

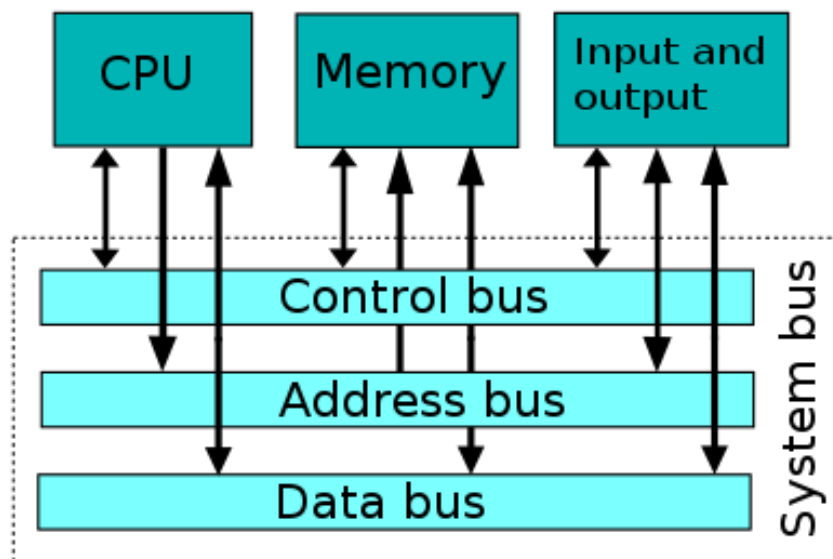


Arquitectura de Computadoras

Age

2024

Bus del sistema



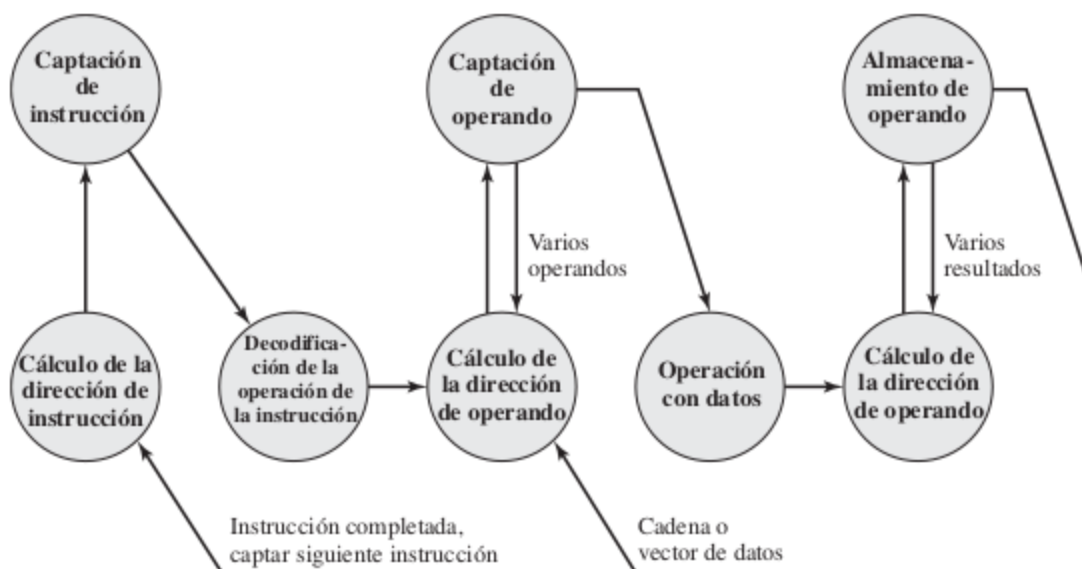
Bus de datos: Las líneas de datos proporcionan un camino para transmitir datos entre los módulos del sistema. El conjunto constituido por estas líneas se denomina bus de datos. El bus de datos puede incluir entre 32 y cientos de líneas, cuyo número se conoce como anchura del bus de datos. Puesto que cada línea solo puede transportar un bit cada vez, el número de líneas determina cuántos bits se pueden transferir al mismo tiempo. La anchura del bus es un factor clave a la hora de determinar las prestaciones del conjunto del sistema.

Bus de direcciones: Las líneas de dirección se utilizan para designar la fuente o el destino del dato situado en el bus de datos. Por ejemplo, si el procesador desea leer una palabra (8, 16 o 32 bits) de datos de la memoria, sitúa la dirección de la

palabra deseada en las líneas de direcciones. Claramente, la anchura del bus de direcciones determina la máxima capacidad de memoria posible en el sistema. Además, las líneas de direcciones generalmente se utilizan también para direccionar los puertos de E/S. Usualmente, los bits de orden más alto se utilizan para seleccionar una posición de memoria o un puerto de E/S dentro de un módulo.

Bus de control: Las líneas de control se utilizan para controlar el acceso y el uso de las líneas de datos y de direcciones. Puesto que las líneas de datos y de direcciones son compartidas por todos los componentes, debe existir una forma de controlar su uso. Las señales de control transmiten tanto órdenes como información de temporización entre los módulos del sistema. Las señales de temporización indican la validez de los datos y las direcciones.

