

Arquitectura de Computadoras

CURSO 2022

Turno:
Clase 3

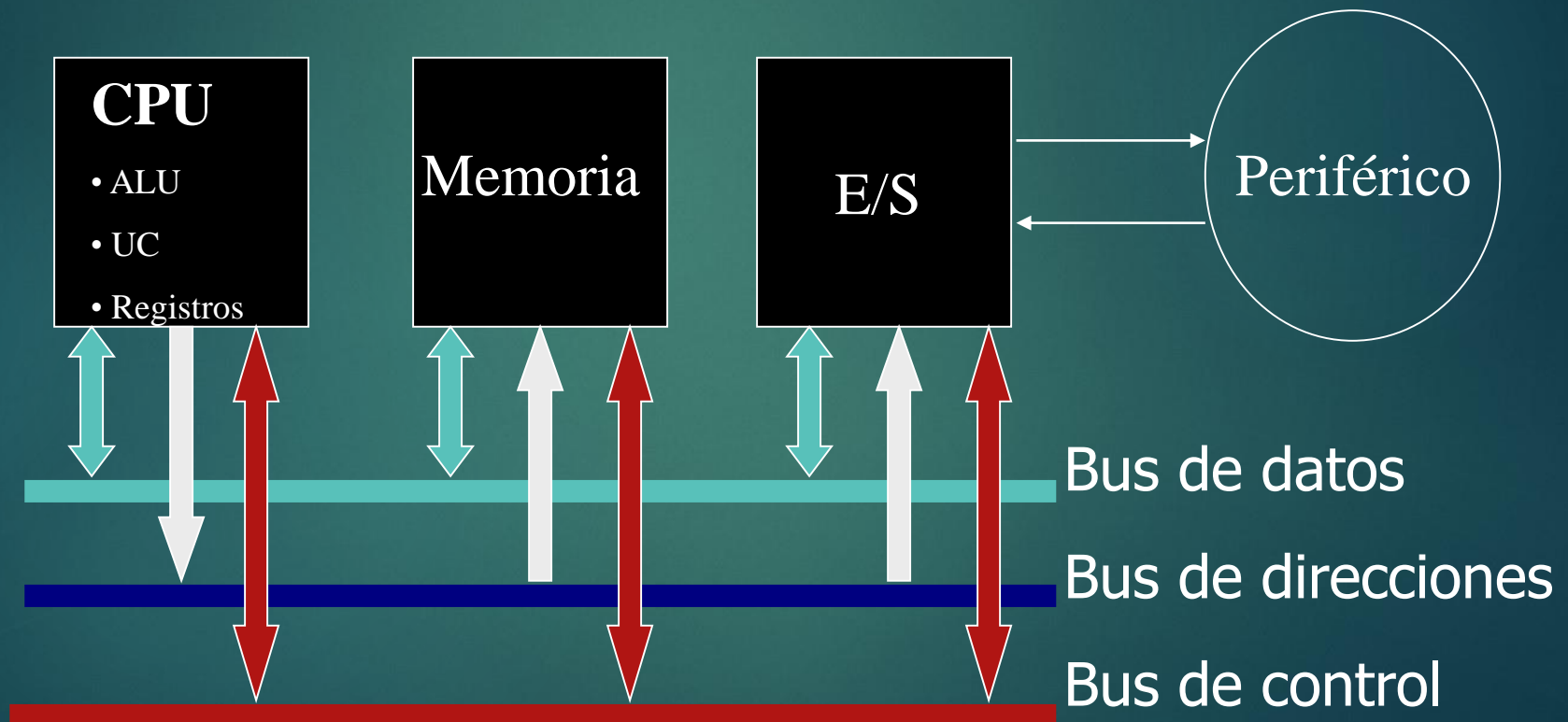
Resumen clase 3

Subsistema Entrada/salida

- Concepto básicos del Subsistema de E/S
- Módulo de E/S
- Tipos de puertos de E/S
- Organización interna de los puertos de E/S
- Acceso al subsistema de E/S
- Gestión de las transferencia de E/S
- Eficiencia en las técnicas de gestión de transferencias de E/S
- DMA – DMAC y canales de E/S

SUBSISTEMA E/S

En la primera clase de este curso se presentó el modelo de Arquitectura de von Neumann basado en 3 subsistemas: CPU, Memoria y E/S.



SUBSISTEMA E/S

El subsistema de E/S comprende los dispositivos que están conectados al bus del sistema y proveen los servicios de transferencia de datos con los Periféricos.

Debido a la gran variedad de Periféricos con los que se requiere intercambiar información, el subsistema de E/S tiene que ser lo suficientemente flexible para permitir:

- Transmisión de diferentes cantidades de datos
- Rango de velocidades de transmisión muy amplio.
- Diferentes formatos de dato y tamaño de palabra.

En general, todos los Periféricos son más lentos que la CPU y la Memoria. Los dispositivos que forman parte del subsistema de E/S permiten descongestionar el trabajo de la CPU.

SUBSISTEMA E/S

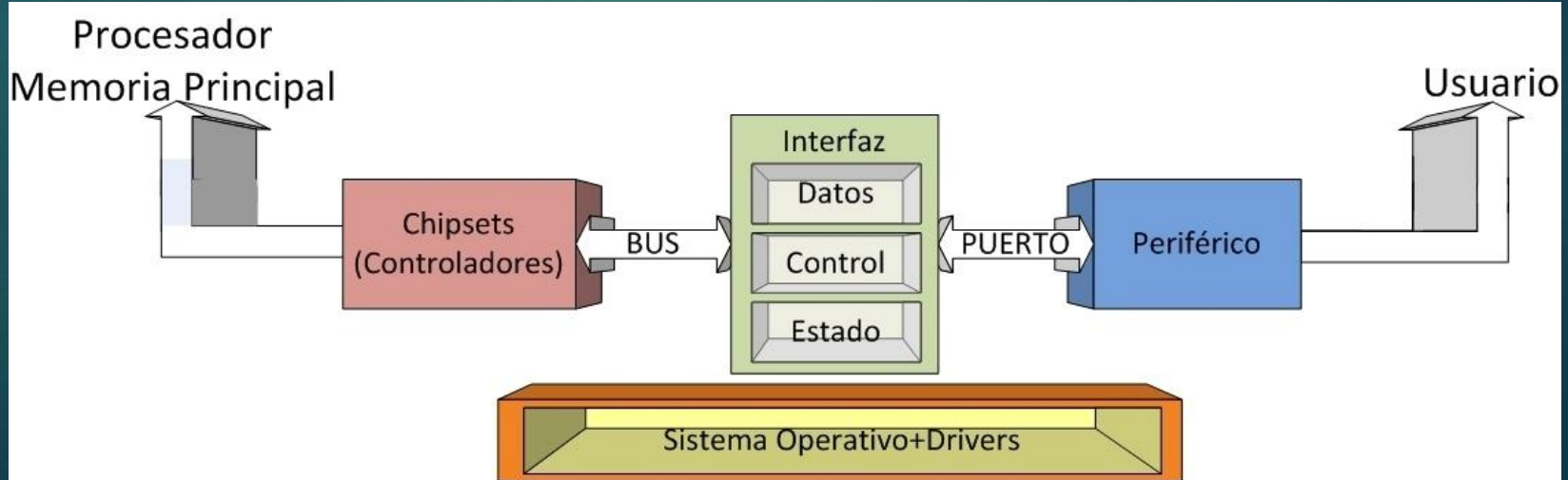
Dispositivos periféricos

- Existe una variedad muy grande de equipos periféricos. Algunos de los tipos más comunes son:
 - Comunicación hombre-máquina: monitor/pantalla, mouse, teclado
 - Almacenamiento: disco duro, CD, DVD
 - Impresión: impresora, escáner
 - Comunicación con dispositivos remotos: modem, placa de red
 - Multimedia: micrófono, parlantes
 - Automatización y control: sensores, alarmas, adquisición de datos

