

Conceptos básicos sobre el hardware

Los sistemas operativos utilizan los recursos hardware de un sistema de computación para ofrecer servicios a los usuarios. De acuerdo con esto, es importante tener ciertos conocimientos sobre el hardware de un sistema para comprender el funcionamiento de los sistemas operativos. El objetivo de este apéndice es ofrecer esos conocimientos a aquellos alumnos que no los posean, siendo conveniente hacer una lectura previa de éste antes de comenzar a estudiar la asignatura. Si el alumno ya conoce estos conceptos puede obviar su lectura.

A.1. Elementos básicos del computador

A alto nivel, un computador consta de procesador, memoria, y componentes de E/S, pudiendo tener uno o más módulos de cada tipo. Estos componentes se encuentran interconectados para lograr la función principal del

computador, que es la ejecución de programas. Por tanto, podemos distinguir cuatro elementos estructurales:

Procesador Controla la operación del computador y realiza sus funciones de procesamiento de datos.

Memoria principal Almacena programas y datos. Se suele conocer también como memoria real o primaria y es de tipo volátil.

Controladoras de E/S Permiten el intercambio de datos entre el computador y el exterior.

Sistema de interconexión Son aquellas estructuras y mecanismos que permiten la comunicación entre los procesadores, la memoria principal y las controladoras de E/S.

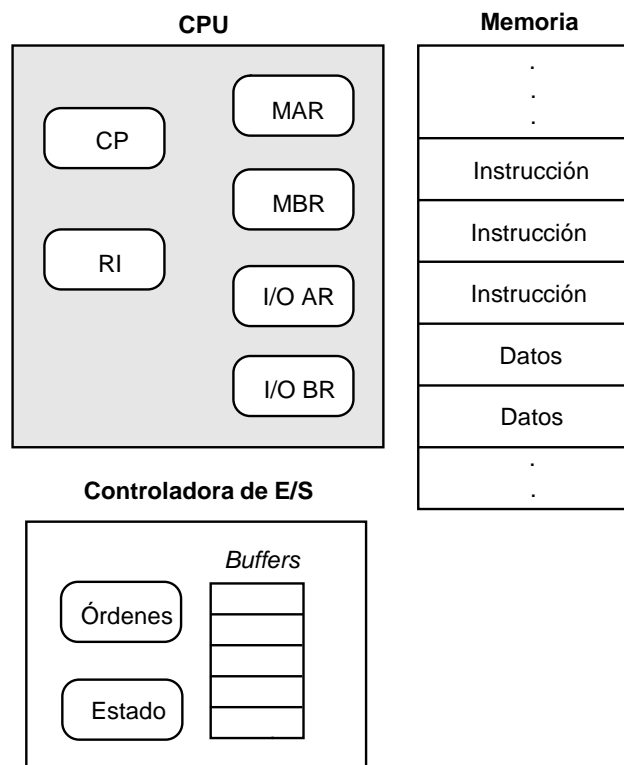


Figura A.1: Visión a alto nivel de un computador

La figura A.1 muestra estos componentes. El procesador realiza funciones de control y de intercambio de información. Para ello dispone de una serie de registros internos cuya función específica veremos más adelante. Un módulo de memoria consta de un conjunto de localizaciones definidas por unas direcciones numeradas secuencialmente. Cada localización contiene un número binario que puede ser interpretado como un dato o una instrucción. Una controladora de E/S transfiere datos desde los dispositivos externos al procesador y a la memoria, y viceversa. Contiene *buffers* que permiten mantener temporalmente los datos hasta que puedan ser enviados, así como unos registros de órdenes y de estado para dar las órdenes a los dispositivos, y conocer el estado de éstos.

