**¿Qué es un computador?**

Es una máquina electrónica digital programable que procesa unos datos de entrada, obteniendo unos resultados que después devuelve como salida. Generalmente está formado por dos partes: el hardware (parte física) y el software (parte intangible).

**Componentes:**

**Procesador:**

● Es el encargado de controlar el funcionamiento del computador.

● Realiza las funciones de procesamiento de datos.

● Si sólo hay un procesador, se le denomina Unidad Central de Proceso (CPU - Central Processing Unit).

● Un mismo computador puede tener más de un procesador.

● Otra de sus funciones es el intercambio de datos con la memoria → SWAPPING Para ello se suelen utilizar dos registros internos del procesador:

● Un registro de dirección de memoria (RDIM), que especifica la dirección de memoria de la siguiente lectura o escritura.

● Un registro de datos de memoria (RDAM), que contiene los datos que se van a escribir en la memoria o que recibe los datos leídos de memoria.

● De forma análoga existen registros para operaciones similares con la E/S (RDIE/S y RDAE/S).

**Memoria principal**

● Es la encargada de almacenar datos y programas en ejecución.

● Habitualmente es una memoria volátil: cuando se apaga el computador, pierde su contenido (en contraste, la memoria secundaria o memoria del disco sí mantiene su contenido cuando el computador se apaga).

● A la memoria principal también se la puede llamar memoria real o memoria primaria.

RAM = MEMORIA DE ACCESO ALEATORIO

* “Acceso Aleatorio”: ● Se puede leer o escribir en una posición de memoria con un tiempo de espera igual para cualquier posición. ● No es necesario seguir ningún orden para acceder (acceso secuencial) a la información.
* En la RAM siempre están: 1. Todas las instrucciones que ejecuta la CPU (se cargan previamente antes de la ejecución). 2. Los datos que están manipulando los programas en ejecución.

Un módulo de memoria está formado por un conjunto de posiciones definidas mediante direcciones numeradas secuencialmente. ● En cada posición hay en patrón de bits que se puede interpretar como una instrucción o como datos

32 bits significa 2^(32)

**Módulos de E/S:**

● Son los encargados de transferir los datos entre el computador y el entorno externo.

● El entorno externo está formado por diversos dispositivos, lo que incluye a los dispositivos de memoria secundaria (por ejemplo, discos), equipos de comunicaciones, terminales, etc.

● Los módulos de E/S transfieren datos desde los dispositivos externos hacia el procesador y la memoria. Contienen buffers (zonas de almacenamiento internas) que mantienen temporalmente los datos hasta que estos se puedan enviar.

**Bus del sistema:**

● Es el encargado de proporcionar la comunicación entre los procesadores, la memoria principal y los distintos módulos de E/S

**Los registros del procesador:**

Los procesadores incluyen un conjunto de registros que proporcionan un tipo de memoria más rápida y de menor capacidad que la memoria principal. Registros visibles para el usuario

● Permiten al programador minimizar las referencias a memoria principal.

● En los lenguajes de alto nivel, un compilador que realice optimización intentará tomar decisiones inteligentes sobre qué variables se asignan a registros y cuáles a direcciones de memoria principal.

● Algunos lenguajes de programación de bajo nivel (como C) permiten que el programador sugiera al compilador qué variables debería almacenarse en los registros.

Los procesadores incluyen un conjunto de registros que proporcionan un tipo de memoria más rápida y de menor capacidad que la memoria principal. Registros de control y estado

● Usados por el procesador para controlar su operación.

● También usados por rutinas privilegiadas del sistema operativo para controlar la ejecución de programas

- REGISTROS VISIBLES PARA EL USUARIO:

Registros de dirección ● Contienen direcciones de memoria utilizadas para el cálculo de direcciones efectivas. A un registro visible para el usuario se puede acceder mediante código máquina y están disponibles para todos los programas (de aplicación y de sistema).

Registros de datos ● Suelen ser de propósito general. ● Pueden ser usados por cualquier instrucción que realice operaciones con datos. ● Frecuentemente tienen restricciones: registros dedicados a operaciones en coma flotante, etc

- REGISTROS DE CONTROL Y ESTADO Registro de instrucción (Instruction Register, IR) ● Contiene la última instrucción leída. Destinados a controlar el funcionamiento del procesador. Por lo general, no son visibles al usuario. Dentro de este grupo están el RDIM, RDAM, RDIE/S y RDAE/S.

Contador de Programa (Program Counter, PC) ● Contiene la dirección de la próxima instrucción que se leerá de la memoria.