SUBSISTEMA E/S

Módulo de interfaz de E/S

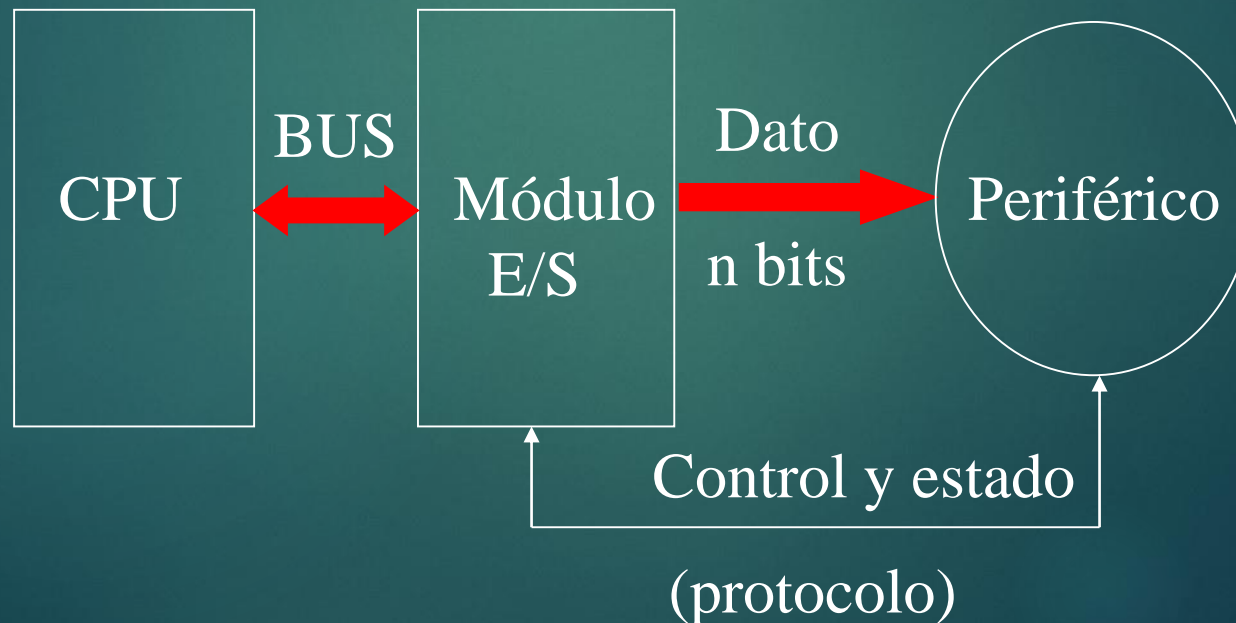
- Son los dispositivos más sencillos para implementar las transferencias de E/S con periféricos.
- Los puertos de E/S conectan la interfaz entre el procesador/memoria y un periférico.
- Son, por lo común, administrado por el SO a través de drivers específicos.



Módulo de E/S

Módulo de E/S

- El esquema básico de transferencia de datos usando un Módulo de E/S sencillo es el que se muestra en la figura siguiente.



Módulo de E/S

Módulo de E/S

- En la figura anterior se puede observar el Módulo de E/S comunicado con la CPU a través del Bus del Sistema.
- La conexión con el Periférico provee 2 tipos de informaciones:
 - 1.- Datos: Información útil a transferir
 - 2.- Control y Estado: información que permite realizar la transferencia (en lo posible libre de errores). Por ejemplo: sentido de la transferencia (de entrada, de salida), operación (de lectura, de escritura), estado del periférico (listo, no-listo, en falla).

Módulo de E/S

9

Módulo de E/S

➤ El Módulo de E/S debe ejecutar 2 tipos de comunicaciones:

1.- Hacia el Periférico (comunicaciones externas con la periferia):

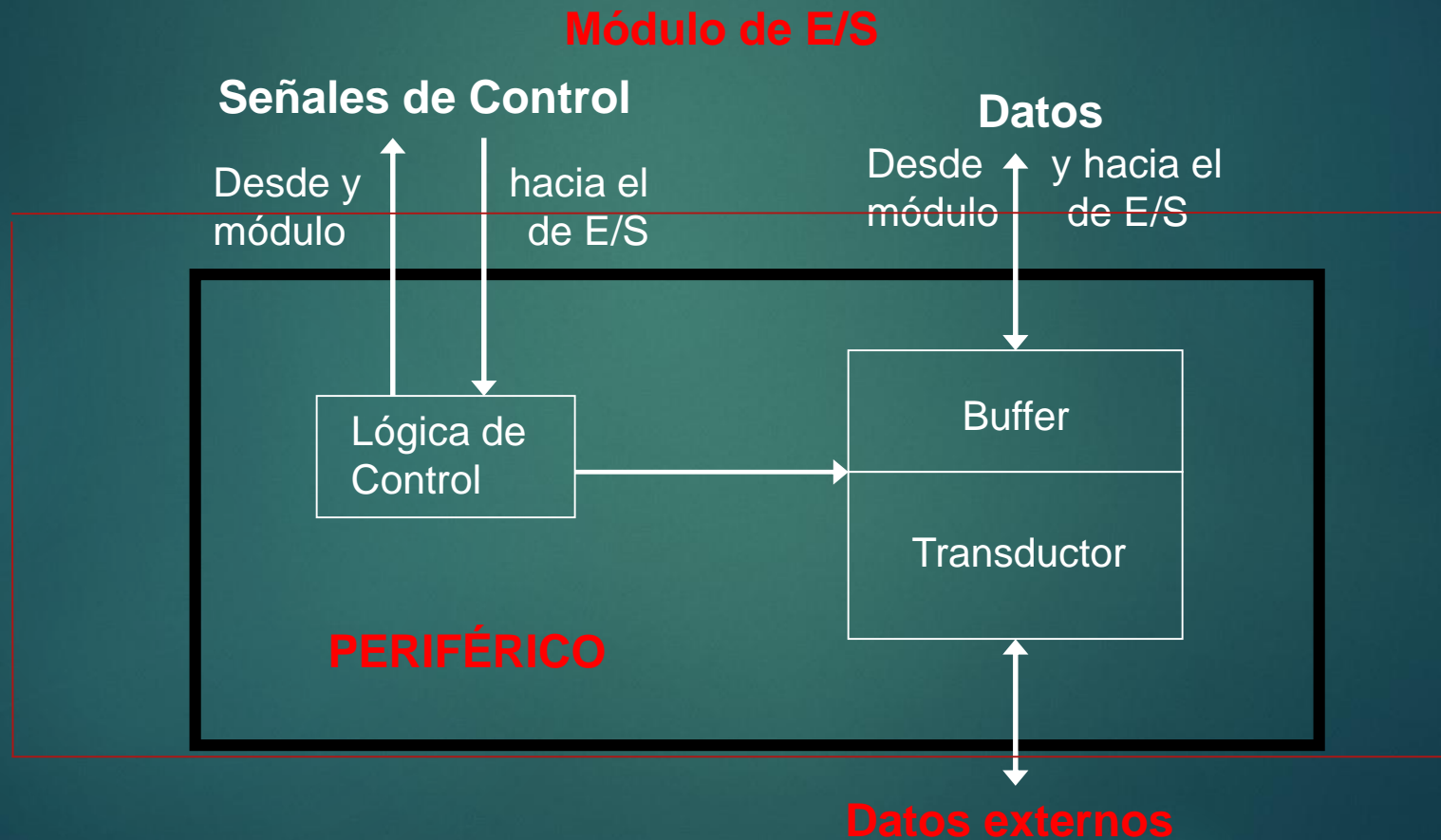
- Transferir datos con el periférico (incluye adaptación eléctrica).
- Controlar y temporizar uno o más dispositivos externos.
- Almacenar temporalmente datos (“buffer”).
- Detectar errores.

2.- Hacia el Bus (internas con CPU y Memoria):

- Interpretar las órdenes que recibe de CPU y transmitirlas al periférico.
- Transferir datos con la CPU (registros) y Memoria.
- Informar a la CPU del estado del periférico.

Módulo de E/S

Esquema simplificado de un Periférico conectado



Módulo de E/S

Módulo de E/S

➤ En el esquema anterior se puede apreciar que un periférico tiene 2 bloques funcionales que manejan la comunicación entre el periférico y el módulo de E/S:

1.- Hacia el exterior del periférico:

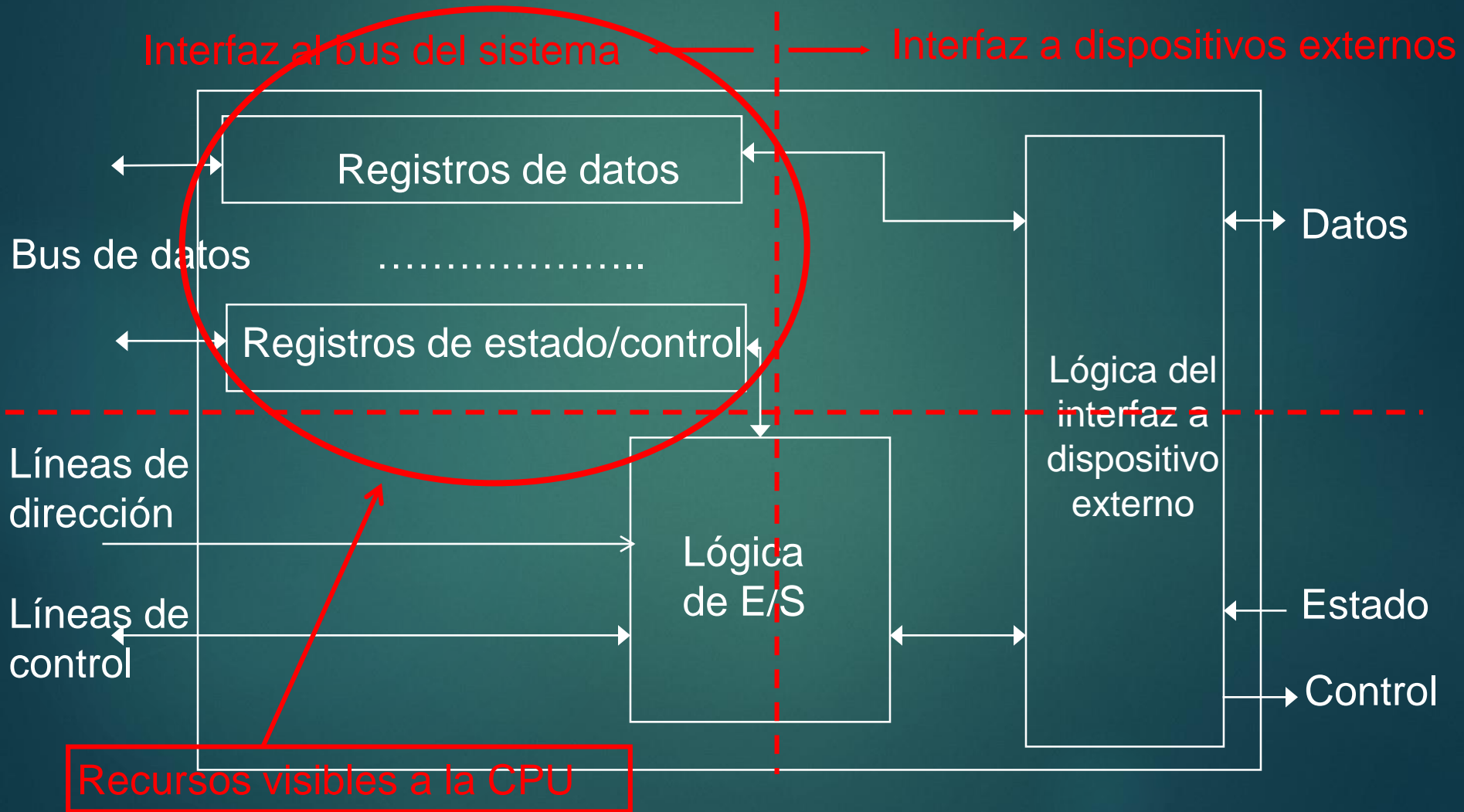
- Sección de manipulación de datos (buffer/trasductor). Almacena y convierte los datos a intercambiar con el periférico
- Sección de control y estado (Lógica de control): recibe y genera las señales de control y estado del periférico