A.2. Registros del procesador

El procesador dispone de un conjunto de registros que proporcionan un nivel de memoria más rápida y de menor capacidad que la principal. Podemos clasificarlos en dos tipos:

Registros visibles al usuario El programador puede acceder a ellos directamente utilizando el lenguaje máquina o ensamblador, con objeto de minimizar las referencias a memoria principal. Esto permite optimizar la velocidad de ejecución del programa. En el caso de los lenguajes de alto nivel, un compilador que optimice intentará hacer una elección inteligente a la hora de decidir qué variables asignar a registros y cuáles a localizaciones de memoria principal. Algunos lenguajes, tales como C, permiten al programador sugerirle al compilador qué variables deberían ser mantenidas en registros.

Registros de control y estado El procesador los utiliza para controlar su funcionamiento, y las rutinas privilegiadas del sistema operativo para controlar la ejecución de los programas.

A.2.1. Registros visibles al usuario

Un registro visible al usuario puede ser referenciado por medio del lenguaje máquina que ejecuta el procesador y normalmente está disponible para todos los programas, tanto los de aplicación como los del sistema. Los registros que entran normalmente dentro de esta categoría son los de datos, de dirección y de código de condición.

A los **registros de datos** les pueden ser asignadas varias funciones por parte del programador. En algunos casos son de propósito general y pueden ser usados con cualquier instrucción de la máquina que realice operaciones sobre los datos. Sin embargo, a menudo hay restricciones. Por ejemplo, puede haber registros dedicados para las operaciones de punto flotante.

Los **registros de dirección** contienen direcciones de memoria principal de datos o instrucciones. Pueden ser de propósito general o estar dedicados a un modo de direccionamiento particular.

Por último, un registro que es, al menos parcialmente, visible al usuario es el que mantiene los **códigos de condición** (o *flags*). Éstos son bits establecidos por el hardware del procesador como resultado de ciertas operaciones. Por ejemplo, una operación aritmética puede producir un resultado positivo, negativo, cero o un desbordamiento. Además de registrarse el resultado en un registro o en memoria, también se suele establecer el código de condición correspondiente.

Los bits de código de condición están reunidos en uno o más registros. Usualmente forman parte de un registro de control. Generalmente, las instrucciones de la máquina permiten que estos bits puedan ser leídos pero no modificados por el programador.

A.2.2. Registros de control y estado

Algunos registros del procesador se emplean para controlar su funcionamiento. En la mayoría de las máquinas, éstos no son visibles al usuario. En algunos casos se puede acceder a algunos de ellos mediante instrucciones máquina ejecutadas en un modo privilegiado.

Máquinas diferentes tendrán organizaciones de registros distintas y usarán diferente terminología. Aquí se proporciona una lista de los tipos de registros más importantes, junto con una breve descripción de éstos.

Una de las funciones del procesador es intercambiar datos con la memoria principal. Para este propósito, hace uso de dos registros internos:

Registro de dirección de memoria (MAR) Especifica la dirección de memoria donde se va a hacer la próxima operación de lectura o escritura.

Registro de *buffer* de memoria (MBR) Contiene los datos que se van a escribir en la memoria, o que se leen de ésta.

Para el intercambio de datos con las controladoras de E/S tenemos:

Registro de dirección de E/S (I/O AR) Especifica una controladora de E/S particular.

Registro de *buffer* de E/S (I/O BR) Se usa para intercambiar datos entre una controladora de E/S y el procesador.

Otros registros muy importantes son:

Contador de programa (CP) Contiene la dirección de la próxima instrucción a ser ejecutada.

Registro de instrucción (RI) Contiene la instrucción que se está ejecutando.

Todos los procesadores también incluyen un registro o conjunto de registros, conocido a menudo como palabra de estado del programa (PSW), que contiene información de estado y códigos de condición. Los campos e indicadores que suele incluir son:

Signo Contiene el bit de signo de la última operación aritmética realizada.

Cero Se establece cuando el resultado de una operación aritmética es cero.

Igual Se activa si una comparación lógica resulta en igualdad.

Desbordamiento Se usa para indicar desbordamiento aritmético.

