

Informática
Guía Teórica
Ingeniería en Mecatrónica

BORQUEZ PEREZ, Juan Manuel-13567

Arquitectura de computadoras

1. ¿Por qué las computadoras se estructuran en capas?

La computadora está limitada a realizar operaciones aritméticas con números (suma, resta, multiplicación y división), realizar operaciones lógicas (por ejemplo realizar la comparación de dos números) y mover datos de una región de la memoria a otra. Estas acciones se comunican directamente a la computadora a través de instrucciones codificadas en una serie de bits (lógica digital), es decir un conjunto de unos y ceros que en el soporte físico están representados por niveles de tensión/corriente altos o bajos y que constituye el lenguaje que la máquina puede entender (lenguaje máquina). Una serie de instrucciones constituyen un programa, pero escribir las mismas en el lenguaje máquina es complicado e inconveniente para el usuario. Para eliminar esta complejidad el sistema de cómputo se estructura en capas, cada una de las cuales permite una abstracción de las capas inferiores y oculta la complejidad de las mismas aportando solo la funcionalidad más relevante a las capas superiores.

2. Describa la relación entre capas, máquinas virtuales y lenguajes

Cada capa de abstracción aporta funcionalidad a las capas superiores y se vale de la funcionalidad aportada por las capas de más bajo nivel.

En la capa de más bajo nivel (nivel 0) opera la máquina real (el hardware del sistema de cómputo) capaz de interpretar y ejecutar instrucciones en el lenguaje máquina, es decir basadas en la lógica digital ("unos y ceros"). A su vez en cierta capa se puede imaginar que opera una máquina virtual (dado que las capas de mayor nivel la ven como una máquina en sí misma aunque no tenga un soporte físico realmente) que solo puede trabajar con programas en su propio lenguaje. Estos programas son interpretados por un intérprete en una máquina virtual de menor nivel o son traducidos al lenguaje de una máquina virtual de menor nivel.

Entonces para que un programa escrito por el usuario en cierto lenguaje de programación sea entendible por la máquina real en el nivel 0, el mismo debe ser traducido a través de las capas a los lenguajes de las máquinas virtuales que operan en cada una de ellas hasta tener su representación en lógica digital.

3. Describa el concepto de trayectoria de datos

El conjunto de la ALU, los registros internos de la CPU y buses internos que interconectan estos componentes constituyen el camino de datos (es la trayectoria a través de la cual se pueden mover los datos).

La ALU así puede operar con los datos ubicados en dos registros (por ejemplo sumar el contenido de los registros o realizar otra operación) y el resultado de la operación almacenarse en otro registro de la CPU. A esto se denomina ciclo del camino de datos. A su

Informática
Guía Teórica
Ingeniería en Mecatrónica

vez este resultado puede ser posteriormente movido a alguna posición de la memoria principal de la computadora.

4. ¿Qué es el microprograma y cuál es su función?

Un microprograma es un conjunto de instrucciones el cual tiene la función de controlar la operación de la trayectoria de los datos.