2.- Hacia el Módulo de E/S :

- Sección de manipulación de datos
- Sección de manipulación de señales de control y de estado

Módulo de E/S

Esquema detallado de un Módulo de E/S



Módulo de E/S

13

Esquema detallado de un Módulo de E/S

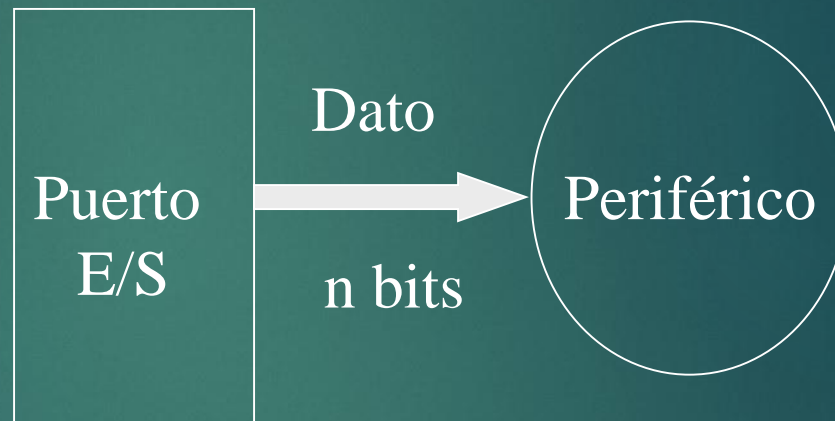
- En el esquema anterior se pueden apreciar los bloques funcionales que manejan la comunicación hacia el periférico y hacia el bus del sistema.
- Hacia el sistema los recursos que son “visibles” a la CPU (y al programador) son básicamente registros. Existen 2 tipos de registros:
 - Registros de datos: que contienen la información útil recibida desde, o transmitida a, el periférico.
 - Registros de control y estado: que controlan las características de la transferencia, y almacenan señales de estado de la comunicación (falla, no listo, etc.).

Puerto de E/S

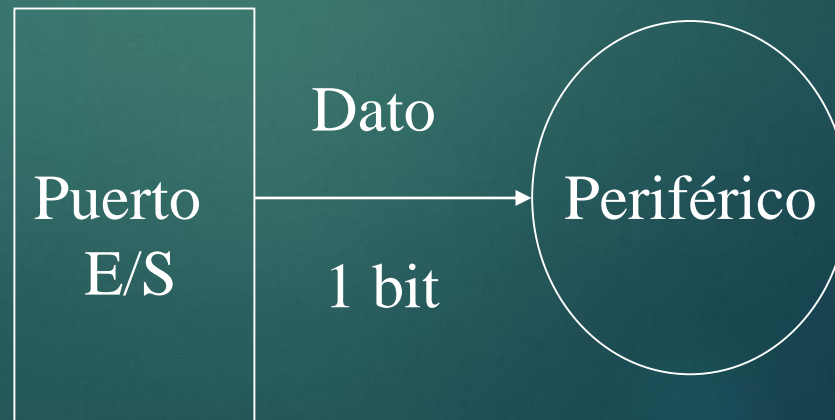
Tipos de Puerto de E/S básicos

Existen 2 tipos básicos: serie y paralelo.

PARALELO



SERIE



Puerto de E/S

15

Tipos de Puerto de E/S básicos

- Puerto paralelo: hay varias líneas de datos (n) que transfieren n bits simultáneamente entre el puerto de E/S y el periférico.
 - Ejemplos: impresora paralelo, scanner
- Se requiere disponer de una conexión mediante un cable que incluya al menos los n bits de datos, lo que lo hace un método bastante costoso.

Puerto de E/S

16

Tipos de Puerto de E/S básicos

- Puerto serie: hay 1 línea de dato para la transferencia entre el puerto de E/S y el periférico.
 - Ejemplos: impresora serie, red Ethernet, mouse, teclado, etc.
- Se requiere disponer de una conexión mediante un cable sencillo. El costo es mucho menor.
- Los datos deben serializarse (transmitirse de 1 bit por vez, uno a continuación del otro) lo que en teoría sería mucho más lento que transmitir de a n bits simultáneamente, como lo hace el puerto paralelo.

CPU y E/S

17

Subsistema de E/S

Las Puertas de E/S son un tipo particular de dispositivo de E/S. Hay otros dispositivos más complejos que proveen de otras prestaciones, además de la tarea básica de implementar la transferencia de los datos, por ejemplo:

- Ocultar las propiedades particulares del dispositivo periférico a la CPU: temporizados, formatos, electromecanismos, etc.
- Manejar múltiples dispositivos simultáneamente.
- Controlar varias funciones del dispositivo

CPU y E/S

Registros de un Puerto de E/S

Desde el punto de vista de la CPU, una operación de E/S requiere acceder a los registros internos del Módulo de Interfaz de E/S. Los registros pueden ser de lectura y/o escritura.

- Dentro de Módulo de E/S hay 2 tipos de registros:
 - de DATOS: interviene en la transferencia de entrada o de salida del dato a intercambiar entre el Sistema de cómputo y el periférico.
 - de CONTROL: registros que controlan y registran el funcionamiento del módulo, la transferencia, y el periférico.

CPU y E/S

Registros de un Puerto de E/S

- de Datos: la transferencia de un dato entre el Sistema de cómputo y el periférico consiste básicamente en:
 - Operación de entrada: lectura de un registro de dato de entrada (es decir un registro escrito por el periférico y leído por la CPU)
 - Operación de salida: escritura de un registro de dato de salida (es decir, un registro escrito por la CPU y leído por el periférico).
- de Control y Estado: la supervisión y/o control de la transferencia requiere de:
 - Control: adecuar la configuración del módulo para ajustar formatos, sincronizaciones, etc.
 - Estado: registrar el estado operativo del módulo y del periférico.

CPU y E/S

Acceso al Subsistema de E/S

Desde el punto de vista de la CPU, el Subsistema de E/S está compuesto por un conjunto de registros a los que accede para una operación de entrada o de salida.

Existen 2 técnicas de acceso a estos registros:

- Espacio de E/S compartido con memoria (memory-mapped)
- Espacio de E/S separada de la memoria

CPU y E/S

Espacio de E/S compartido con memoria (memory-mapped)

- En esta técnica los registros de los dispositivos de E/S y memoria comparten un único espacio de direcciones.
- Los registros de la E/S se comportan idéntico a una memoria de lectura/escritura.
- No hay instrucciones específicas para E/S, se usan las mismas instrucciones de movimiento de datos a memoria.

Ej: Mov Reg_dato, AL

Donde Reg_dato es la dirección de un registro de salida del módulo de E/S (dirección idéntica a la correspondiente a una posición de memoria).



CPU y E/S

22

Espacio de E/S separado de memoria (modelo Intel)

- En esta técnica los registros de los dispositivos de E/S y la memoria están en diferentes espacios de direcciones.
- Dado que el bus de direcciones es compartido por la memoria y el subsistema de E/S, se requieren señales de control adicionales para identificar a donde está accediendo la CPU: a la memoria, o a la E/S.
- Hay instrucciones específicas de E/S, distintas de las instrucciones de acceso a la memoria.



CPU y E/S

23

Espacio de E/S separado de memoria (modelo Intel)

- El procesador dispone de instrucciones específicas para acceder a los registros que están en el subsistema de E/S.
- Cuando se ejecutan estas instrucciones específicas, en el bus de control se identifica el acceso al mapa de direcciones de E/S. Para el resto de las instrucciones en el bus de control se identifica el acceso a la Memoria.