Ciclo de instrucción basico (sin instrucciones)



1. **Cálculo de la dirección de la instrucción (IAC, Instruction Address Calculation):** determina la dirección de la siguiente instrucción a ejecutar. Normalmente, esto implica añadir un número fijo a la dirección de la instrucción previa. Por ejemplo, si las instrucciones tienen un

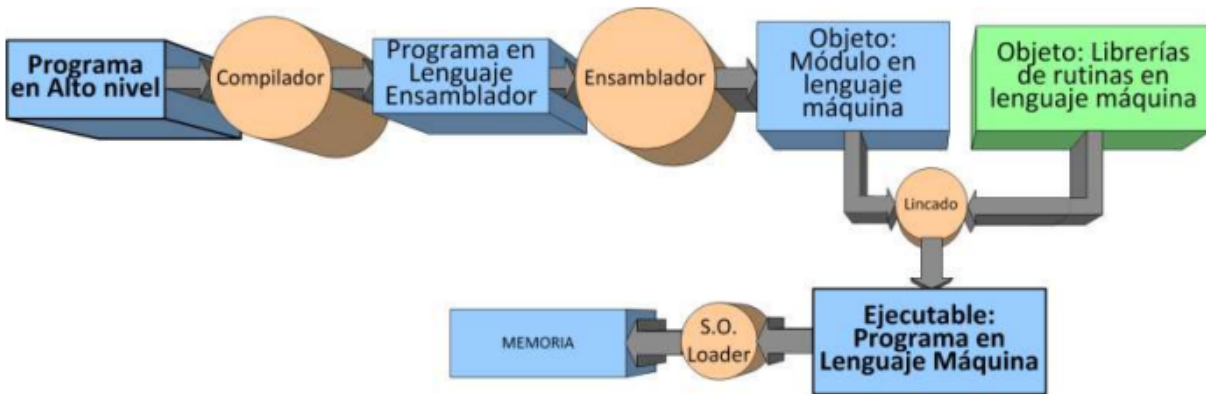
tamaño de 16 bits y la memoria se organiza en palabras de 16 bits, se suma 1 a la dirección previa. En cambio, si la memoria se organiza en bytes (8 bits) direccionables individualmente, entonces hay que sumar 2 a la dirección previa.

2. Captación de instrucción (if, Instruction Fetch): la CPU lee la instrucción desde su posición en memoria.
3. Decodificación de la operación indicada en la instrucción (IOD, Instruction Operation Decoding): analiza la instrucción para determinar el tipo de operación a realizar y el (los) operando(s) a utilizar.
4. Cálculo de la dirección del operando (OAC, Operand Address Calculation): si la instrucción implica una referencia a un operando en memoria o disponible mediante E/S, determina la dirección del operando.
5. Captación de operando (OF, Operand Fetch): capta el operando desde memoria o se lee desde el dispositivo de E/S.
6. Operación con los datos (DO, Data Operation): realiza la operación indicada en la instrucción.
7. Almacenamiento de operando (OS, Operand Store): escribe el resultado en memoria o lo saca a través de un dispositivo de E/S.

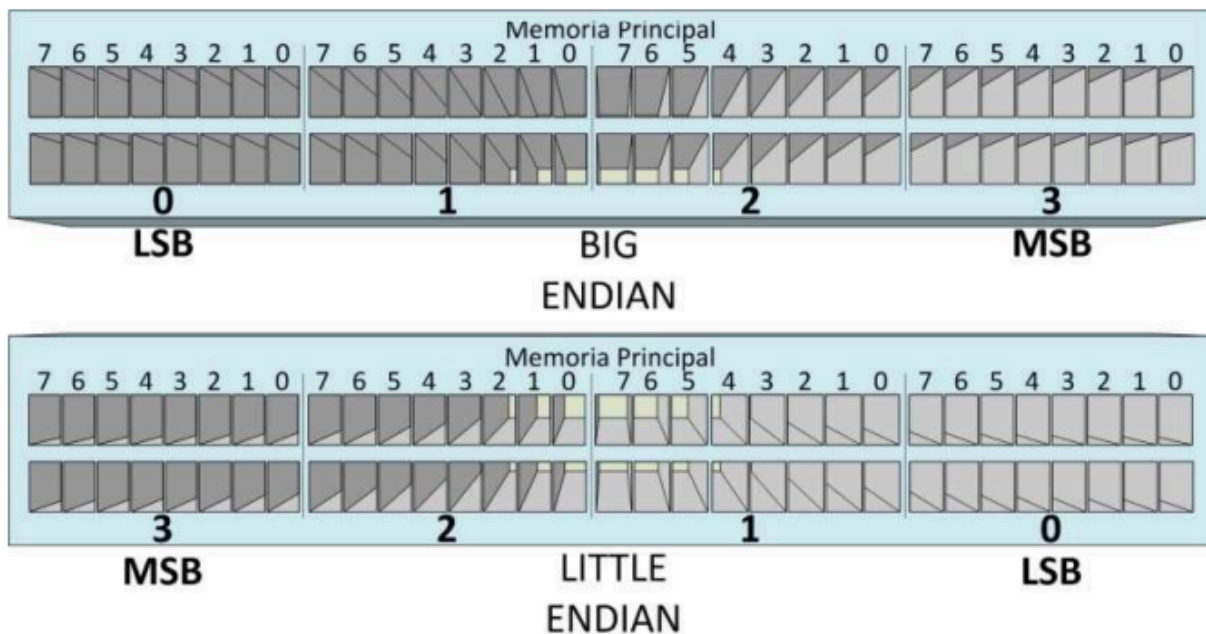
Diseño del procesador

Procesadores RISC: Reduced Instruction Set Computer. Solo se puede ejecutar una instrucción a la vez, repertorio con menos instrucciones. Mientras que los

CISC, tienen mas instrucciones, mas modos de direccionamiento y operaciones (mas caros).



