los dispositivos de entrada/salida (E/S o I/O) forman junto con la CPU y la memoria los elementos más importantes del computador

Uno de sus objetivos principales es la eficiencia en las operaciones de entrada/salida, minimizando el trabajo a realizar por la CPU

Las velocidades de los dispositivos de E/S son muy variadas:

- dispositivos lentos (p.e., ratón, teclado)
- dispositivos medios (p.e., impresora)
- dispositivos rápidos (p.e., red, disco)

Para acomodar las velocidades se usan circuitos de interfaz

# Velocidades de dispositivos de E/S

Dispositivos típicos de computadores personales y estaciones de trabajo	Velocidad de transferencia (bps)
Gigabit Ethernet	= $10^9$
Gráficos	< $6 \cdot 10^8$
Disco duro	< $10^8$
Ethernet	= $10^7$
Disco óptico	< $10^7$
Escáner	< $5 \cdot 10^6$
Impresora láser	< $2 \cdot 10^6$
Disquete	< $10^6$
Módem	< $10^5$
Ratón	< $2 \cdot 10^2$
Teclado	< $10^2$

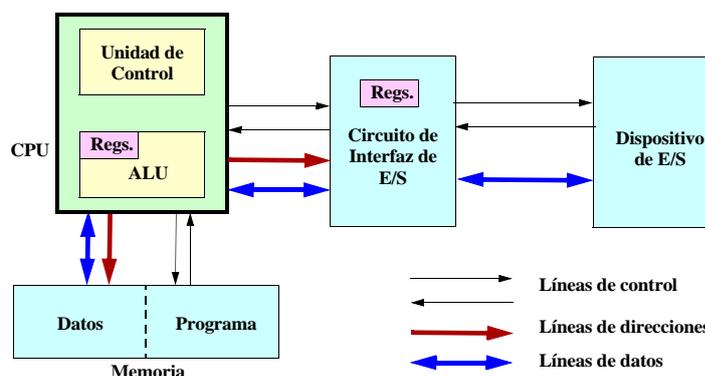
# Rendimiento y E/S

La generación actual de procesadores es capaz de manejar los datos producidos por los dispositivos, pero el principal problema está en la transferencia de datos entre el procesador y el periférico

Algunas soluciones están en:

- uso de cachés y almacenamientos intermedios
- buses de interconexión de mayor velocidad y con estructuras más elaboradas
- el uso de configuraciones multiprocesador puede ayudar también a satisfacer altas demandas de E/S

# Arquitectura de la E/S



## Arquitectura de la E/S (cont.)



### Periférico:

- Dispositivo que permite al microprocesador su interacción con el entorno

### Interfaz:

- Conjunto de módulos electrónicos que permiten el control de un periférico adaptando las diferentes velocidades y modos de funcionamiento del microprocesador y los periféricos

### Bus:

- Normalmente se refiere al conjunto de señales con las que se comunica el microprocesador con el entorno: memoria o periféricos (a través de las interfaces)

## Arquitectura de la E/S (cont.)



### Las principales razones de uso de una interfaz son:

- Existen una gran variedad de periféricos con normas de funcionamiento diferentes y resultaría imposible implementar en el procesador tal diversidad de controladores
- Las diferencias de velocidad de los dispositivos hace que no sea práctico comunicarse con ellos directamente a través del bus del sistema
- En algunos casos incluso la velocidad del dispositivo es mayor que la de la memoria o el procesador
- Los periféricos utilizan datos con formatos y tamaños de palabra diferentes de los del microprocesador al que se conectan

## Funciones de la interfaz de E/S



### Control y temporización

- de la transferencia de datos entre el dispositivo y el procesador
- si el sistema utiliza un bus cada interacción de la interfaz de E/S con el procesador puede requerir uno o más arbitrajes del bus

### Comunicación con el procesador

- decodificación de órdenes: la interfaz acepta órdenes del procesador
- datos: intercambio de datos a través del bus
- información de estado: saber si el dispositivo está preparado para transferir datos o no

## Funciones de la interfaz de E/S (cont.)



### Comunicación con los dispositivos

- intercambio de órdenes, información de estado y datos

### Almacenamiento temporal de datos

- fundamental para equilibrar las diferencias de velocidad del procesador y de los periféricos
- Ej.: el procesador puede enviar datos a ráfagas de la memoria a un dispositivo que los procesa (de su almacén temporal) a una velocidad varios órdenes de magnitud inferior

### Detección de errores

- defectos mecánicos o eléctricos en el funcionamiento del dispositivo (atasco de papel, cambio de un bit, etc.)