Ejemplos:

- 1) Entrada: IN dest, fuente

Donde dest es AL o AX (8 o 16 bits) y fuente un número de 8 bits sin signo (0 y 255) o DX (número entre 0 y 65535).

- 2) Salida: OUT dest, fuente

Donde fuente es AL o AX (8 o 16 bits) y dest un número de 8 bits sin signo (0 y 255) o DX (número entre 0 y 65535).

Gestión de la transferencia

Desde el punto de vista de la gestión para transferir datos entre el Sistema de cómputo y el periférico, existen 3 estrategias básicas de implementación:

- E/S Programada y espera de respuesta
 - E/S Programada y administrada por interrupción
 - E/S con acceso directo a memoria (DMA)
- Las 2 primeras opciones requieren intervención directa de la CPU, es decir, que la CPU participa en la transferencia de todos los datos (byte o word) a transferir.

Gestión de la transferencia

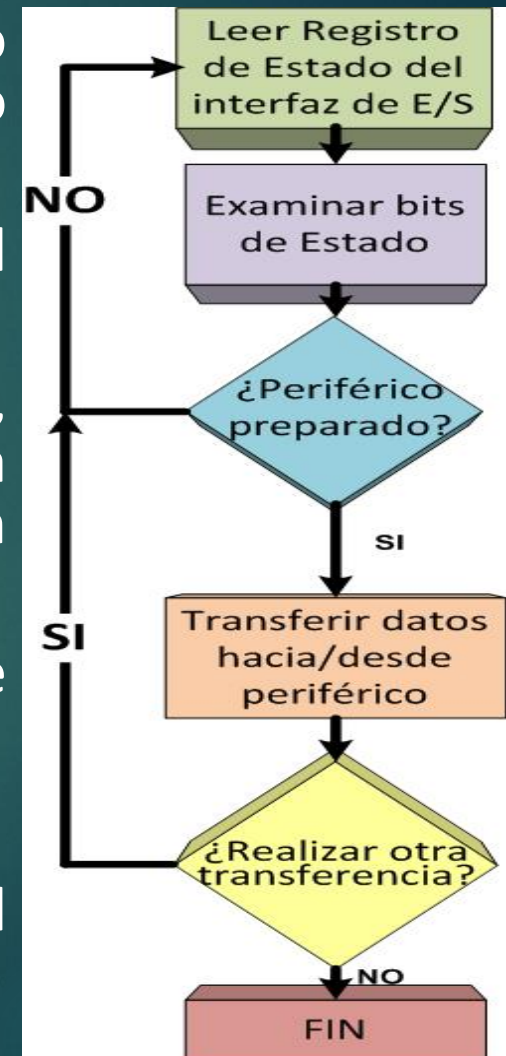
E/S Programada y espera de respuesta

- La CPU interviene directamente en la transferencia de cada unidad de información (byte, word) con el módulo.
- Es decir que la CPU tiene control casi directo sobre la operación de E/S. Entre otras acciones hace:
 - Comprueba el estado del dispositivo
 - Envía los comandos requeridos (por ejemplo de lectura, escritura)
 - Realiza la transferencia de todos los datos (de a uno)
- En cada dato que es transferido, la CPU espera que el módulo E/S termine la operación, típicamente que el periférico “acepte” el dato.
- Durante la espera, la CPU permanece ociosa (no deseable).

Gestión de la transferencia

La secuencia de acciones que ejecuta la CPU son :

1. La CPU verifica el estado de periférico (preparado/no-preparado) leyendo un registro del módulo de interfaz.
2. Examina el estado del periférico chequeando el bit (o bits) que identifican dicho estado.
3. Si el dispositivo no está listo (por ejemplo bit=0), la CPU vuelve al paso 1. Este lazo significa que la CPU “espera” hasta que el periférico se pone en “preparado”, es decir, listo para la transferencia.
4. Cuando el dispositivo está listo, la CPU transfiere 1 dato hacia o desde el módulo de interfaz.
5. Si hay más datos que transferir vuelve al paso 1.
- 6.- Si se completó la transferencia, termina el servicio de E/S.



Gestión de la transferencia

27

E/S Programada y administrada por interrupción

- La CPU interviene directamente en la transferencia de cada unidad de información (byte, word) con el módulo.
- Cada vez que el módulo está listo (o completó una transferencia) avisa a la CPU con un pedido de interrupción.
- Ahora la CPU no tiene que ejecutar el lazo de comprobación del estado del módulo (pasos 1, 2 y 3 de la figura anterior). Solo inicia la transferencia al recibir el pedido de interrupción del periférico.
- Durante el tiempo que el periférico no está listo, la CPU no tiene que esperar, puede seguir ejecutando otra tarea.

Gestión de la transferencia

28

E/S Programada y administrada por interrupción

A continuación se describen la secuencia de acciones para una transferencia de entrada de 1 dato (por ejemplo, 1 carácter) administrado por interrupción.

- La CPU inicia la operación de lectura (entrada) enviando una orden de lectura (READ) al módulo de E/S.
- El módulo de E/S solicita el dato al periférico.
- El periférico busca el dato, mientras la CPU continúa con sus tareas.
- Cuando el módulo E/S tiene el dato enviado por el periférico emite un pedido de interrupción a la CPU.
- La CPU detecta el pedido, interrumpe el proceso, y bifurca al servicio de la interrupción.
- Durante la interrupción, la CPU lee el dato desde el módulo de E/S.
- La CPU retorna a la tarea interrumpida.

Gestión de la transferencia

29

Interrupciones en operaciones de E/S

Cuando hay varios dispositivos periféricos, con esta forma de administración de las transferencias de E/S se requiere poder identificar la fuente de interrupción.

Como se vió durante el análisis de Interrupciones, existen varias estrategias distintas para identificar la fuente de la interrupción:

- 1) Diferentes líneas de interrupción
- 2) Una sola línea de interrupción y encuesta por software
- 3) Una sola línea de interrupción con conexión en cadena (daisy chain) tipo “hard poll” (encuesta por hardware)
- 4) Una sola línea de interrupción y vectorizado

Gestión de la transferencia

1) Diferentes líneas de interrupción

- Se dispone de una línea de interrupción por cada dispositivo.
- Es sencilla de implementar.
- Hay una limitación en la cantidad de dispositivos a conectar debido a la cantidad restringida de señales de interrupción que puede manejar la CPU.

2) Una sola línea de interrupción y encuesta por software

- Se dispone de 1 sola línea de interrupción para todos los dispositivos.
- Cuando ocurre el pedido de interrupción, la CPU tiene que consultar a cada dispositivo (es decir, a cada módulo de E/S) para determinar quien fue el demandante.
- Este esquema de encuesta por software (también conocido como “polling”) puede resultar sumamente lento.

Gestión de la transferencia

31

3) Una sola línea de interrupción con conexión en cadena

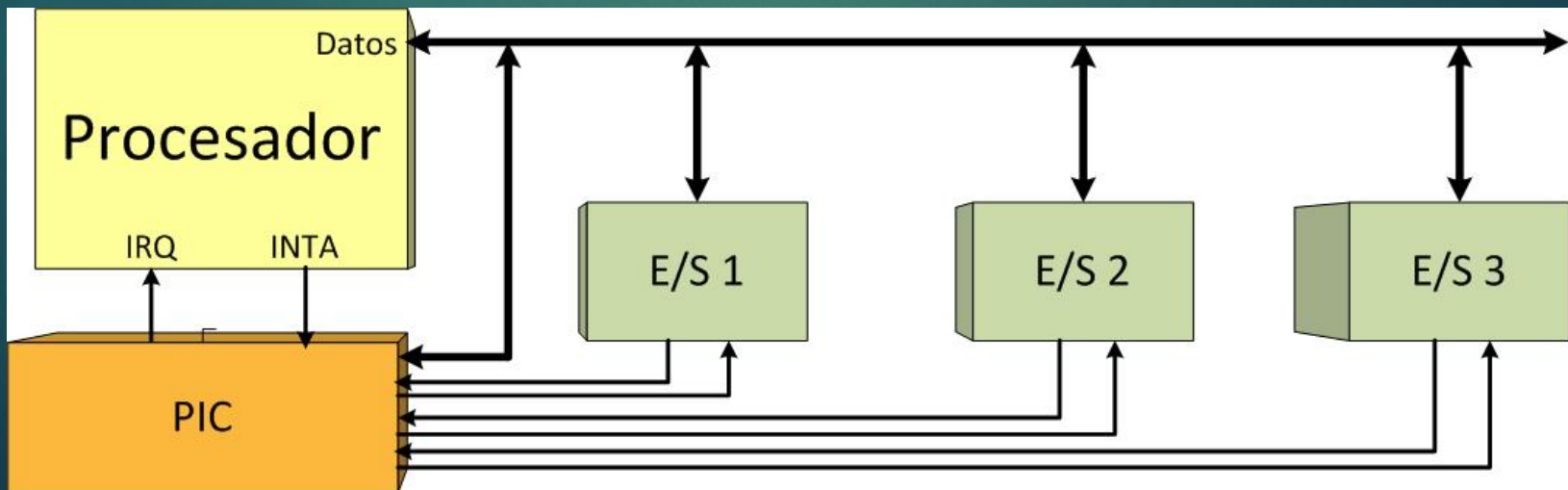
- Se dispone de 1 sola línea de interrupción INTR para todos los dispositivos.
- La línea de respuesta de la CPU INTA (reconocimiento de interrupción) se conecta encadenadamente a todos los módulos (conexión tipo “margarita” o “Daisy chain”).
- Una vez enviada la confirmación de parte de la CPU, el módulo que está más adelante (más próximo a la CPU) en la conexión Daisy chain responderá colocando un vector (palabra), en el bus, que lo identifica.
- Si hay otros pedidos de interrupción más abajo del que respondió, deberán esperar la terminación del servicio del que respondió.