Proceso de transformación de un lenguaje en alto nivel a lenguaje maquina-ensamblador



Modo de almacenamiento en memoria. Low significant bit - Most significant bit

Los accesos no alineados a memoria son muy lentos, ya que estos suponen varios accesos alineados a memoria.

En la CPU, hay registros que son propios del mismo procesador (no accesibles) y hay registros creados para usar nosotros (ax,bx, etc), que para usarlos voy a

necesitar hacer uso de las instrucciones, así mismo, lo que contengan los registros puede o no estar en memoria. El único registro que tenía Von Neumann era el acumulador.

Transferencia de datos (preg)

El código de operación, será el que dicte cuáles y cuántos operandos espera en la instrucción.

Tipos de direccionamiento:

- **Inmediato:** El operando se codifica dentro de la instrucción. La desventaja es que el tamaño del número está restringido a la longitud del campo de direcciones que, en la mayoría de los repertorios de instrucciones, es pequeño comparado con la longitud de palabra. Ej: `mov ax,10`
 - **Registro:** se incluye en la instrucción el identificador del registro del procesador en el que está almacenado el operando. Ej: `mov ax, bx`
 - **Directo o absoluto:** se incluye en la instrucción la dirección de memoria en la que está almacenado el operando. Ej: `mov ax, 40h`
 - **Indirecto:** se incluye en la instrucción el identificador del registro del procesador en el que está almacenada la dirección de memoria en la que se encuentra el operando. Ej: `mov ax, [bx]`
 - **Indirecto con desplazamiento:** caso particular del indirecto, en la que se desplaza el operando. Ej: `mov ax, [bx + 2]`
-

Funcionamiento de la pila

La pila es un sector de memoria con acceso LIFO (last in - first out), que permite dos operaciones con registros de 16bits.

Push:

- Se decrementa el SP en 1 (para salir del tope de pila 8000h)
- Se guarda la parte alta del registro en la posición del SP y repite para la parte baja.

Pop:

- Se guarda en la parte baja del registro lo que contiene la dirección SP
 - Se incrementa en 1 y repite para parte alta.
-

Saltos

Los saltos son instrucciones que permiten al programador cambiar el orden de ejecución del programa según sea necesario, dentro de ensamblador existen dos tipos de salto principales:

Saltos Incondicionales: Los saltos incondicionales se utilizan mediante la instrucción JMP, la cual transfiere el control a la línea especificada después de la palabra JMP, la cual puede ser un valor directo o una etiqueta.

También se puede contar como un salto incondicional la instrucción CALL, la cual llama un procedimiento y al terminarla devuelve el control a la línea siguiente de donde se inició la llamada a procedimiento.

Saltos Condicionales: Los saltos condicionales transfieren el control del programa a la ubicación que se les dé como parámetro si al hacer una comparación se cumple la condición establecida en el salto.

Interrupciones

Son un mecanismo mediante el cual se puede cambiar el procesamiento normal de la CPU.

- **Por programa:** Generadas por alguna condición que se produce como resultado de la ejecución de una instrucción, tal como desbordamiento aritmético (overflow), división por cero, intento de ejecutar una instrucción máquina inexistente e intento de acceder fuera del espacio de memoria permitido para el usuario.
- **Por temporizador:** Generadas por un temporizador interno al procesador. Esto permite al sistema operativo realizar ciertas funciones de manera regular.
- **Por E/S:** Generadas por un controlador de E/S, para indicar la finalización sin problemas de una operación o para avisar de ciertas condiciones de error.
- **Por fallo de hardware:** Generadas por un fallo tal como la falta de potencia de alimentación

o un error de paridad en la memoria.

Interrupciones por Hardware: Muchas interrupciones por hardware están relacionadas con los dispositivos de entrada/salida. Por ejemplo, cuando un dispositivo necesita atención del procesador, como la recepción de datos desde un teclado, ratón, tarjeta de red o disco duro, puede generar una interrupción para alertar al procesador de que hay datos listos para ser procesados. Estas interrupciones por hardware son esenciales para manejar la comunicación entre el procesador y los dispositivos periféricos de un sistema informático.

Sin embargo, las interrupciones por hardware no se limitan solo a dispositivos de entrada/salida. También pueden ser generadas por eventos internos del procesador, como errores de hardware, violaciones de memoria, o eventos relacionados con la gestión del sistema, como temporizadores de hardware.

Interrupciones por Software:

Las interrupciones por software, como las llamadas al sistema o las interrupciones de depuración, no están necesariamente ligadas a los módulos de entrada/salida. Por ejemplo, una llamada al sistema para solicitar la asignación de memoria o para crear un nuevo proceso no está directamente relacionada con dispositivos de E/S, pero aún así puede generar una interrupción por software para solicitar la atención del sistema operativo.

Manejo de Interrupciones:

Cuando ocurre una interrupción, el procesador detiene la ejecución del programa actual y salta a una ubicación predefinida en la tabla de vectores de interrupción, donde se encuentra el código para manejar esa interrupción específica. Este código de manejo de interrupciones, también llamado rutina de interrupción, guarda el estado actual del procesador, procesa la interrupción y luego restaura el estado del procesador para que el programa principal pueda continuar su ejecución normalmente.