## Conexión de las interfaces de E/S



### Conexión mapeada en memoria

- el circuito de interfaz se conecta como si fuera memoria
- se accede a los registros leyendo o escribiendo una variable en una posición de memoria concreta

### Conexión mediante puertos de entrada/salida

- el circuito de interfaz se conecta mediante líneas especiales
- se accede a los registros mediante instrucciones especiales (*in*, *out*), especificando un número de puerto

## Tipos de E/S



### Entrada/salida por consulta o programada

- la CPU accede a los registros desde programa
- para saber si el dispositivo está listo, se hace una consulta periódica

### Entrada/salida por interrupciones

- el dispositivo avisa a la CPU cuando está listo
- la entrada/salida se hace mediante una rutina de servicio de interrupción

### Entrada/salida por acceso directo a memoria

- el dispositivo accede directamente a la memoria
- avisa a la CPU del inicio o final de la operación

## a) E/S por consulta

La operación de E/S es controlada por la CPU

Antes de realizar la operación se comprueba el (los) registro(s) de estado, para ver si el dispositivo está listo

Ventajas: sencillez

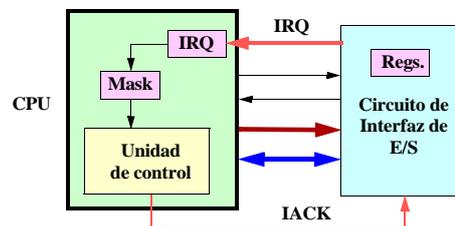
Desventajas:

- ritmo de transferencia limitado por la velocidad de la CPU
- tiempo de respuesta elevado, mayor que el periodo de consulta
- sobrecarga de la CPU para operaciones de consulta que podrían evitarse

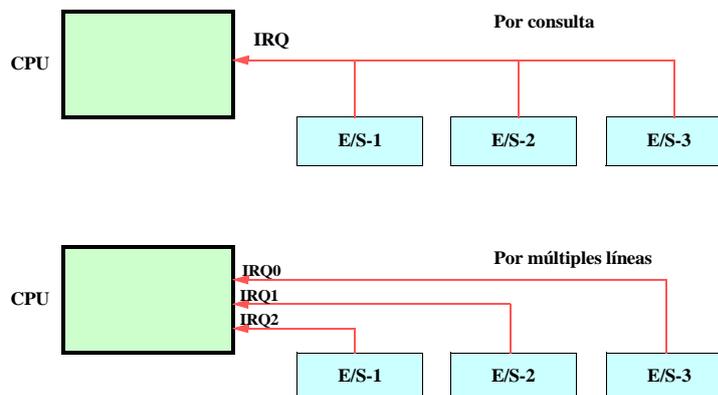
## b) E/S por interrupciones

Permite al dispositivo marcar el instante en que se hace la transferencia de datos

El mecanismo de interrupción está presente en casi todos los computadores



## Identificación de la fuente de interrupción



Las interrupciones se pueden enmascarar

- se utiliza para evitar la interrupción cuando se accede a datos compartidos con ella

Para cada interrupción se puede instalar una rutina de servicio de interrupción

- al llegar la interrupción, el procesador interrumpe el programa en ejecución y enmascara esa interrupción
- después ejecuta la rutina de servicio de interrupción
- al acabar, el procesador restaura el estado anterior y el programa interrumpido continúa

## La rutina de servicio de interrupción

Estructura habitual

- Acceder al dispositivo causante de la interrupción y hacer que cese la petición de interrupción
- Si es necesario, acceder al controlador de interrupciones para hacer lo mismo
- realizar la transferencia de datos

El modelo es el de una tarea concurrente más

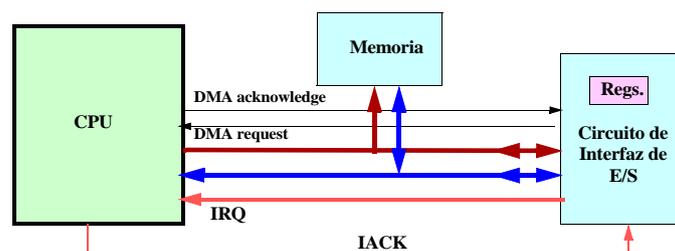
- ejecuta a la máxima prioridad del sistema