Gestión de la transferencia

32

4) Una sola línea de int. y vectorizado

- Se dispone de 1 sola línea de interrupción INTR para todos los dispositivos.
- Un controlador dedicado (PIC) provee el vector que identifica la fuente de interrupción.
- Las líneas de interrupción tienen un orden de prioridad, las líneas con más prioridad pueden interrumpir a las líneas con menor prioridad.
- Si existe un maestro del bus, solo él puede interrumpir.



Gestión de la transferencia

33

En general:

- Las operaciones de E/S administradas por interrupción son más eficientes que las programadas con espera.
- Ambas técnicas requieren la intervención directa de la CPU.
- Al tener que intervenir la CPU en la transferencias de los datos se presentan 2 problemas:
 - La velocidad de transferencia depende de la capacidad de la CPU de atender estas tareas. Aunque la CPU es muy rápida, puede ser que en determinadas circunstancias no sea capaz de administrar varias transferencias simultáneamente.
 - La CPU puede permanecer ocupada mucho tiempo durante la operación, sin poder hacer otras tareas.
- Además, si el volumen de datos a transferir es grande, el tiempo de ocupación de la CPU crece también.

Gestión de la transferencia

34

Para analizar el comportamiento de la CPU en transferencias de E/S, vamos a analizar la respuesta de la CPU en 2 casos:

- 1) Transferencia hacia un periférico lento (impresora de 20 páginas/minuto)
- 2) Transferencia hacia un periférico rápido (disco de 10Mb/s)

Para los 2 casos, se analizará la respuesta del sistema para transferencias de E/S manejadas por la CPU con espera de respuesta y administrada por interrupción.

Gestión de la transferencia

Sobre la CPU

- Se tiene un procesador con un reloj de 200 MHz (período del reloj = 5 ns).
- El procesador, en promedio, necesita 2 ciclos de reloj por instrucción. Este parámetro se conoce como CPI. En este caso $CPI=2$
- Es decir, que una instrucción tarda, en promedio:

$$\text{Ciclo de instrucción} = 2 \times 5 \text{ ns} = 10 \text{ ns} = 10^{-8}$$

- El procesador tiene, por lo tanto, una capacidad de procesamiento de:

Nº de instrucciones por segundo = $1 / \text{tiempo de ciclo de instrucción}$

Nº instrucciones por segundo = $1/10^{-8}$

Nº Instrucciones por segundo = 100 millones de instrucciones por segundo (100 Mips)

Gestión de la transferencia

36

Caso 1: Transferencia a una Impresora

- Se tiene que imprimir (operación de salida) un archivo de 10 Kbytes en una impresora láser de 20 páginas por minuto.

- Considerando que:

1 página tiene (estimado) 3.000 letras (es decir, 3000 caracteres)

y:

1 carácter = 1 byte

- La impresora imprime:

$20 \text{ ppm} = 20 \text{ pag/min} \times 3000 \text{ car/pag} = 60.000 \text{ caracteres por minuto}$

Es decir, que la velocidad de transferencia de la impresora es de :

$V_t = 60000 \text{ car./min} = 60000 \text{ car./60 seg.} = 1000 \text{ car./s} = 1 \text{ Kbyte/s}$

Gestión de la transferencia

37

Caso 1: Transferencia a una impresora

a) E/S con espera de respuesta

- La CPU entra en un bucle y envía un nuevo byte cada vez que la impresora está preparada para recibirlo
- Si la impresora tarda 1 seg en imprimir 1 Kbyte (ya que la tasa de transferencia es de 1 kbyte/seg), necesitará:

Tiempo total de transferencia = 10 seg para 10 Kbytes.

- Conclusión:

La CPU está ocupada con la operación de E/S durante 10 seg.

- Observación:

Ya que la velocidad de la CPU es de 100MIPs, en ese tiempo la CPU podría haber ejecutado:

Nºinstrucciones en 10 seg. = 1000 millones de instrucciones

Gestión de la transferencia

38

Caso 1: Transferencia a una impresora

b) E/S administrada con interrupciones

- La impresora genera una interrupción cada vez que está preparada para recibir un nuevo byte.
- Si la gestión de la interrupción (que se llama ATI) requiere 10 instrucciones (entre las que se incluyen salvar contexto, comprobar estado, transferir byte, restaurar contexto, y retornar), entonces:

Para transferir 10 Kbytes se requiere ejecutar 10.000 veces la ATI.

- El tiempo que dura el servicio de la transferencia es, aproximadamente, el requerido por las 10.000 ATI. Recordando que 1 instrucción tiene un tiempo de ejecución de 10^{-8} seg

$$\begin{aligned}\text{Tiempo total de transferencia} &= 10.000 \text{ ATI} = 10.000 \times 10 \text{ instr} = \\ &= 10.000 \times 10 \times 10^{-8} \text{ seg} = 100.000 \times 10^{-8} \text{ seg} = 10^{-3} \text{ seg} = 0,001 \text{ seg}\end{aligned}$$

- Conclusión:

La CPU está ocupada con la operación de E/S durante 0,001 seg.

Gestión de la transferencia

39

Conclusiones Caso 1 - Transferencia a una impresora:

- De los resultados anteriores en el uso de la CPU para el caso de una transferencia con un dispositivo relativamente lento, usando las 2 técnicas de E/S, con espera y administrada por interrupción, se pueden sacar las siguientes conclusiones.
- La transferencia dura 10 segundos, porque es el tiempo que tarda la impresora en imprimir los 10.000 caracteres.
- El tiempo de uso de la CPU en ambos casos es:
 - Caso a) con espera: 10 seg.

La CPU está los 10 segundos ocupada administrando la transferencia.

- Caso b) administrada por interrupción: 0,001 seg

De los 10 segundos que dura la transferencia, la CPU le ocupa solo 0,001 segundo en administrarla.

- Es decir, que la E/S por interrupciones reduce en 10.000 veces el tiempo que la CPU está ocupada gestionando la impresora, y por lo tanto, es mucho más eficiente.

Gestión de la transferencia

Caso 2: Transferencia a un disco

- Se tiene que transferir un archivo de memoria a disco de 10 Mbytes

- Considerando que:

El disco posee una velocidad de transferencia de 10 MB/s (1 byte cada 10^{-7} seg ó 100 nanoseg)

significa que:

Tiempo total de transferencia= 1 segundo.

Gestión de la transferencia

41

Caso 2: Transferencia a un disco

a) E/S con espera de respuesta

- La CPU entra en un bucle y envía un nuevo byte cada vez que el disco está preparado para recibirlo
- Si el disco transfiere 10 Mbytes/seg, necesita 1 seg para recibir un archivo de 10 Mbytes.

Tiempo total de transferencia = 1 seg para 10 Mbytes.

- Conclusión:

La CPU está ocupada con la operación de E/S durante 1 seg.

- Observación:

Ya que la velocidad de la CPU es de 100MIPs, en ese tiempo la CPU podría haber ejecutado:

Nºinstrucciones en 1 seg. = 100 millones de instrucciones

Gestión de la transferencia

42

Caso 2: Transferencia a un disco

b) E/S administrada con interrupciones

- El disco genera una interrupción cada vez que está preparado para recibir un nuevo byte
- Si la gestión de la interrupción (que se llama ATI) requiere 10 instrucciones (entre las que se incluyen salvar contexto, comprobar estado, transferir byte, restaurar contexto, y retornar), entonces:

Para transferir 10 Mbytes tenemos que ejecutar 10^7 veces la ATI

- El tiempo que dura el servicio de la transferencia es, aproximadamente, el requerido por las 10.000.000 ATI. Recordando que 1 instrucción tiene un tiempo de ejecución de 10^{-8} seg

$$\begin{aligned}\text{Tiempo total de transferencia} &= 10.000.000 \text{ ATI} = 10^7 \times 10 \text{ instr} = \\ &= 10^7 \times 10 \times 10^{-8} \text{ seg} = 1 \text{ seg}\end{aligned}$$

- Conclusión:

La CPU está ocupada con la operación de E/S durante 1 seg.

Gestión de la transferencia

43

Conclusiones Caso 2 – Transferencia a un disco:

- De los resultados anteriores en el uso de la CPU para el caso de una transferencia con un dispositivo relativamente rápido, usando las 2 técnicas de E/S, con espera y administrada por interrupción, se pueden sacar las siguientes conclusiones.
- La transferencia dura 1 segundo, porque es el tiempo que tarda el disco en transferir los 10Mbytes.
- El tiempo de uso de la CPU en ambos casos es:
 - Caso a) con espera: 1 seg.