Prioridades de Interrupción:

Algunos sistemas permiten asignar prioridades a las interrupciones, lo que significa que las interrupciones de mayor prioridad serán atendidas antes que las de menor prioridad. Esto es importante para garantizar que eventos críticos, como el manejo de errores, se resuelvan rápidamente.

Importante: los procesadores modernos repiten la instrucción interrumpida, ya que usualmente responden a dispositivos de E/S. El gestor solo volverá si ya cumplió con lo que la interrupción tendrá que hacer.

Manejo de interrupciones y prioridades:

La primera opción es desactivar todas las interrupciones mientras se está procesando una de ellas. Esto significa que cuando ocurre una interrupción, detenemos temporalmente la atención a otras interrupciones hasta que hayamos terminado de manejar la actual. Luego, reactivamos las interrupciones y comprobamos si hay otras pendientes. Es un enfoque simple pero puede ser lento para eventos importantes.

La segunda opción es asignar prioridades a las interrupciones. Esto significa que algunas interrupciones son más importantes que otras. Si llega una interrupción de alta prioridad mientras se maneja una de menor prioridad, el sistema atiende primero la de alta prioridad. Luego, vuelve a la interrupción original. Este método permite manejar eventos críticos más rápidamente, pero requiere un sistema más complejo. En resumen, la elección entre estos enfoques depende de qué tan críticas sean las interrupciones y cómo necesitamos manejarlas en el sistema.

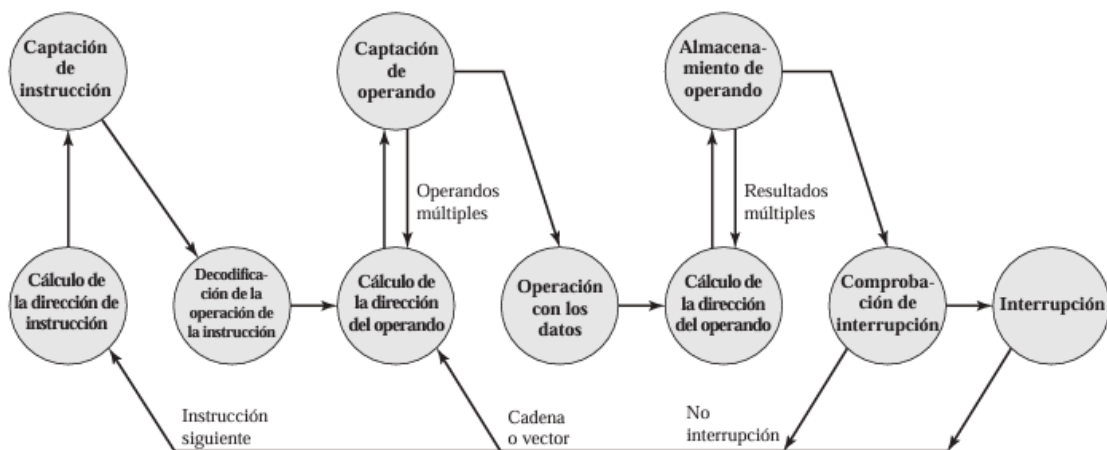


Figura 3.12. Diagrama de estados de un ciclo de instrucción, con interrupciones.

verdadero ciclo de instrucción

PIC

Registro de Máscara de Interrupción (IMR): Este registro se utiliza para habilitar o des-habilitar las interrupciones procedentes de diferentes fuentes. Cada bit en este registro representa una línea de interrupción y puede ser configurado para permitir o bloquear la interrupción correspondiente.

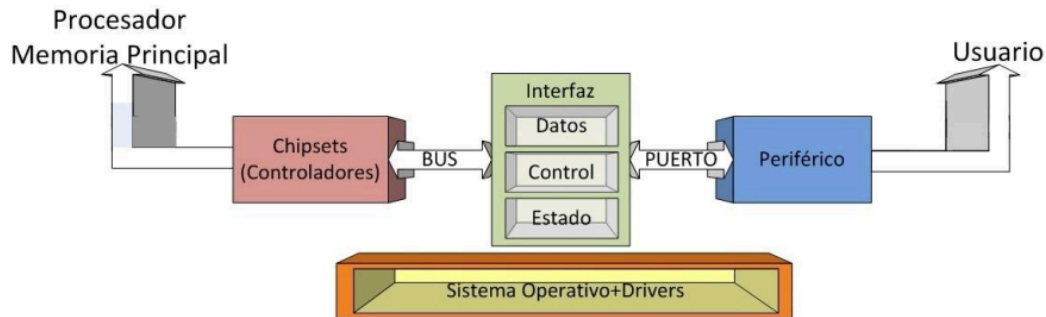
Registro de Estado (ISR): Proporciona información sobre el estado actual del PIC, como qué interrupciones están pendientes de ser atendidas y cuáles han sido atendidas.

Registro de Receptor de Interrupciones (IRR): Este registro mantiene un registro de todas las fuentes de interrupción que han solicitado atención al PIC, independientemente de si han sido atendidas o no. Cada bit en el IRR corresponde a una línea de interrupción y se establece cuando una fuente de interrupción solicita atención al PIC. El PIC consulta este registro para determinar qué fuente de interrupción debe ser atendida en caso de que haya varias solicitudes simultáneas.

Registros de Vectores de Interrupción: Estos registros contienen las direcciones de memoria de las rutinas de manejo de interrupciones para cada fuente de interrupción. Cuando ocurre una interrupción, el PIC consulta estos registros para determinar a qué dirección de memoria debe saltar para manejar la interrupción.