## c) E/S por acceso directo a memoria

El ritmo de transferencia es superior al de los otros métodos

El instante de E/S lo marca el dispositivo con interrupciones

Hay líneas para desconectar a la CPU de la memoria



## Funcionamiento de la E/S directa

Registros del dispositivo:

- **IODIR**: Dirección de memoria para la operación de E/S
- **CONT**: Contador de número de bytes a transferir

Funcionamiento habitual:

- La CPU carga los valores en los registros del dispositivo
- El dispositivo solicita el uso de la memoria: **DMA-request**
- La CPU se lo concede: **DMA-acknowledge**
- El dispositivo transfiere los datos. Para cada uno, incrementa **IODIR** y decrementa **CONT**
- Cuando **CONT** llega a 0 se devuelve el control de la memoria a la CPU (**DMA-request**) y se envía una interrupción de aviso

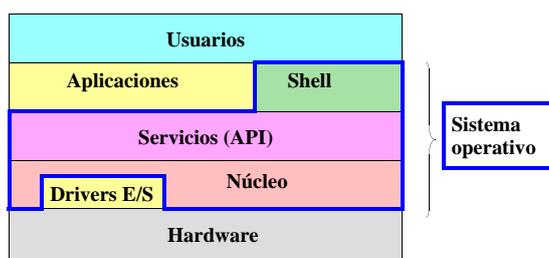
## El Sistema Operativo

Controla el uso por parte de los programas de aplicación de todos los recursos del computador: memoria, CPU, unidades de entrada y salida

- Independiza al programa de aplicación del hardware
  - incluidos los periféricos
- Proporciona comunicación con otros computadores
  - las redes forman parte de los dispositivos periféricos
- Ejecuta órdenes de los usuarios
  - controles de uso, gestión de ficheros, instalación de nuevos servicios, etc.
- Ejecuta servicios para los programas o aplicaciones
  - APIs para casi todo, concurrencia, sincronización, tiempo real, gestión de dispositivos, etc.

## El Sistema Operativo (cont.)

El control de los periféricos se realiza mediante los **drivers** de dispositivos, integrados en el núcleo



Aunque existen estándares de SO como POSIX (UNIX), los drivers no se han estandarizado

## El Sistema Operativo (cont.)



Algunos de los servicios del sistema operativo son especialmente importantes para la programación de drivers de dispositivos:

- organización de los propios drivers dentro del SO
- gestión de la memoria (intercambio de datos con las aplicaciones, o bloqueos para el uso con DMA)
- sincronización (también con los manejadores de interrupción)
- acceso al hardware (API para acceder a los puertos de E/S)
- uso de interrupciones (manejadores)
- mecanismos para aligerar el código de los manejadores de interrupción

## Clasificación de los periféricos



En la bibliografía se encuentran múltiples clasificaciones, por ejemplo:

- entrada
  - teclado, ratón, lápiz óptico, lector óptico, lector de caracteres imanables, lector de bandas magnéticas, lector de tarjetas inteligentes (*Smart Card*), lector de marcas, lector de caracteres manuscritos, lector de códigos de barras, reconocedores de voz, *joystick*, digitalizador o tabla gráfica, pantalla sensible al tacto, *scanner*
- salida
  - impresora, sintetizado de voz, pantallas, trazador de gráficos o *plotter*, monitor, microfilm, instrumentación científica o industrial.

## Clasificación de los periféricos (cont.)



- unidades de memoria masiva auxiliar
  - cinta magnética, disco magnético, disco óptico, CD-ROM, DVD.
- mixtos: los que no son puros de entrada o salida ni de memoria
  - terminal interactivo, terminal teletipo, pantalla sensible al tacto, lectora/perforadora de tarjetas, módem.
  - ¿y las comunicaciones?

Otra posible clasificación en función de la distancia:

- locales: se encuentran cerca del ordenador
- remotos: si debido a su situación lejana la conexión hay que realizarla a través de líneas especiales de transmisión

W. Stallings distingue sólo dos grupos:

- Memoria externa: como parte del sistema de memoria
  - discos magnéticos
  - discos ópticos (CDs, DVDs)
  - cintas magnéticas, etc.
- Dispositivos de E/S:
  - de interacción con humanos: comunicación usuario computador
  - de interacción con máquinas: comunicación con elementos del equipo
  - de comunicación: para el acceso a equipos remotos

*En cualquier caso los periféricos hacen lo que hacen, y lo que nos debe preocupar es el modo en que se conectan al sistema y en el que pueden ser controlados desde las aplicaciones.*

## Algunas observaciones

Nos damos cuenta de que la variedad de los dispositivos y sus diferentes modos de funcionamiento son tales que resulta difícil incluso hacer una clasificación en la que alguno no se queda fuera de los patrones previstos.