La CPU está 1 segundo ocupada administrando la transferencia.

- Caso b) administrada por interrupción: 1 seg

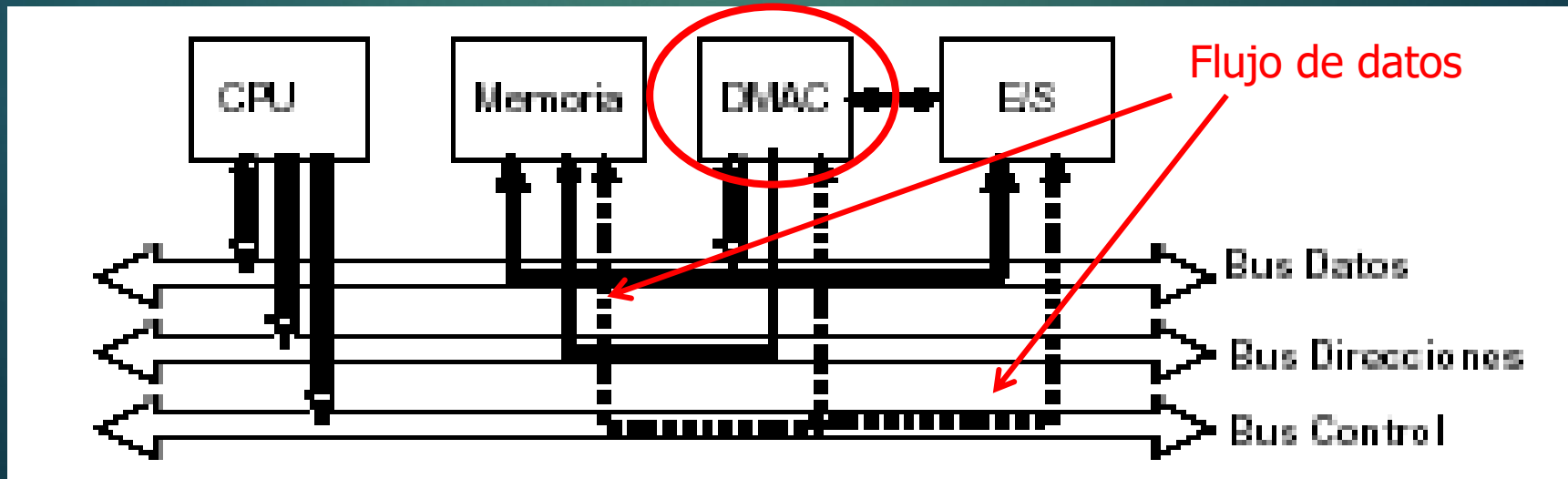
La CPU está 1 segundo ocupada administrando la transferencia.

- Es decir, que no hay diferencia entre las 2 técnicas. En ambas la CPU está ocupada 1 segundo el 100% del tiempo.
- Si la velocidad del dispositivo fuera mayor, la CPU no podría hacerla.

Acceso directo a memoria (DMA)

44

- El Acceso Directo a Memoria (DMA) es una técnica de transferencia de datos entre periférico y Memoria sin intervención directa de la CPU.
- Comúnmente llevada a cabo por un “Controlador de DMA” (o DMAC) específico encargado de llevar a cabo la transferencia.
- Físicamente está ubicado como se muestra a continuación.



Acceso directo a memoria (DMA)

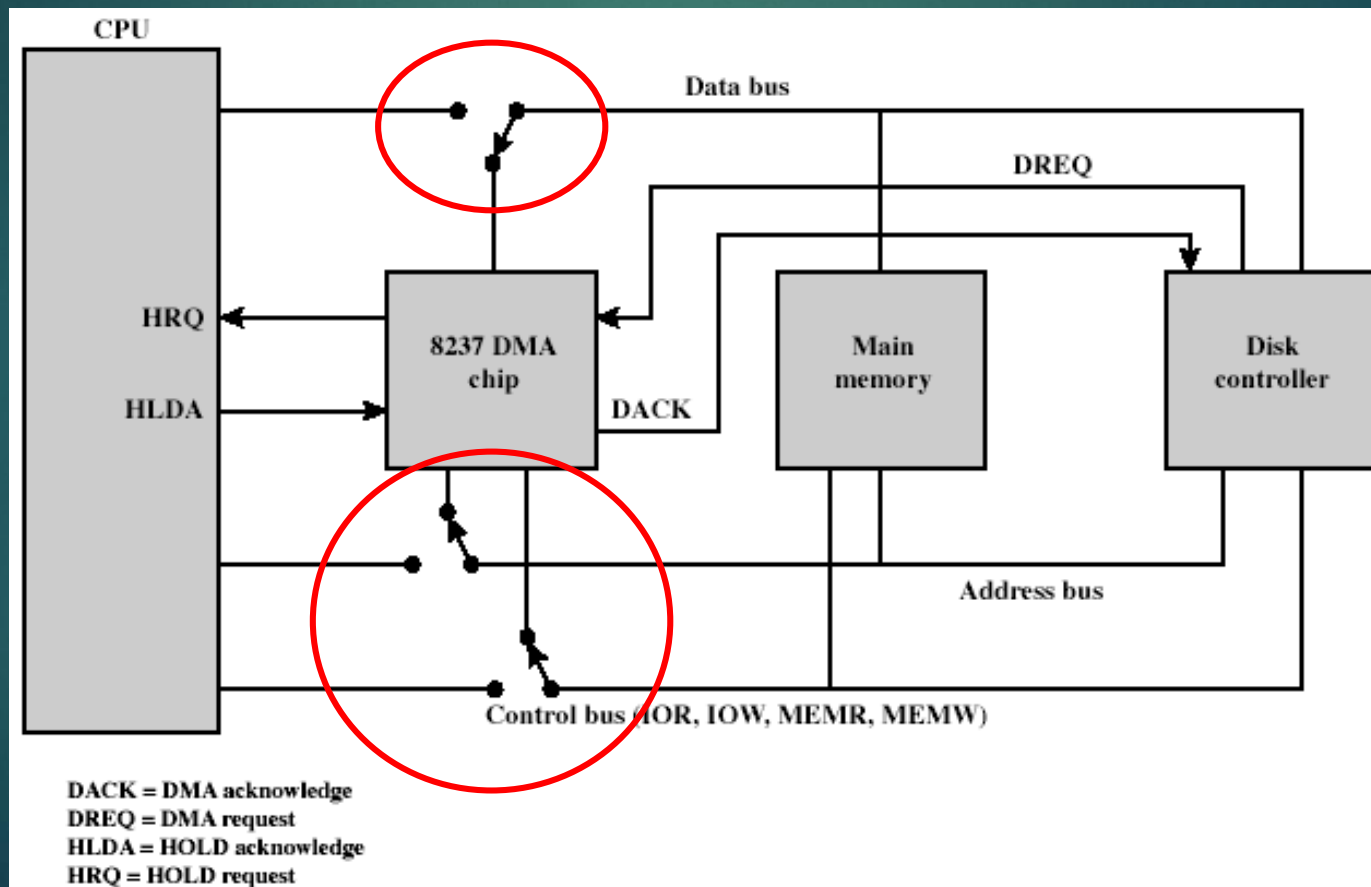
45

- Dado que la transferencia por DMA requiere el uso del Bus, tanto el DMAC como la CPU pueden tomarlo. El DMAC y la CPU “compiten” por el uso del Bus.
- Cuando el DMAC toma el Bus, actúa como “master” durante la transferencia por DMA, y debe ser capaz de:
 - Solicitar el uso del bus mediante las señales y la lógica de arbitraje necesarias
 - Especificar la dirección de memoria sobre la que se realiza la transferencia
 - Generar las señales de control del bus
 - Especificar el tipo de operación (lectura/escritura)
 - Generar las señales de sincronización de la transferencia

Acceso directo a memoria (DMA)