Activar/Desactivar Interrupción Se utiliza para activar o desactivar las interrupciones.

Modo Indica el modo de ejecución del procesador, usuario o supervisor.

A.3. Ejecución de una instrucción

La función básica que realiza un computador es la ejecución de programas. Éstos constan de un conjunto de instrucciones y datos almacenados en memoria.

A continuación se va a analizar con detalle la ejecución de un programa. De una forma simple podemos considerar que el procesamiento de una instrucción, denominado **ciclo de instrucción**, consta de dos pasos (ver figura A.2):

Ciclo de lectura El procesador trae la instrucción de la memoria.

Ciclo de ejecución El procesador ejecuta esa instrucción.

La ejecución del programa consiste en la repetición de estos pasos tantas veces como sea necesario. Los dos pasos enumerados se pueden subdividir en varias acciones cada uno de ellos.

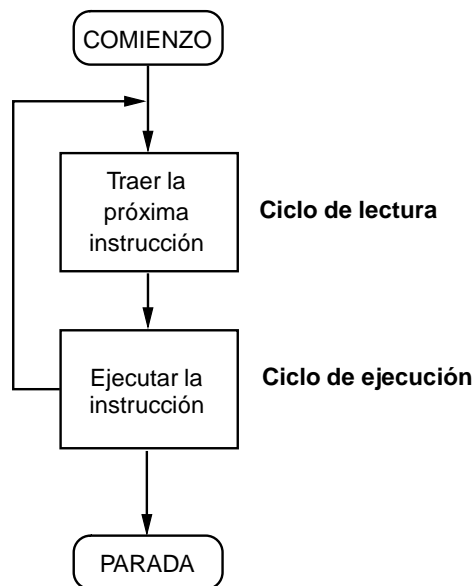


Figura A.2: Ciclo de instrucción básico

Al comienzo de cada ciclo, el procesador lee una instrucción de la memoria. ¿Cómo sabe éste qué instrucción es la que debe leer? Normalmente, el procesador cuenta con un registro que se denomina contador de programa que le indica la próxima instrucción a leer. A menos que se le diga otra cosa, cada vez que el procesador carga una nueva instrucción incrementa el valor de este registro.

La instrucción leída se carga en el registro de instrucción. La instrucción está en código binario y especifica las acciones a realizar por el procesador.

Éste interpretará la instrucción y ejecutará la acción requerida. En general, estas acciones se pueden clasificar en cuatro categorías:

Procesador-Memoria Los datos pueden ser transferidos desde el procesador a la memoria o desde la memoria al procesador.

Procesador-E/S Los datos pueden ser transferidos hacia o desde un periférico.

Procesamiento de datos El procesador puede realizar alguna operación aritmética o lógica sobre los datos.

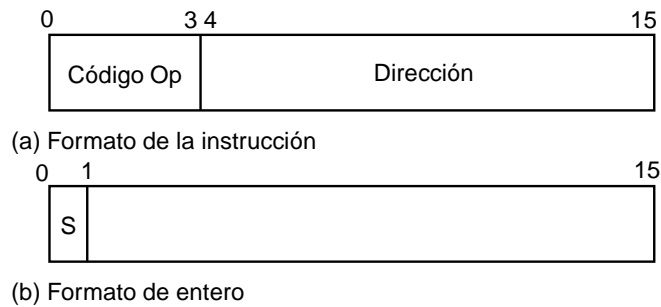
Control Una instrucción puede especificar que la secuencia de ejecución sea alterada. Por ejemplo, el procesador puede leer una instrucción de la localización 357, que especifique que la próxima instrucción será la que se encuentra en la localización 419. Para recordarlo, éste almacenará en CP la dirección 419. Así en el próximo ciclo de lectura, se tomará la instrucción de la localización 419 en vez de la 358.

La ejecución de una instrucción puede implicar una combinación de algunas de estas acciones. Consideremos un ejemplo simple usando una máquina hipotética que incluye las características listadas en la figura A.3.