Las interfaces permiten controlar el funcionamiento del dispositivo a través de alguno de los buses del sistema.

Las interfaces se programan normalmente en el nivel del sistema operativo a través de los drivers:

- no ha llegado a converger a un estándar
- cada SO lo hace a su manera

## Algunas observaciones (cont.)

Dificultades en la programación de dispositivos:

- La labor de programación de drivers no es del todo "reutilizable" en el sentido por ejemplo de los sistemas operativos, lenguajes de programación, u otros niveles mayores de abstracción del software:
  - aunque el dispositivo funcione igual como mínimo habrá que modificar la interfaz software, y las estructuras de datos
- En muchas ocasiones la información sobre el funcionamiento del hardware es difícil de conseguir:
  - se ofrece a entornos concretos que son cerrados
  - en entornos abiertos se puede hacer ingeniería inversa

Esto sólo significa que este nivel de programación requiere un esfuerzo extra: ¡No hay que desanimarse por ello!

## Tema 1. Arquitectura de E/S

- Aspectos básicos de la arquitectura de Entrada/Salida (E/S)
- Dispositivos e interfaces
- **Buses**

## Principales buses de interconexión en PCs modernos



- **Bus del procesador**
  - llamado FSB (*Front-Side Bus*) es el de mayor velocidad del sistema
  - comunica el procesador con la caché, la memoria principal, o el *North Bridge* del *chipset*
  - velocidades en MHz de: 66, 100, 133, 200, 266, 400, 533, 800, 1066
- **Bus AGP**
  - el bus AGP (*Accelerated Graphics Port*) tiene 32 bits y está destinado a tarjetas gráficas
  - velocidades en MHz de: 66 (AGPx1), 133 (AGPx2), 266 (AGPx4), ó 533 (AGPx8)
  - permite anchos de banda de hasta 2133 MBps
  - se conecta al *North Bridge* o a un *Hub* controlador de la memoria
  - en los sistemas más modernos está en desuso en favor del PCI-Express

## Principales buses de interconexión en PCs modernos (cont.)



- **PCI-Express**
  - tercera generación del bus PCI (*Peripheral Component Interconnect*)
  - puede ser generado por el *North Bridge* o el *South Bridge*
  - su velocidad se describe en términos de carriles (*lanes*)
  - cada carril bidireccional (*dual-simplex*) soporta un ritmo de transferencia de 2.5 Gbps en cada dirección (2 Gbps efectivos), se conoce como *slot* PCI-Express x1
  - algunos sistemas soportan PCI-Express x4 (10 Gbps en cada dirección)
  - las tarjetas gráficas generalmente usan el *slot* PCI-Express x16 (40 Gbps en cada dirección)

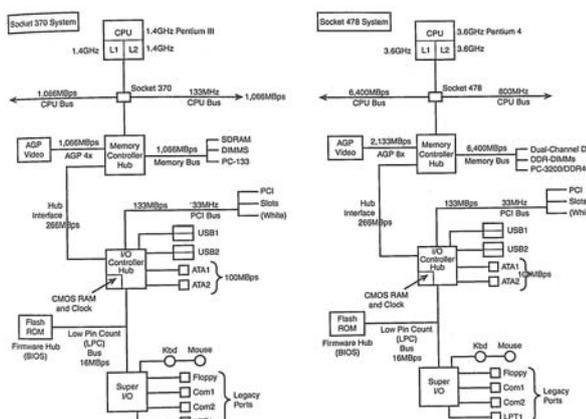
## Principales buses de interconexión en PCs modernos (cont.)

- **PCI-X**
  - segunda generación del bus PCI (*PCI Extended*)
  - soporta **slots** de 64 bits compatibles con PCI de 32 y 64 bits
  - la velocidad en la versión 1 es de 133 MHz y en la 2.0 hasta 533 MHz
  - el ancho de banda en la versión 2.0 se divide entre los **slots** PCI-X y PCI
  - aunque algunos **South Bridge** puede generar PCI-X, la mayoría de los **chipset** usan un chip aparte
- **Bus PCI**
  - bus de 32 bits a 33 MHz desde los primeros Intel 486
  - opcionalmente hay versiones de 66 MHz y 64 bits
  - puede ser generado por el **North Bridge**, el **South Bridge**, o **Hubs** controladores de I/O
  - usado para periféricos rápidos