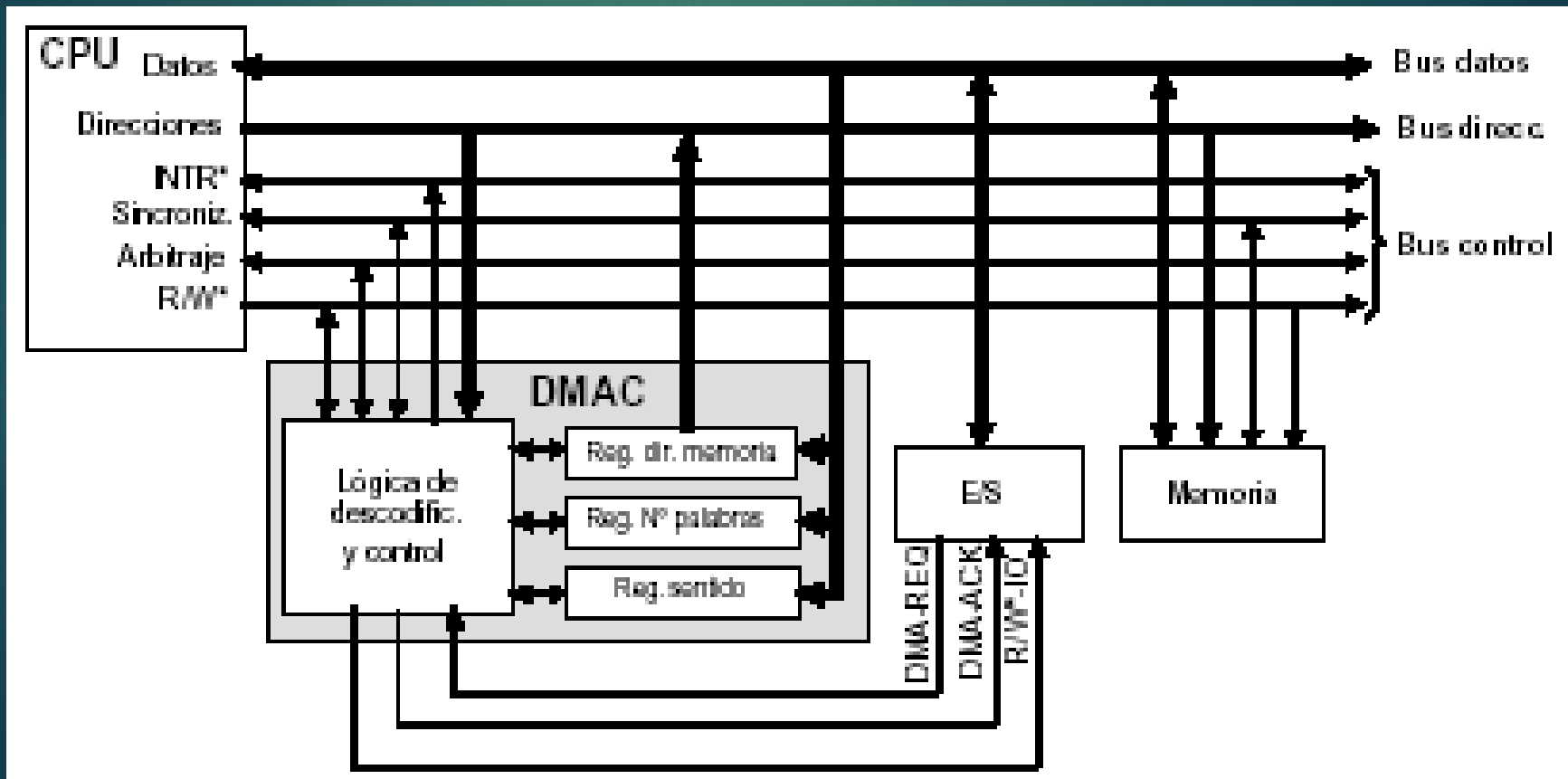
46

- Cuando la CPU entrega el bus al DMAC (8237 DMA chip en la figura), se desconecta lógicamente del mismo, y es el DMAC el que toma el control del bus.



Acceso directo a memoria (DMA)

- En la figura siguiente se muestra un detalle de la forma en que se conecta al Bus del sistema y de su estructura interna.



Acceso directo a memoria (DMA)

- El proceso de transferencia requiere realizar una serie de acciones o fases relativamente complejas. Las principales son:
 - 1.- Fase de Inicialización
 - 2.- Fase de ejecución de la transferencia
 - 3.- Fase de finalización y análisis de la transferencia

Acceso directo a memoria (DMA)

➤ 1.- Fase de inicialización

En la fase de inicialización la CPU debe configurar el módulo de E/S y el DMAC con los parámetros de la transferencia.

➤ Inicialización interfaz de E/S:

- Tipo de transferencia (lectura/escritura)
- Configuración del periférico
- Otra información de control para el periférico (por ejemplo si es un disco se especifica el número de pista, sector, etc.)

➤ Inicialización DMAC

- Nº de bytes o palabras a transferir
- Tipo de transferencia (lectura/escritura)
- Dirección de memoria inicial para la transferencia
- Otra información la transferencia.

Acceso directo a memoria (DMA)

50

➤ 2.- Fase de ejecución de la transferencia

- Cuando el periférico está listo, pide al DMAC iniciar la transferencia mediante una señal física.
- Cuando el DMAC recibe el pedido del periférico, pide el control del bus mediante alguna señal especial a la CPU. La CPU típicamente dispone de algunas señales destinadas a implementar las transferencias por DMA.
- Cuando reconoce el pedido de DMA, la CPU entrega (libera) el bus y se “desconecta” (ya no controla el bus) lógicamente del mismo.
- La CPU avisa al DMAC que liberó el bus mediante otra señal especial.
- Al liberar la CPU el bus, el DMAC toma el control del bus y ejecuta la transferencia hasta terminarla.

Acceso directo a memoria (DMA)

51

- El DMAC avisa al periférico que puede iniciar la transferencia.
- El periférico comienza a transferir los datos, a través de bus, con la memoria, de a uno por vez.
- La transferencia implica que:
 - Bus master: DMAC + Periférico - Bus slave: Memoria
- Después de la transferencia de cada palabra se actualizan los registros del DMAC:
 - Nº de bytes faltantes (o cuenta de los que se transfirieron)
 - Próxima dirección de memoria a donde guardar el dato (anterior o posterior de la corriente)
- Cuando el número de bytes faltantes es igual a 0 significa que transfirió todos los datos y terminó la transferencia.

Acceso directo a memoria (DMA)

52

- 3.- Finalización de la transferencia
 - Una vez que termina la fase 2, el DMAC libera el bus y le avisa a la CPU por medio de una señal física.
 - La CPU retoma el control del bus.
 - El DMAC suele activar, además, una señal de interrupción para indicar a la CPU la finalización de la operación de E/S solicitada.
 - La CPU, mediante la interrupción, verifica el resultado de la transferencia vía los registros internos del DMAC. Algunos resultados a verificar son:
 - Transferencia OK/fallida?
 - Errores? Tipo de errores?
 - Estado periférico

Acceso directo a memoria (DMA)

53

Ventajas e inconvenientes del DMA

- La principal ventaja es la eficiencia, dado que la CPU se libera de tener que controlar la transferencia de los datos. Solo prepara la transmisión, y verifica el resultado de la misma.
- La principal desventaja se origina en el uso del bus. Como las transferencias por DMA pueden tener mayor prioridad que la CPU, se puede degradar el rendimiento de la CPU si el DMAC hace uso intensivo del bus.

Acceso directo a memoria (DMA)

54

Ventajas e inconvenientes del DMA

- Sin embargo, no necesariamente la CPU necesite todo el tiempo el bus porque:
 - En el caso de Computadoras con memoria caché:
 - La mayor parte del tiempo, la CPU lee instrucciones de la cache, por lo que no necesita usar el bus de memoria.
 - El DMAC puede aprovechar estos intervalos en los que la CPU está leyendo instrucciones de la cache (y por tanto no usa el bus de memoria) para realizar las transferencias.
 - En el caso de computadores sin cache
 - El procesador no utiliza el bus en todas las fases de la ejecución de una instrucción.
 - El DMAC puede aprovechar las fases de ejecución de una instrucción en las que la CPU no utiliza el bus.

Acceso directo a memoria (DMA)

55

Técnicas de transferencia por DMA

Hay varias formas distintas de implementar las transferencias por DMA. Vamos a ver 2:

- Por ráfagas (burst)
- Por robo de ciclo (cycle-stealing)

Acceso directo a memoria (DMA)

56

Método de transferencia de DMA por ráfaga

- Es la vista hasta ahora. El DMAC solicita el control del bus a la CPU.
- Cuando la CPU concede el bus, el DMAC no lo libera hasta haber finalizado la transferencia de todo el bloque de datos completo.
- VENTAJAS:

La transferencia se realiza de forma muy rápida, limitada por la velocidad del periférico.

➤ DESVENTAJAS:

Durante el tiempo que dura la transferencia la CPU no puede utilizar el bus con memoria, lo que puede degradar el rendimiento del sistema.

Acceso directo a memoria (DMA)

57

Método de transferencia de DMA por robo de ciclo

- El DMAC solicita el control del bus a la CPU.
- Cuando la CPU concede el bus al DMAC, se realiza la transferencia de una única palabra y después el DMAC libera el bus.
- El DMAC solicita el control del bus tantas veces como sea necesario hasta finalizar la transferencia del bloque completo. El uso del bus se reparte entre la CPU y el DMAC.

➤ VENTAJAS:

No se degrada tanto el rendimiento del sistema y de la CPU.

➤ DESVENTAJAS:

La transferencia puede tardar un poco más de tiempo.

Acceso directo a memoria (DMA)

58

Método de transferencia de DMA por robo de ciclo

- Notar que dado que la toma y liberación del bus por parte de la CPU no es una interrupción, es decir que el procesador no debe guardar el contexto (es decir, no está interrumpiendo su tarea).
- Si bien el trabajo de la CPU es más lento (que si no estuviera presente la transferencia por DMA), no será tanto como si ella tuviera que estar desconectada del bus todo el tiempo.
- En general, para transferencia de E/S de múltiples palabras, la técnica por robo de ciclo es la más eficiente, ya que permite implementar la transferencia por DMA al mismo tiempo que la CPU continúa trabajando en su tarea.