Informática
Guía Teórica
Ingeniería en Mecatrónica

5. Realice una tabla con las capas de una arquitectura típica, y comente brevemente la función de cada una

Capas	Descripción
Nivel 0: Nivel de lógica digital	Los elementos integrantes de este nivel son las compuertas digitales. Estas compuertas tienen una o más entradas digitales (que aceptan señales de tensión o corriente que representan 1 o 0) y la salida de las mismas es el resultado de una operación sencilla (como OR o AND). A base de compuertas pueden construirse memorias elementales y a base de esas se construyen los registros. A su vez se puede construir la ALU en sí a base de compuertas
Nivel 1: Nivel de microarquitectura	La función principal consiste en tomar dos registros internos, hacer que la ALU lleve a cabo una operación con los mismo y almacenar el resultado en otro registro (o sea básicamente operar la trayectoria de datos).
Nivel 2: Nivel de arquitectura del conjunto de instrucciones (Nivel ISA)	Su función consiste en interpretar un conjunto de instrucciones para que luego puedan ser ejecutadas por él microprograma o por los circuitos de ejecución en hardware.
Nivel 3: Nivel máquina de sistema operativo	Casi todas las instrucciones de este nivel están también en el nivel ISA y cumple más o menos la misma función que este nivel. La diferencia es que cuenta con un nuevo conjunto de instrucciones, diferente organización de memoria, la capacidad de ejecutar más de un programa en simultáneo y otras características. Estas nuevas funciones son interpretadas por el SO en el nivel 2 y aquellas instrucciones compartidas entre nivel 2 y 3 son ejecutadas por el microprograma o por el control por hardware.
Nivel 4: Nivel de lenguaje ensamblador	Su función principal es lograr la traducción entre el nivel 4 y los niveles anteriores. El lenguaje de este nivel es una representación simbólica entendible por el usuario de los lenguajes numéricos (incomprensibles para el usuario) de los niveles más bajos. Ofrece así, un método para escribir programas para las máquinas en niveles más bajos. El programa antes de ser interpretado en esos niveles es traducido a través de un programa denominado ensamblador. A través de este nivel se pueden resolver problemas relacionados con el diseño e implementación de máquinas virtuales para máquinas de los niveles inferiores
Nivel 5: Nivel de lenguaje orientado hacia problemas	Proporciona lenguajes para ser utilizados directamente por programadores de aplicaciones que requieren resolver problemas específicos. Los programas en este nivel se traducen al nivel 3 o 4 mediante compiladores aunque en ocasiones son interpretados.

Informática
Guía Teórica
Ingeniería en Mecatrónica

6. ¿Qué es la arquitectura de una computadora, y cuáles son los aspectos de los que se ocupa?

La arquitectura de una computadora hace referencia a aquellos atributos que repercuten directamente en la ejecución lógica de los programas y por lo tanto son visibles o importantes para un programador. Por ejemplo, el tamaño de la palabra que maneja el CPU, el set de instrucciones o la memoria disponible son atributos arquitectónicos de una computadora

7. ¿Qué contiene la CPU?

La CPU contiene:

ALU (Unidad Aritmético Lógica): que es la unidad encargada de realizar las operaciones aritméticas (suma, resta, multiplicación y división) y las operaciones lógicas (AND, OR, determinar si un número es mayor a otro, etc).

UC (unidad de control): que es la unidad encargada de decodificar las instrucciones que llegan al CPU y enviar las correspondientes señales de control para llevarlas a cabo (ya sea una operación en la ALU o mover algún dato)

Registros: que son básicamente un conjunto de unidades de memoria de acceso rápido que sirven de almacén temporal de ciertos datos, instrucciones o información de control. Pueden ser de propósito general (los registros de la ALU) o registros especializados como por ejemplo los registros de dirección a memoria, registro de datos, registro de instrucción, contador de programa, etc.

8. Mencione los 3 grandes pasos que realiza la CPU para ejecutar una instrucción, y describa brevemente cada uno

Búsqueda: El PC indica la dirección en memoria de la próxima instrucción que debe ejecutarse. Se lee la posición en memoria indicada para obtener la instrucción y a través del bus de datos se lleva y almacena en el registro de instrucción

Decodificación: Se separa de la instrucción el código de instrucción asociado para determinar el tipo de operación a realizar. La UC decodifica. Además se determina si es necesario colocar una palabra de la memoria principal en algún registro de la CPU, en caso de ser así, la dirección en memoria del dato se almacena en un registro de dirección para buscar la palabra

Ejecución: La UC activa las señales de control correspondientes para ejecutar la instrucción ya sea una operación de la ALU o un movimiento de alguna palabra

Informática
Guía Teórica
Ingeniería en Mecatrónica

9. ¿Cuál es la ventaja de las CPU RISC sobre CISC?