El procesador contiene un registro de datos simple llamado acumulador (AC). Tanto las instrucciones como los datos son de 16 bits de longitud. El formato de la instrucción proporciona cuatro bits para el código de operación. Así podemos tener 2^4 códigos de operación diferentes, y 2^{12} palabras de memoria que pueden ser directamente direccionadas.

La figura A.4 ilustra la ejecución parcial de un programa, donde se muestran la memoria y los registros del procesador implicados. El programa añade el contenido de la palabra de memoria en la dirección 725 al contenido de la palabra de memoria en la dirección 726 y almacena el resultado en esta última localización. Para ello se requieren tres instrucciones:

1. El contador de programa contiene 120, la dirección de la primera instrucción. El contenido de la localización 120 se carga en el registro de instrucción.
2. Los cuatro primeros bits en el RI indican que se va a cargar el acumulador. Los restantes 12 bits especifican la dirección, que es 725.
3. Se incrementa el contador de programa, y se lee la próxima instrucción.



Contador de Programa (CP) = Dirección próxima instrucción

Registro de instrucción (RI) = Instrucción actual

Acumulador (AC) = Almacenamiento temporal

(c) Registros internos de la CPU

0001 = Cargar AC de la memoria

0010 = Almacenar AC en memoria

0101 = Añadir el contenido de la dirección de memoria a AC

(d) Lista parcial de códigos de operaciones

Figura A.3: Características de una máquina hipotética

4. Se suman el contenido antiguo de AC y el contenido de la localización 726; el resultado se almacena en AC.
5. Se incrementa nuevamente el contador de programa, y se lee la próxima instrucción.
6. El contenido de AC se almacena en la localización 726.

En este ejemplo se han necesitado tres ciclos de instrucción para añadir el contenido de la localización 725 al de la 726. Disponiendo de un conjunto de instrucciones más complejas se podrían necesitar menos ciclos. Los procesadores más modernos incluyen instrucciones que contienen más de una dirección. Así, el ciclo de ejecución para una instrucción particular puede implicar más de una referencia a memoria. También, en vez de referencias a memoria, una instrucción puede especificar una operación de E/S.