**Ejecución de instrucciones**:

¿Y qué es una instrucción? Pues igual que todo en un ordenador, es un conjunto de 0s y 1s

Simplificando un poco, el procesamiento de una instrucción consta de dos partes: 1. El procesador lee (busca) instrucciones en memoria (una cada vez) y…

2. Ejecuta cada una de ellas. → La ejecución de una instrucción puede involucrar una o varias acciones. La ejecución de un programa consiste en repetir estos dos procesos:

1. Al principio de cada ciclo de instrucción, el procesador lee una instrucción de memoria. En un procesador típico, el contador de programa (PC) almacena la dirección de la siguiente instrucción que se va a leer.

2. A menos que se le indique otra cosa, el procesador siempre incrementa el PC después de cada instrucción ejecutada, de manera que se leerá la siguiente instrucción en orden secuencial (es decir, la instrucción situada en la siguiente dirección de memoria más alta).

3. La instrucción leída se carga dentro del registro de instrucción (IR).

4. La instrucción contiene una serie de bits que especifican la acción que debe realizar el procesador: código de operación. 5. El procesador interpreta la instrucción y lleva a cabo la acción requerida.

Generalmente estas acciones pertenecen a una de estas cuatro categorías (una ejecución de una instrucción puede involucrar una combinación de estas acciones).

1. PROCESADOR - MEMORIA ● Se pueden transferir datos desde el procesador a la memoria o viceversa.
2. 2. PROCESADOR - E/S ● Se pueden enviar datos a un dispositivo periférico o recibirlos desde el mismo, transfiriéndolos entre el procesador y un módulo de E/S..
3. CONTROL ● Una instrucción puede especificar que se va a alterar la secuencia de ejecución. ● Por ejemplo, el procesador puede leer una instrucción de la posición de memoria 50, que especifica que la siguiente instrucción estará en la posición 180. El procesador almacenará en el contador del programa el valor de 180. Como consecuencia, en la siguiente fase de búsqueda, se leerá la instrucción de la posición 180 en vez de la 51.
4. 4. PROCESAMIENTO DE DATOS ● El procesador puede realizar algunas operaciones aritméticas o lógicas sobre los datos

Tabla

Descripción generada automáticamente

**SISTEMA DE E/S**

● Se pueden intercambiar datos directamente entre un módulo de E/S y el procesador. ● Al igual que el procesador puede iniciar una lectura o escritura en memoria, especificando la dirección de la posición de memoria, también puede leer o escribir datos en un módulo de E/S. ● Estos intercambios entre procesador y E/S son interesantes para aliviar la carga de la memoria principal.

**Interrupciones**

Prácticamente todos los computadores proporcionan un mecanismo por el cual otros módulos (memoria y E/S) pueden interrumpir el secuenciamiento normal del procesador. ¿Esto para qué sirve? Las interrupciones constituyen una manera de mejorar la utilización del procesador.

**Ejemplo**: Supongamos que el procesador está transfiriendo datos a una impresora. Después de cada instrucción de escritura, el procesador debe parar y permanecer inactivo hasta que la impresora la lleve a cabo. La longitud de esta pausa puede ser del orden de muchos miles o incluso millones de ciclos de instrucción → ENORME DESPERDICIO DE LA CAPACIDAD DEL PROCESADOR

La mayoría de los dispositivos de E/S son mucho más lentos que el procesador.

Imaginemos que un computador personal opera a una frecuencia de 1 GHz. Ejecuta 10⁹ instrucciones por segundo → 1,000,000,000 instrucciones por segundo. Un disco duro típico tiene una velocidad de rotación de 7200 rpm, que corresponde con un tiempo de rotación de media pista de 4 ms. ES 4 MILLONES DE VECES MÁS LENTO QUE EL PROCESADOR

Gracias a las interrupciones, el procesador puede dedicarse a ejecutar otras instrucciones mientras que la operación de E/S se está llevando a cabo.

A través de una interrupción se puede echar a un proceso del procesador para ejecutar otro que tenga una prioridad más alta:

Los tipos de interrupciones más habituales son:

-Interrupciones de programa: Generada por alguna condición que se produce como resultado de la ejecución de una instrucción, tales como un desbordamiento aritmético, una división por cero, un intento de ejecutar una instrucción de máquina ilegal, y las referencias fuera del espacio de la memoria permitido para un usuario

-Interrupciones por temporizador: Generada por un temporizador del procesador. Permite al sistema operativo realizar ciertas funciones de forma regular.