EOI	20H		Le avisa al PIC que la interrupción ya fue atendida
IMR	21H		Para habilitar o deshabilitar alguna interrupción
IRR	22H		Indica cuáles dispositivos externos solicitan interrumpir
ISR	23H		Indica cuál dispositivo externo está siendo atendido
INT 0	24H		Contiene <i>ID</i> asignado al F10
INT 1	25H		Contiene <i>ID</i> asignado al Timer
INT 2	26H		Contiene <i>ID</i> asignado al Handshake
INT 3	27H		Contiene <i>ID</i> asignado al CDMA

Entrada/Salida



Cada módulo se conecta al bus del sistema o a un conmutador central y controla uno o más dispositivos periféricos.

Las principales funciones y características de un modulo de entrada y salida son:

- Control y temporización.
- Comunicación con el procesador.
- Comunicación con los dispositivos.
- Almacenamiento temporal de datos.
- Detección de errores.

Estructura de un modulo E/S

El módulo se conecta al resto del computador a través de un conjunto de líneas (por ejemplo, líneas del bus del sistema). Los datos que se transfieren a, y desde, el módulo se almacenan temporalmente en uno o más registros de datos. Además, puede haber uno o más registros de estado que proporcionan información del estado presente. Un registro de estado también puede funcionar como un registro de control, para recibir información de control del procesador. La lógica que hay en el módulo interactúa con el procesador a través de una serie de líneas de control. Estas son las que utiliza el procesador para proporcionar las órdenes al módulo de E/S. Algunas de las líneas de control pueden ser utilizadas por el módulo de E/S (por ejemplo, para las señales de arbitraje y estado). El módulo también debe ser capaz de reconocer y generar las direcciones asociadas a los dispositivos que controla. Cada módulo de E/S tiene una dirección única o, si controla más de un dispositivo externo, un conjunto único de direcciones. Por

último, el módulo de E/S posee la lógica específica para la interfaz con cada uno de los dispositivos que controla.

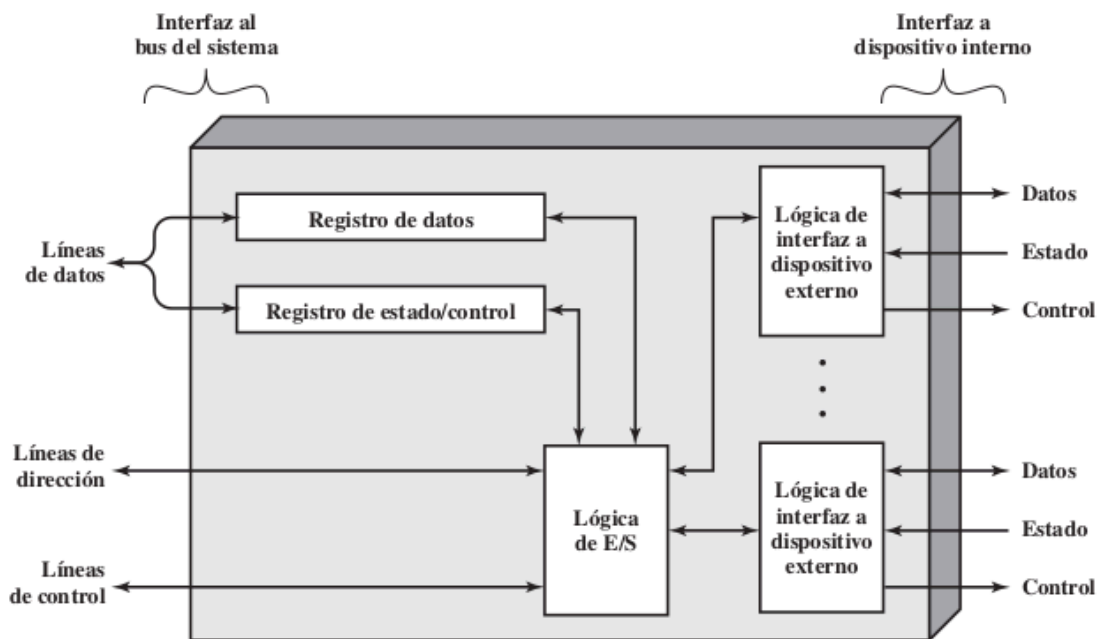
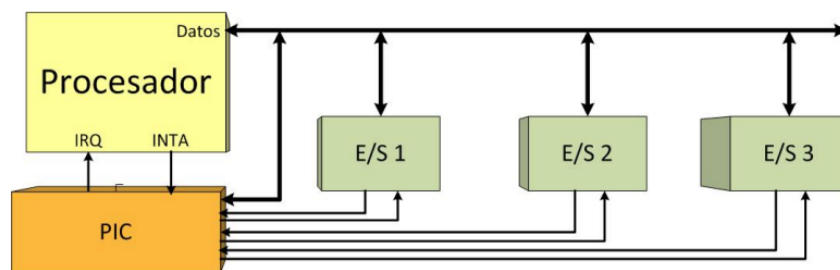


Figura 7.3. Diagrama de bloques de un módulo de E/S.

Un módulo de E/S que se encarga de la mayoría de los detalles del procesamiento, presentando al procesador una interfaz de alto nivel, se denomina generalmente canal de E/S o procesador de E/S. Un módulo que sea bastante simple y requiera un control detallado normalmente se denomina controlador de E/S o controlador de dispositivo. Los controladores de E/S usualmente aparecen en microcomputadores, mientras que los canales de E/S se utilizan en grandes computadores centrales (mainframes).