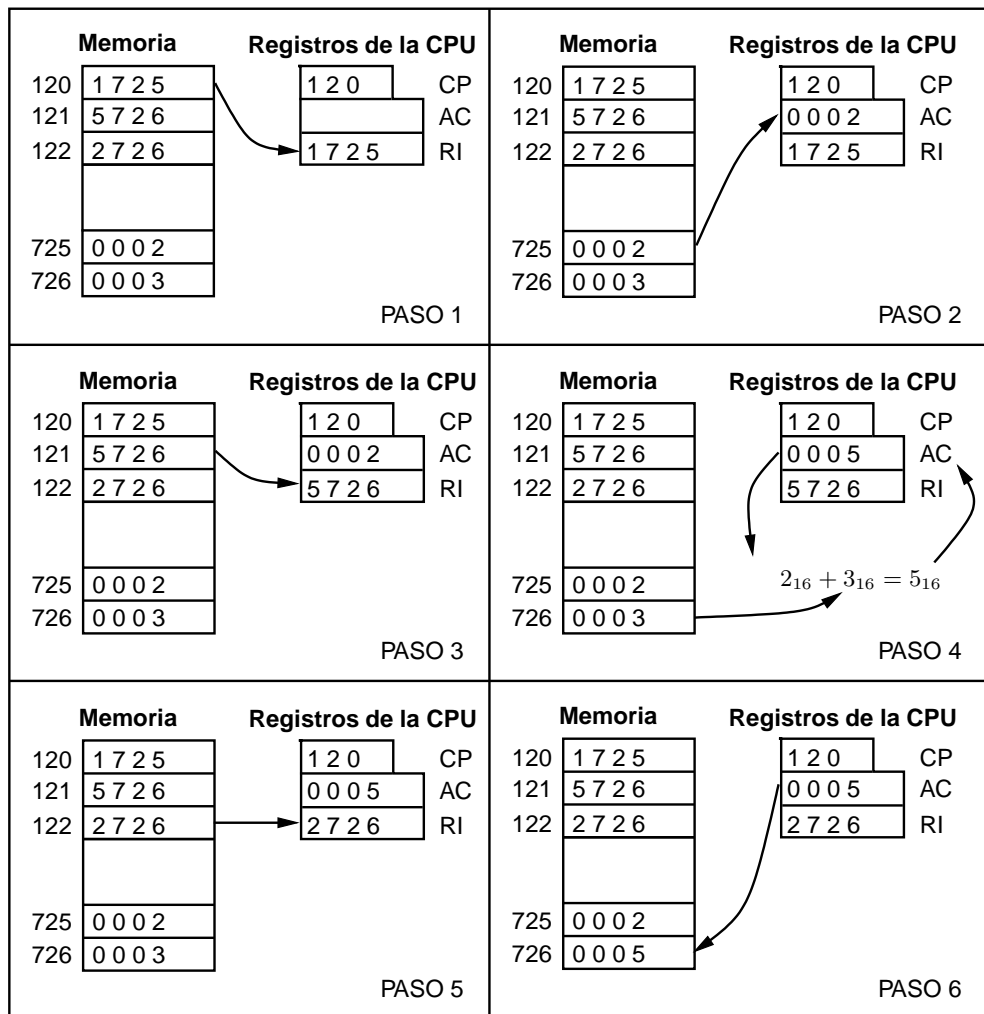


Figura A.4: Ejemplo de ejecución de un programa

A.4. Interrupciones

Todos los computadores proporcionan un mecanismo mediante el cuál otros módulos (dispositivos de E/S) pueden interrumpir el procesamiento normal del procesador.

Las interrupciones fueron proporcionadas inicialmente como una forma de mejorar la eficiencia del procesamiento. Sin la existencia de interrupciones, cada vez que se inicia una operación de E/S el procesador tendría que permanecer inactivo hasta que ésta terminara. Dado que los dispositivos de

E/S son bastante más lentos que la CPU, esto supondría un desaprovechamiento considerable del procesador. Con las interrupciones, el procesador puede ejecutar otras instrucciones mientras se está realizando una operación de E/S.

Desde el punto de vista del programa de usuario, una interrupción sólo supone interrumpir la secuencia normal de ejecución. Cuando termina el procesamiento de la interrupción, la ejecución continúa. Por tanto, los programas de usuario no necesitan contener un código especial para tratar las interrupciones; el procesador y el sistema operativo son los responsables de suspenderlo y reanudarlo posteriormente en el mismo punto.