Canales de E/S

- Las transferencias de E/S se pueden dividir, en función de la capacidad para interactuar con los periféricos, en varios niveles:
 - Nivel 1: CPU + Módulo de Interfaz de E/S o controlador: la CPU controla directamente los periféricos e interfaz, y administra la transferencia por programa (con espera).
 - Nivel 2: CPU + Módulo de Interfaz E/S o controlador con interrupción: la CPU controla directamente los periféricos y administra la transferencia con programa e interrupciones.
 - Nivel 3: DMA (DMAC + Módulo de E/S): la CPU no interviene directamente, solo prepara y supervisa la transferencia.
 - Nivel 4: Canal de E/S básico (Procesador básico + módulo de E/S): la CPU interviene mínimamente.
 - Nivel 5: Canal de E/S inteligente (Procesador inteligente + módulo de E/S): la CPU no interviene, excepto situaciones especiales.

Canales de E/S

60

- En el nivel más alto de la escala de transferencias de E/S están los canales de E/S, que representan una extensión al concepto de DMA.
- Los canales de E/S tienen la habilidad de ejecutar programas de servicios de E/S, lo que les permite tener un completo control de la transferencia de datos. La CPU no ejecuta las instrucciones de E/S, las realiza el procesador incluido en el canal. El programa que ejecuta el procesador interno del canal está almacenado en la memoria principal.
- La CPU solamente interviene para iniciar la transferencia, y dar la orden de ejecutar el programa de E/S que está en memoria. El programa de servicio de E/S especifica dispositivos, áreas de memoria a usar, prioridades y acciones ante errores.

Canales de E/S

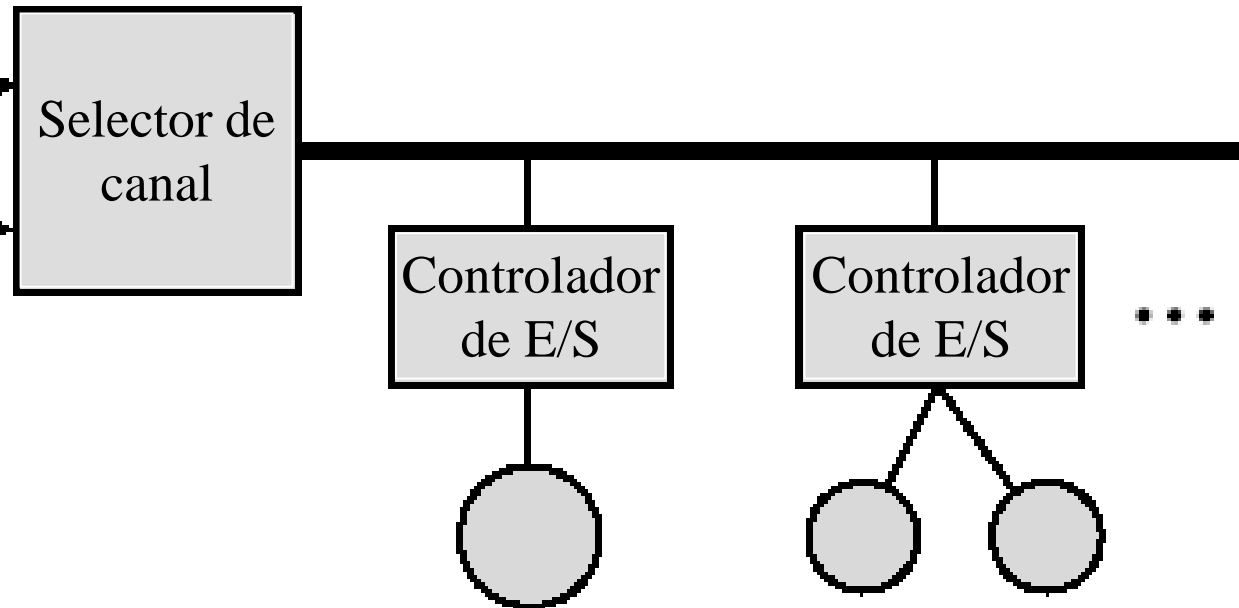
- Hay 2 tipos básicos de canales de E/S:
 - Selector
 - Multiplexor
- Canal selector de E/S
 - El canal controla varios dispositivos de alta velocidad, de a uno por vez.
 - El canal selecciona un dispositivo y efectúa la transferencia sobre el dispositivo seleccionado.
 - Cada dispositivo tiene asociado un controlador o módulo de E/S que lo maneja.
 - Por lo tanto el canal de E/S ocupa el lugar de la CPU en el control del módulo de E/S.

Canales de E/S

- En la imagen siguiente se puede apreciar el esquema de funcionamiento de un Canal selector de E/S.
- Solo puede atender 1 dispositivo a la vez.

Datos y direcciones
canal a mem.ppal.

Señales de control
CPU.- Selector



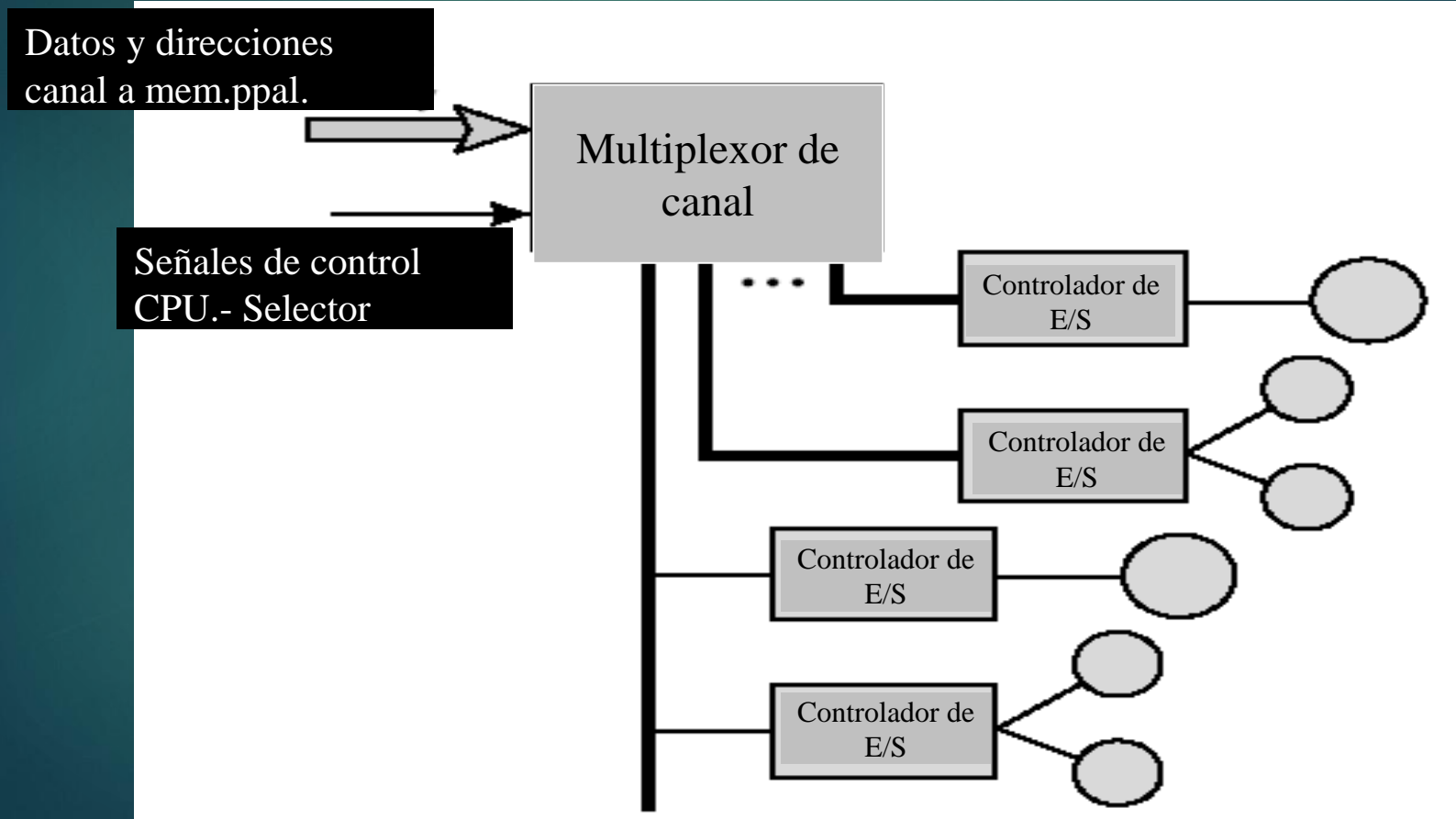
Canales de E/S

63

- Canal Multiplexor de E/S
 - El canal controla varios dispositivos de alta velocidad, incluso simultáneamente.
 - El canal multiplexa la atención entre los dispositivos seleccionados. El multiplexado puede ser:
 - Multiplexor de bytes: acepta y transmite de a caracteres.
 - Multiplexor de bloques: intercala bloques de datos desde distintos dispositivos.

Canales de E/S

- En la imagen siguiente se puede apreciar el esquema de funcionamiento de un Canal multiplexor de E/S.
- Puede atender varios dispositivos a la vez.



Referencias

- W. Stallings, 5º Ed - Capítulo 6.
- <http://www.pcguide.com/ref/mbsys/res/irq/func.htm>