Dado que las instrucciones implementadas en RISC son simples, la ejecución de las mismas es más rápida en comparación con las instrucciones complejas de CISC. La ejecución de una instrucción en una arquitectura RISC requiere de una menor cantidad de ciclos de reloj mientras que en una arquitectura CISC la ejecución de una instrucción puede requerir de varios ciclos. Esto favorece la segmentación en RISC respecto a CISC. Por otro lado, el código de operación de las instrucciones en RISC es de tamaño fijo mientras que en CISC es variable lo cual implica unidades de decodificación más complejas en CISC. Esta complejidad mayor en CISC también significa una mayor cantidad de transistores en el CPU y por lo tanto un mayor consumo energético. Por esto en los dispositivos de poco consumo (como los smartphones) es habitual el uso de procesadores basados en RISC.

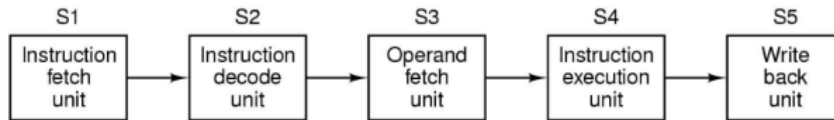
Informática

Guía Teórica

Ingeniería en Mecatrónica

10. Verdadero o Falso: Todas las instrucciones se ejecutan en 1 ciclo de CPU.

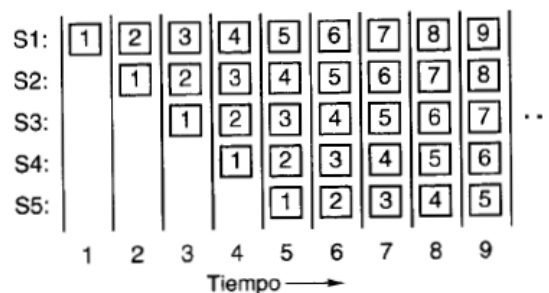
Falso dado que la ejecución de una instrucción requiere de 5 etapas cada una de las cuales implica un ciclo de la CPU como se indica:



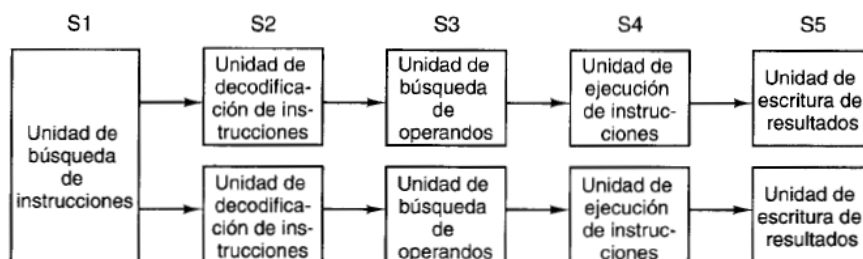
Por otro lado cuando la aseveración del enunciado también es falsa dado que en el caso de una arquitectura RISC, la ejecución de una meta instrucción requiere de varios ciclos de instrucción.

11. ¿Cuál es la diferencia entre el uso de filas de procesamiento y las arquitecturas superescalares?

El concepto de fila de procesamiento (pipeline) consiste en la segmentación en etapas de la ejecución de una instrucción, cada una de las cuales depende de hardware dedicado y que pueden operar de forma paralela. La figura del inciso anterior representa una línea de procesamiento de 5 etapas. De esta manera en cada ciclo de reloj se pueden ejecutar de forma simultánea etapas distintas de instrucciones distintas. Con esto se consigue que en cada ciclo de reloj concluya la ejecución de una instrucción. La siguiente imagen ilustra el concepto



Entendiendo que una fila de procesamiento es buena, para acelerar la ejecución de las instrucciones se pueden utilizar más filas de procesamiento individuales como se indica a continuación.



Para poder implementar este sistema, el par de instrucciones que se ejecutan en simultáneo en filas de procesamiento paralelas han de ser compatibles y no tener interdependencia. Condición que tiene que ser verificada en la compilación del programa o con el agregado de hardware para la detección y corrección de los errores.