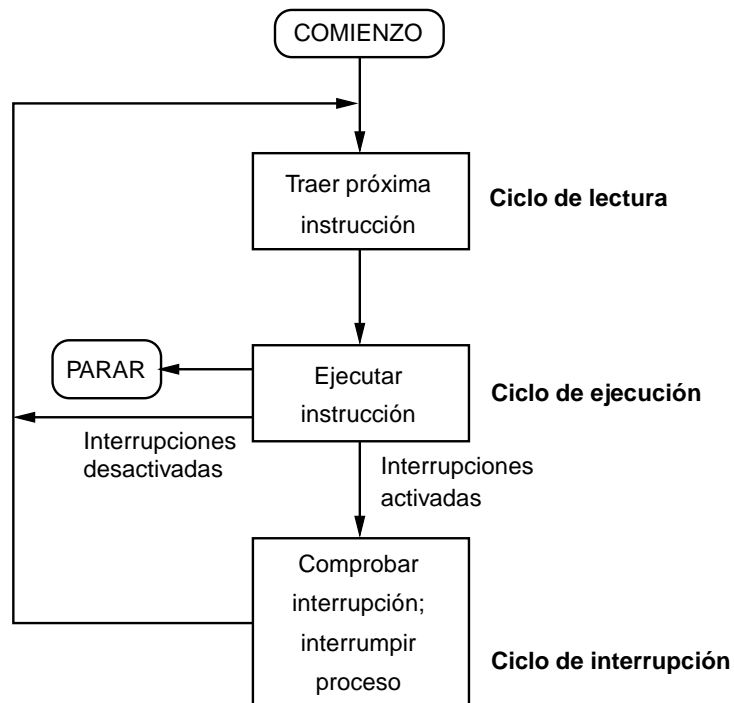


Figura A.5: Ciclo de instrucción con interrupciones

Para tratar las interrupciones se puede añadir un ciclo de interrupción al de instrucción, como se muestra en la figura A.5. En el ciclo de instrucción el procesador comprueba si ha ocurrido alguna interrupción; esto puede venir indicado por la presencia de alguna señal. Si no hay interrupciones pendientes, el procesador procede a la lectura de la próxima instrucción del programa actual. Si hay alguna interrupción pendiente, se suspende la ejecución del programa actual y se ejecuta la rutina manejadora de la interrupción.

Generalmente, ésta forma parte del sistema operativo. Esta rutina determina la naturaleza de la interrupción y realiza las acciones que se necesitan. Cuando finaliza su ejecución, el procesador puede reanudar la ejecución del programa de usuario en el punto de la interrupción.

Todo este proceso introduce una cierta carga adicional en el sistema. Deben ejecutarse instrucciones extras (el manejador de interrupciones) para determinar la naturaleza de la interrupción y decidir la acción apropiada. Sin embargo, dada la gran cantidad de tiempo que se podría gastar esperando en una operación de E/S, se obtiene un uso más eficiente del procesador con el uso de interrupciones.

A.4.1. Procesamiento de la interrupción

Una interrupción provoca una serie de eventos, que implican tanto al hardware como al software. La figura A.6 muestra una secuencia típica.

Cuando un dispositivo de E/S completa una operación, tienen lugar los siguientes eventos en los que está implicado el hardware:

1. La controladora emite una señal de interrupción al procesador.
2. El procesador finaliza la ejecución de la instrucción actual antes de responder a la interrupción.
3. El procesador comprueba si ha habido una interrupción, determina que se ha producido una, y envía una señal de reconocimiento a la controladora.
4. El procesador se prepara para transferir el control a la rutina de la interrupción. Para empezar, necesita guardar la información necesaria para volver a reanudar el programa actual en el punto de la interrupción. La información mínima requerida es la palabra de estado del programa (PSW) y la localización de la siguiente instrucción a ser ejecutada, que está contenida en el contador de programa. Es lo que se denomina el cambio de contexto de un proceso.
5. El procesador carga ahora en el contador de programa la localización de la rutina manejadora de la interrupción.

Una vez que el contador de programa ha sido cargado, el procesador procede al próximo ciclo de instrucción, que empieza con la lectura de una