Interrupciones múltiples con PIC

Dispositivo tipo

La lógica de control asociada al dispositivo controla su operación en respuesta a las indicaciones del módulo de E/S. El transductor convierte las señales eléctricas asociadas al dato a otra forma de energía en el caso de una salida y viceversa en el caso de una entrada. Usualmente, existe un buffer asociado al transductor para almacenar temporalmente el dato que se está transfiriendo entre el módulo de E/S y el exterior; es común un tamaño de buffer de 8 a 16 bits.

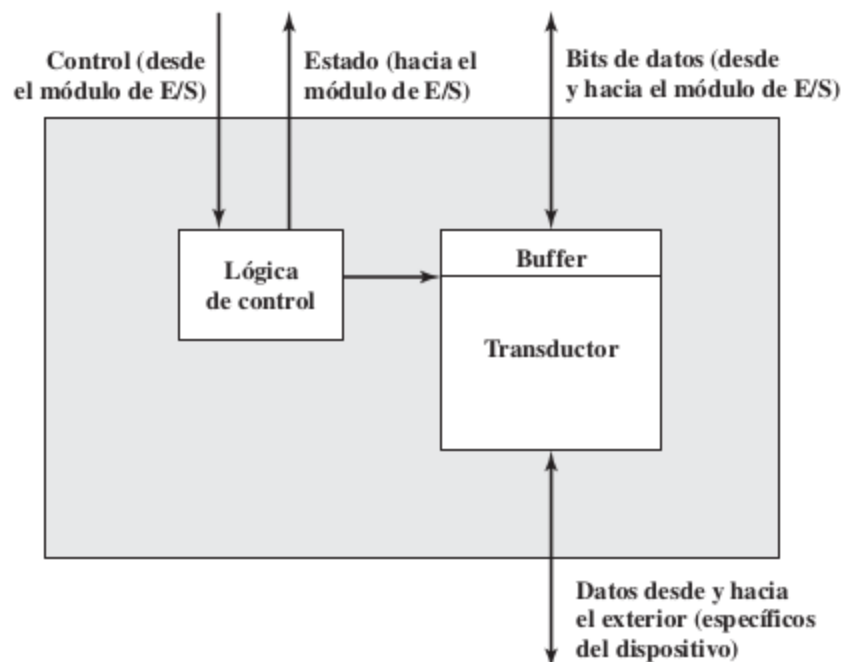


Figura 7.2. Diagrama de bloques de un dispositivo externo.

Operaciones de E/S

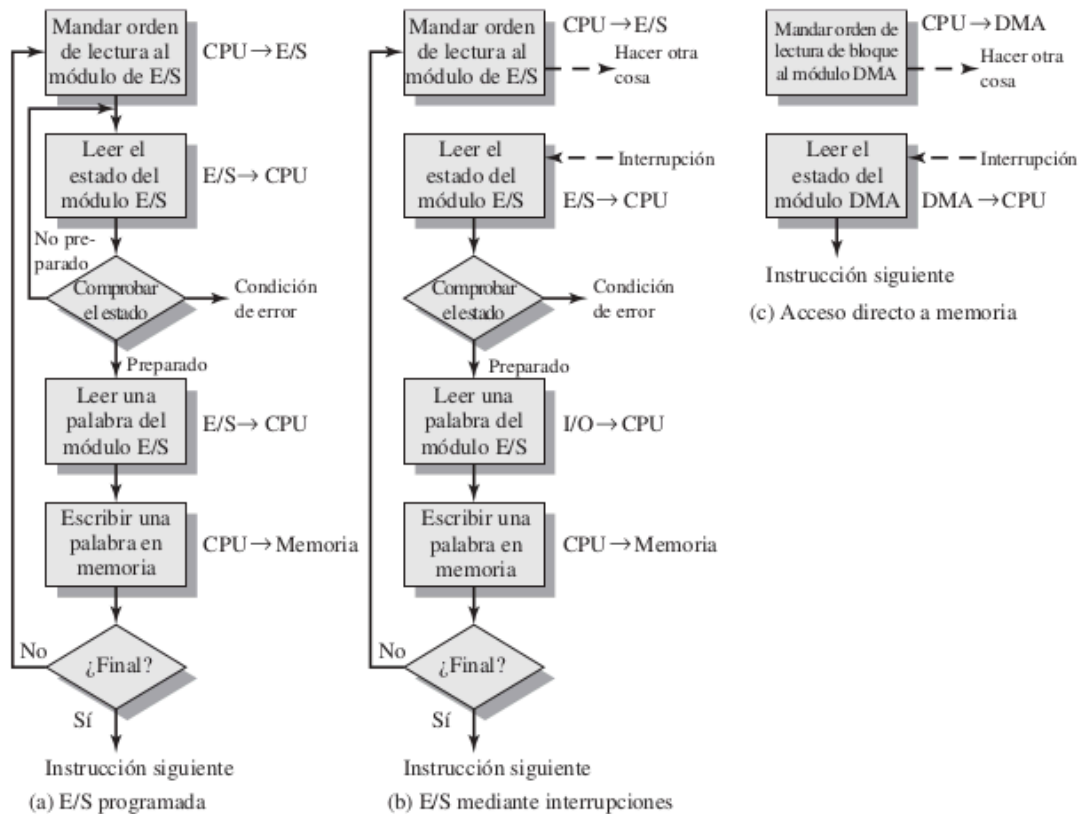


Figura 7.4. Tres técnicas para la entrada de un bloque de datos.