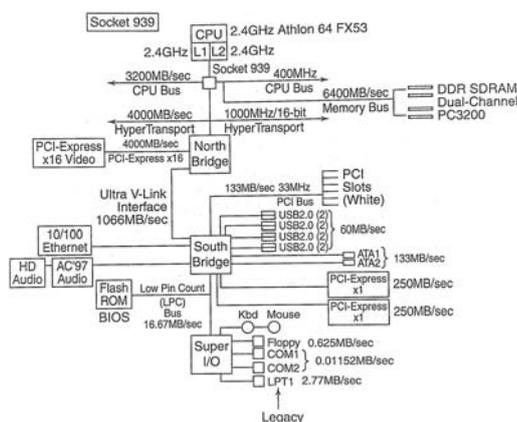
## Principales buses de interconexión en PCs modernos (cont.)

- **Bus ISA**
  - bus de 16 bits a 8 MHz (el original de 1984 8 bits a 5 MHz)
  - desaparece en los sistemas más modernos
  - utilizado para periféricos lentos: modems o tarjetas de sonido
  - se genera en el **South Bridge** de la placa base, que actúa como controlador del bus ISA y como interfaz con el bus PCI (más rápido)
  - el **chip Super I/O** normalmente se conecta al bus ISA en los sistemas con estos **slots**
  - el **chip Super I/O** normalmente integra dispositivos que estaban separados en tarjetas de expansión en sistemas antiguos:
    - controlador de disquetera
    - uno o dos controladores de puertos serie
    - controlador de puerto paralelo

## Arquitectura de buses Pentium III/ Celeron y Pentium 4



S. Mueller [1]



S. Mueller [1]

## Otras arquitecturas de buses



- **PC/104 es un estándar de ordenador embebido o empotrado que define el formato de la placa base y el bus del sistema:**
  - está diseñado para aplicaciones específicas, como adquisición de datos o sistemas de control industrial
  - La arquitectura de la placa base no es la típica placa de circuitos integrados (*backplane*) en la que se insertan los componentes
  - los componentes se encuentran en módulos que son apilados unos encima de otros
  - una instalación típica incluye una placa base, convertidores analógico-digital y módulos I/O digitales
  - versiones
    - PC/104: original con un bus ISA de 104 pines
    - PC/104-Plus: incorpora el bus ISA-104 y un bus PCI con 120 pines
    - PCI-104: incluye sólo el bus PCI con 120 pines

## Otras arquitecturas de buses (cont.)



- **Bus CAN (Controller Area Network)**
  - protocolo de comunicaciones desarrollado por la firma alemana Robert Bosch GmbH para automoción
  - basado en una topología bus para la transmisión serie de mensajes en ambientes distribuidos
  - protocolo normalizado, con lo que se simplifica y economiza la tarea de comunicar subsistemas de diferentes fabricantes sobre una red común o bus
  - el procesador delega la carga de comunicaciones a un periférico inteligente, por lo que dispone de mayor tiempo para ejecutar sus propias tareas
  - al ser una red multiplexada, reduce considerablemente el cableado y elimina las conexiones punto a punto
  - se puede ver como una interfaz de comunicaciones

- **Bus I2C (Inter-Integrated Circuit)**
  - bus de comunicaciones serie diseñado por Philips
  - velocidad de 100Kbps en el modo estándar, aunque también permite velocidades de 3.4 Mbps
  - muy usado en la industria, principalmente para comunicar microcontroladores y sus periféricos en sistemas empotrados
  - generalizado más para comunicar circuitos integrados entre sí que normalmente residen en un mismo circuito impreso
  - sólo usa dos hilos para transmitir la información: por uno van los datos y por otro la señal de reloj que sirve para sincronizarlos (hay un tercer hilo de masa que no siempre es necesario)

## A modo de conclusión

En sistemas como los PCs en los que puede haber varios buses interconectados (incluso sistemas jerárquicos) parte de la complejidad en el uso de los mismos puede quedar oculta por el sistema operativo.

En sistemas más sencillos (algunos empotrados) es posible que no exista SO, o que esté limitado a funciones muy básicas.

En cualquier caso es fundamental conocer la arquitectura del sistema y sus modos de funcionamiento:

- buses de interconexión
- las interfaces y su programación
- funcionalidad del periférico

## Bibliografía

- [1] Scott Mueller, "Upgrading and Repairing PCs", 17th Ed., QUE, 2006
- [2] William Stallings, "Organización y arquitectura de computadores", 7ª Ed., Pearson, 2006.