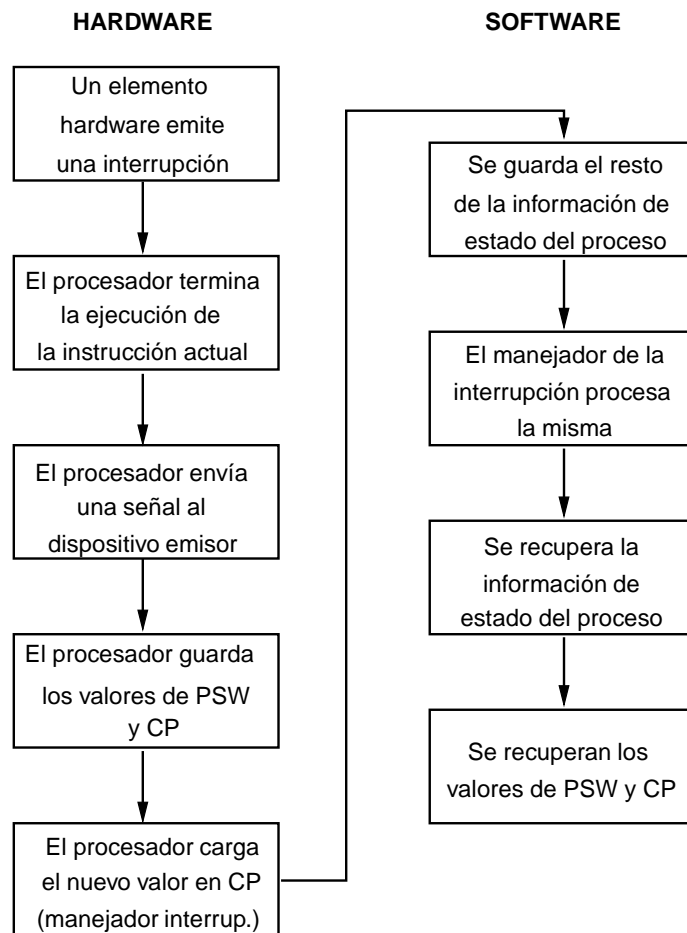


Figura A.6: Procesamiento de una interrupción

instrucción del manejador de la interrupción. La ejecución de éste conlleva las siguientes operaciones:

1. Guarda el resto de la información del proceso anterior que es necesaria para su posterior reanudación. En este caso, un programa de usuario es interrumpido después de la instrucción localizada en N . El contenido de todos los registros más la dirección de la próxima instrucción, $N + 1$, son colocados en la pila del núcleo. El puntero a la pila es actualizado y el contador de programa apunta ahora al principio de la rutina de servicio de la interrupción.
2. El manejador de la interrupción procederá ahora a procesar la misma.

Esto puede incluir un examen de la información de estado correspondiente a la operación de E/S o de otro evento que haya causado la interrupción y puede implicar el envío de órdenes adicionales a la controladora de E/S.

3. Cuando el procesamiento de la interrupción ha finalizado, los valores de los registros que antes fueron guardados son recuperados de la pila del núcleo y grabados de nuevo en los registros.
4. La acción final es recuperar los valores de PSW y del contador de programa de la pila. Como resultado de esto la próxima instrucción a ser ejecutada será la del programa previamente interrumpido.

A.5. Buses

Un computador consta de un conjunto de componentes (procesador, memoria, controladoras de E/S, etc.) que se comunican entre sí. En la figura A.7 se representan los tipos de intercambios que se necesitan indicando las formas principales de entrada y salida para cada tipo de módulo:

Memoria Un módulo de memoria suele constar de N palabras de igual longitud. A cada palabra se le asigna una dirección numérica única. Las operaciones que se pueden realizar son la lectura o escritura de palabras de datos. La localización de la operación se especifica mediante una dirección.

Controladora de E/S Desde un punto de vista interno (al sistema de computación), la E/S es similar funcionalmente a la memoria. Hay también dos operaciones, lectura y escritura. Una controladora de E/S puede controlar más de un dispositivo externo, para referirnos a cada uno de ellos se le debe dar una dirección única. Además, hay caminos para los datos internos y externos, tanto para la entrada como para la salida de datos. Por último, una controladora de E/S debe ser capaz de mandar señales de interrupción al procesador.

Procesador El procesador lee instrucciones y datos, y posteriormente escribe los datos obtenidos como resultado de su procesamiento. También utiliza señales de control para controlar la operación global del sistema y recibe señales de interrupción.