E/S programada

Cuando el procesador está ejecutando un programa y encuentra una instrucción relacionada con una E/S, ejecuta dicha instrucción mandando una orden al módulo de E/S apropiado. Con E/S programada, el módulo de E/S realizará la acción solicitada y después activará los bits apropiados en el registro de estado de E/S (Figura 7.3). El módulo de E/S no realiza ninguna otra acción para avisar al procesador. En concreto, no interrumpe al procesador. De esta forma, el procesador es responsable de comprobar periódicamente el estado del módulo de E/S hasta que encuentra que la operación ha terminado.

1. **E/S asignada en memoria:** En este modo, hay un único espacio de direcciones para las posiciones de memoria y los dispositivos de E/S. El procesador considera a los registros de estado y de datos de los módulos de E/S como posiciones de memoria y utiliza las mismas instrucciones máquina para acceder tanto a memoria como a los dispositivos de E/S. Esto permite una programación más eficiente ya que se puede utilizar un amplio repertorio de

instrucciones. Sin embargo, una desventaja es que se utiliza parte del valioso espacio de direcciones de memoria.

2. **E/S aislada:** En este modo, el espacio de direcciones de E/S está aislado del de memoria. Los puertos de E/S solo son accesibles mediante una orden específica de E/S, que activa las líneas de órdenes de E/S del bus. El rango completo de direcciones está disponible para ambos, memoria y E/S. Aunque solo existen unas pocas instrucciones de E/S, la ventaja es que no se utiliza el espacio de direcciones de memoria.

E/S con interrupciones

Es un método de E/S en el que un dispositivo envía una señal de interrupción al procesador cuando necesita su atención. El procesador interrumpe su tarea actual, atiende la solicitud del dispositivo de E/S y luego vuelve a su tarea original. Este método mejora la eficiencia del sistema al permitir que el procesador realice otras tareas mientras espera que se complete la operación de E/S.

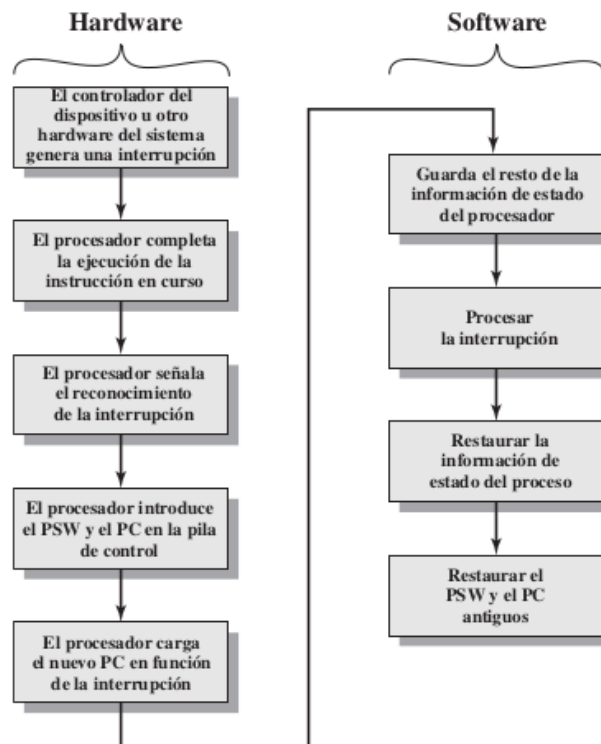


Figura 7.6. Procedimiento de interrupción simple.

1. **Consulta software:** Cuando el procesador detecta una interrupción, consulta a cada módulo de E/S para determinar cuál provocó la interrupción. Una vez identificado el módulo, el procesador ejecuta la rutina de servicio específica para ese dispositivo. La desventaja de esta técnica es el tiempo que consume.
2. **Conexión en cadena (Daisy Chain):** En esta técnica, todos los módulos de E/S comparten una línea común para solicitar interrupciones. Cuando el procesador recibe una interrupción, activa el reconocimiento de interrupción. Esta señal se propaga a través de la secuencia de módulos de E/S hasta que alcanza un módulo que solicitó interrupción. Este módulo responde colocando una palabra (vector) en las líneas de datos, que el procesador utiliza como un puntero a la rutina de servicio de dispositivo apropiada. Esta técnica se conoce como interrupción vectorizada.
3. **Arbitraje de bus:** Con esta técnica, un módulo de E/S debe primero obtener el control del bus antes de poder activar la línea de petición de interrupción. Cuando el procesador detecta la interrupción, responde mediante la línea de reconocimiento de interrupción. Luego, el módulo que solicitó la interrupción sitúa su vector en las líneas de datos.

Direct Memory Access (DMA)

Existe un compromiso entre estos dos inconvenientes. Con la E/S programada, el procesador se dedica completamente a la tarea de E/S, permitiendo una transferencia de datos a alta velocidad pero a costa de no poder realizar otras tareas. Por otro lado, la E/S con interrupciones permite al procesador realizar otras tareas, pero a expensas de reducir la velocidad de E/S.

Cuando se requiere transferir grandes volúmenes de datos, se utiliza una técnica más eficiente. Con DMA, el controlador de DMA puede acceder directamente a la memoria del sistema para transferir datos, liberando así al procesador de esta tarea. Esto permite transferencias de datos más rápidas y libera al procesador para realizar otras tareas. Sin embargo, la implementación de DMA puede ser más compleja y costosa.