-Interrupciones de E/S: Generada por un controlador de E/S para señalar la conclusión normal de una operación o para indicar diversas condiciones de error

- Interrupciones por fallo del hardware: Generada por un fallo, como un fallo en el suministro de energía o un error de paridad en la memoria del sistema

Desde el punto de vista del programa de usuario, una interrupción suspende la secuencia normal de ejecución. ● Cuando se completa el procesamiento de la interrupción, se reanuda la ejecución. ● El programa de usuario no tiene que contener ningún código especial para tratar las interrupciones: el procesador y el sistema operativo son responsables de suspender el programa de usuario y, posteriormente, reanudarlo en el mismo punto en el que estaba

Para poder tratar las interrupciones, se añade una fase de interrupción al ciclo de la instrucción. ● En esta fase, el procesador comprueba si se ha producido alguna interrupción, hecho indicado por la presencia de una señal de interrupción. ● Si no hay interrupciones pendientes, el procesador continúa con la fase de búsqueda y lee la siguiente instrucción del programa actual. ● Si está pendiente una interrupción, el procesador suspende la ejecución del programa actual y ejecuta la rutina del manejador de interrupción (rutina que generalmente es parte del sistema operativo). ● Esta rutina determina la naturaleza de la interrupción (¿qué es lo que ha pasado?) y realiza las acciones que sean necesarias.

**Diagrama

Descripción generada automáticamente**

● Cuando se completa la ejecución del manejador de interrupciones, el procesador puede reanudar la ejecución del programa de usuario en el punto de la interrupción. Evidentemente, este proceso implica SOBRECARGA.

● Deben ejecutarse instrucciones adicionales (en el manejador de interrupción) para determinar la naturaleza de la interrupción y decidir sobre la acción apropiada.

● Sin embargo, debido a la gran cantidad de tiempo que se gastaría simplemente esperando por una operación de E/S, es más eficiente utilizar el procesador para gestionar interrupciones.

**Interfaz de usuario gráfica

Descripción generada automáticamente**

Pero… ¿qué pasa cuando llegan varias interrupciones a la vez?

OPCIÓN 1. INHABILITAR LAS INTERRUPCIONES MIENTRAS SE ESTÁ PROCESANDO UNA INTERRUPCIÓN ● Si llega una nueva, se quedará a la espera de que termine la anterior. ● El problema es que aquí no tenemos en cuenta la importancia relativa de las instrucciones (algunas son más prioritarias que otras).

OPCIÓN 2. DEFINIR PRIORIDADES PARA LAS INTERRUPCIONES ● Si llega una de mayor prioridad, se interrumpe el procesamiento de la anterior.

**Jerarquía de memoria**

3 restricciones de diseño claves en la memoria de un computador:

1. ¿Cuál es su capacidad? Esta en el fondo no es muy importante, podemos hacer memorias gigantes y la gente seguirá desarrollando aplicaciones que intenten usarla toda.
2. ¿Cuál es su velocidad? Respuesta sencilla. El rendimiento máximo ocurrirá si la memoria puede alcanzar la velocidad del procesador
3. ¿Cuál es su coste? Para que el sistema tenga sentido, debe tener un coste similar al resto de los componentes.

NUESTROS PROBLEMAS: ● Cuanto menor tiempo de acceso, mayor coste por bit. ● Cuanto mayor capacidad, menor coste por bit. ● Cuanto mayor capacidad, menor velocidad de acceso. La única solución posible es no utilizar un único componente de memoria, sino emplear varios, organizados en base a una jerarquía de memoria.

Diagrama

Descripción generada automáticamente con confianza media

**La última siendo la clave, disminuye la frecuencia…**

**Memoria caché:**

MOTIVACIÓN: ● En todos los ciclos de instrucción, el procesador accede a memoria al menos una vez, para leer la instrucción. ● Frecuentemente, una o más veces adicionales, para leer y/o almacenar resultados. La velocidad a la que el procesador puede ejecutar instrucciones está claramente limitada por el tiempo que se tarda en leer y/o escribir en memoria Esto es un: PROBLEMÓN

OBJETIVO ● El propósito de la caché es proporcionar un tiempo de acceso a memoria rápido (cercano al de los registros del procesador), pero ofreciendo un tamaño de memoria relativamente grande

La memoria caché contiene una parte de la memoria principal:Para el diseño de la memoria caché de nuestro sistema debemos tener en cuenta algunas condiciones: ● Tamaño de la caché. ● Tamaño del bloque. ● Función de correspondencia. ● Algoritmo de reemplazo. ● Política de escritura