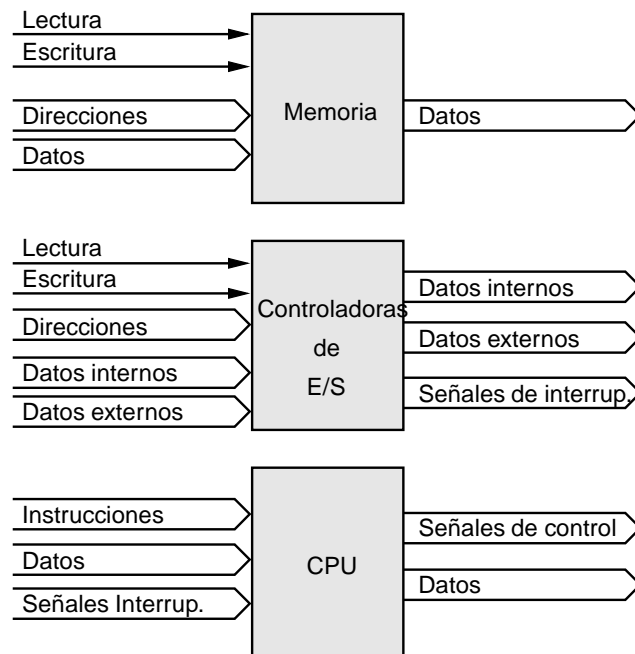


Figura A.7: Módulos del computador

Además de los datos que se necesitan intercambiar, se necesita también una estructura de interconexión que permita que cada componente se comunique directamente con todos los demás. Para ello se suelen utilizar los **buses**.

Un bus es un medio de transmisión compartido que conecta dos o más módulos, que consta de varias líneas de comunicación. Cada línea es capaz de transmitir señales que representan un 1 ó un 0. Dado que el bus es compartido, una señal transmitida por cualquier módulo o en cualquier línea está disponible para su recepción por cualquiera de los módulos unidos a esa línea. Por tanto, habrá que establecer algunas normas para que no haya problemas.

Un bus suele constar de 50 a 100 líneas separadas. A cada línea se le suele asignar un significado o función particular. Aunque hay muchos diseños diferentes de buses, en cualquiera de ellos las líneas se suelen clasificar en tres grupos funcionales, líneas de datos, direcciones y control. Además puede haber líneas de distribución de potencia que suministren corriente a los módulos unidos a ellas.

Las líneas de datos proporcionan una vía para el movimiento de datos entre los distintos módulos del sistema. Estas líneas son llamadas de forma

conjunta bus de datos. Los buses de datos suelen constar de 8, 16, o 32 líneas; el número de líneas se suele referenciar como la anchura del bus de datos. Dado que cada línea sólo puede llevar 1 bit a un tiempo, el número de líneas determina cuantos bits pueden ser transferidos a un tiempo. La anchura del bus de datos es un factor clave al determinar el rendimiento global del sistema. Por ejemplo, si el bus de datos es de 8 bits y cada instrucción tiene 16 bits, el procesador debe acceder al módulo de memoria dos veces durante cada ciclo de lectura de la instrucción.

Las líneas de direcciones son utilizadas para designar la fuente o destino de los datos en el bus de datos. Por ejemplo, si el procesador ejecuta una instrucción que referencia una palabra (8, 16 o 32 bits) de datos que tiene que ser leída de la memoria, el procesador pone la dirección de la palabra deseada en la líneas de dirección. La anchura del bus de dirección determina la capacidad de memoria máxima posible que puede tener el sistema. Las líneas de dirección también se suelen usar para direccionar los puertos de E/S.

Las líneas de control son usadas para controlar el acceso y el uso de las líneas de datos y de direcciones. Dado que éstas son compartidas por todos los componentes debe haber un medio de controlar su uso.

Un bus opera de la siguiente forma. Si un módulo quiere enviar datos a otro, debe hacer dos cosas, primero obtener el uso del bus, y posteriormente transferir los datos a través del bus. Si un módulo desea recibir datos de otro, debe obtener el uso del bus, y después transferir una petición al otro módulo a través de las líneas de control y de dirección apropiadas. Posteriormente, debe esperar a que el segundo módulo le envíe los datos.