Figura 7.11. Diagrama de bloques típico de un módulo de DMA.

DMA modo rafaga: el DMA solicita el bus a la cpu para transferir un bloque de datos y no lo libera hasta que este proceso finalice. La ventaja es que la transferencia se realiza rapido, pero por otra parte, se degrada mucho el rendimiento de la cpu y se tiene la tiene en espera mientras tanto.

Dma modo robo de ciclo: el DMA solicitara el bus y la cpu se lo consedera siempre y cuando no lo necesite, entonces el DMA hara transferencias de una palabra y repetira esto tantas veces sea necesario, no es una interrupcion. La ventaja es que no interferimos en el ciclo normal de la cpu, pero la transferencia tardara un poco mas.

A mas grande el volumen de la instruccion, usar interrupciones no me asegura que la cpu este menos tiempo ejecutandose, algo que si pasa cuando la transferencia es de volumenes chicos. (relacionar con ejemplo pagina 30-31). Claramente se ve que no depende de la velocidad reloj de la cpu, sino de la velocidad del dispositivo.

El DMA realiza un pedido de solicitud para usar la memoria y alguna e.s. Es una cuasi-interrupcion.

▼ Preguntas - Investigar

- Identificar líneas de una sola línea o buses.
- Funciones es lo que debe hacer el modulo, mientras que capacidad es lo que puede hacer.
- Hard poll (daisy chain)
- Int - IntA - vectores que piden int.
- Funcionamiento DMA . no me cierra

Memoria

Memoria cache: memoria de acceso por contenido (pregunta si tiene un elemento en vez de ir a una direccion). Su objetivo es tener velocidad, no mas almacenamiento.

A medida que nos vamos "alejando" de la cpu, hay mas almacenamiento pero el tiempo que tarda en acceder a los datos es mayor tambien.

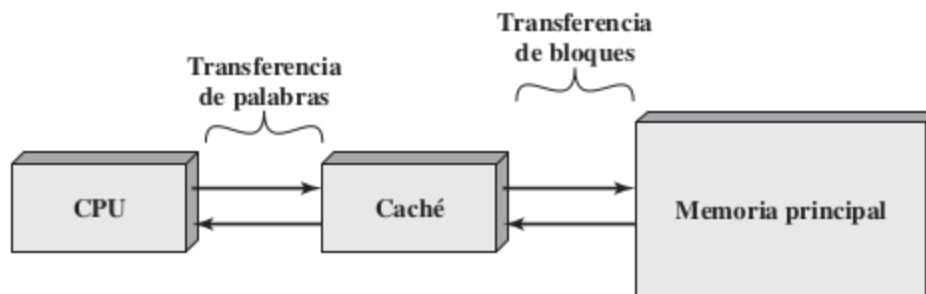


Figura 4.3. Memorias caché y principal.

principio basico y dibujo cache - memoria

Un bloque de memoria principal es una linea de memoria cache.

La cpu no pide el dato directamente a la cache, sino que primero le pregunta si lo tiene para ver si se lo da o lo va a buscar a memoria. Esto lo hace xq la cache esta "mas cerca" que la memoria.

El par de dos niveles de cache (baratas, del 10% del fallo), la penalización total se reduce al 1%.

diseño cache 137.

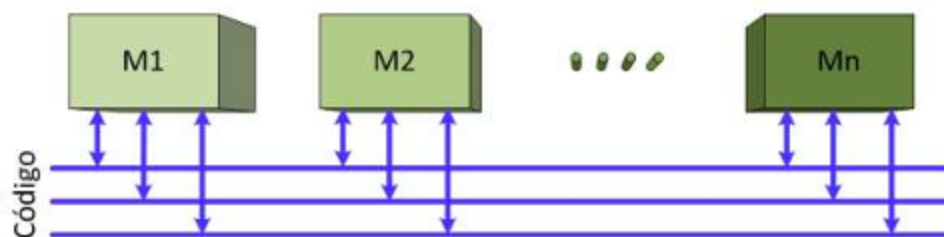
▼ Preguntas- investigar

- como es eso de que la cache puede no darte el dato que esta almacenado ahi?

Anexo clase 4

Descentralizado es un solo controlador, mientras que distribuido es una comunicacion de los modulos para ver quien lo utiliza.

Aribitrage distribuido: cada modulo puede responder al pedido, ya que todos estan conectados a la linea de pedidos. Aca surge los criterios del uso descentralizado, estableciendo prioridades. Este esquema permite conectar 2^3 modulos, es simple ya que todos pueden estar conectados al mismo tiempo y ademas si quiero agregar +8 modulos, solo basta con agregar una linea mas.



arbitraje distribuido

La temporización sincrona determina cuanto tiempo puede estar un modulo conectado al bus. Normalmente se sincroniza pasando de bajo a alto (flanco de subida). Depende la lectura o escritura es quien es el maestro o esclavo.

Bus PCI

Frases celebres



La unidad de control es casi DIOS



Maquina hay una sola, como la vieja



Mientras cacarea por acá, busca por otro lado, poliamor le dicen a eso



Todavía no llegamos a multiplicar los panes



El PIC es el secretario del medico, "el que atiende las interrupciones de la cpu"



Aguanta unos libros mas



CHAN



Los apel (refiriendose a Apple)



"Las mujeres caminan antes que los hombres, así como también hablan antes"



Floutin POINT