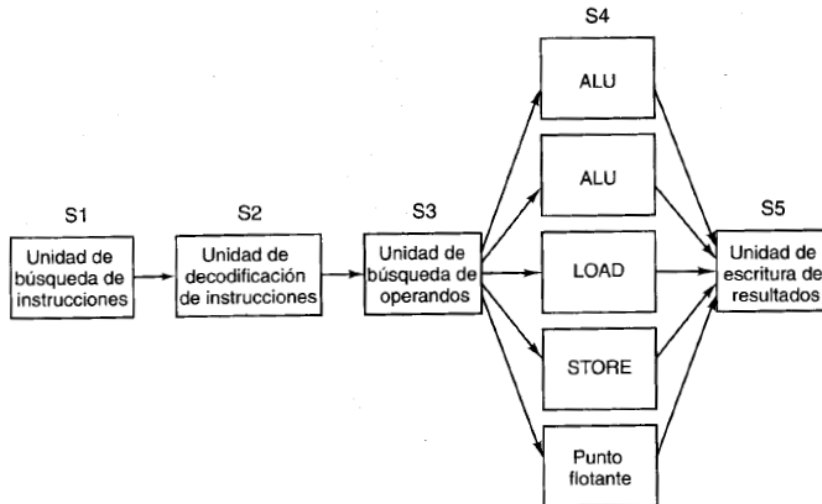
En principio es posible colocar más filas de procesamiento, pero esto aumenta la cantidad

Informática

Guía Teórica

Ingeniería en Mecatrónica

de hardware necesario. En su lugar la estrategia que se implementa es utilizar una sola línea de procesamiento pero a la misma se incorporan varias unidades de ejecución que operan en paralelo como se indica a continuación:



Esta estrategia se basa en la idea de que la tasa de que es mayor la tasa de emisión de instrucción por la etapa S3 es mucho mayor que la tasa de ejecución de las mismas por cada una de las unidades funcionales. Esto es cierto y en general las unidades de la etapa 4 tardan más de un ciclo del reloj en ejecutarse

12. ¿Cuál es la diferencia entre los multiprocesadores y las multicomputadoras?

Un multiprocesador es un sistema de cómputo con varias CPU cada una de las cuales dispone de una memoria local, pero que además dispone de una memoria compartida a la que se conectan a través de un bus común. En cambio las multicomputadoras están constituidas de muchas computadoras individuales que se interconectan a través de conexiones de red y no tienen una memoria compartida

13. Verdadero o Falso: el tamaño de la palabra de memoria es igual para todas las memorias

FALSO

El tamaño de la palabra que maneja una CPU viene dado por el tamaño de sus registros internos. El tamaño de estos registros varía de arquitectura en arquitectura. Así por ejemplo se pueden tener tamaños de palabra de 32 bits o de 64 bits. El tamaño de las unidades de memoria se formatea de acuerdo a este tamaño de palabra.

14. ¿Qué diferencia hay entre el esquema little endian y el esquema big endian?

La terminación (endiannes) hace referencia al modo en que se almacenan en memoria los datos que ocupan más de un Byte en memoria.

En la terminación big endian los bytes más significativos se ubican en posiciones en memoria anteriores a las posiciones de los bytes menos significativos mientras en la

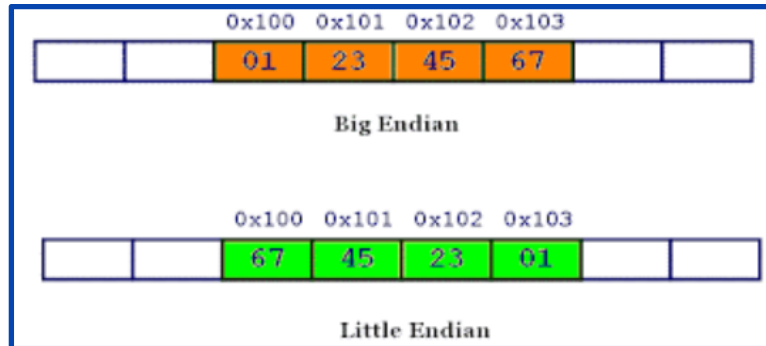
Informática

Guía Teórica

Ingeniería en Mecatrónica

terminación little endian los bytes menos significativos ocupan posiciones de memoria anteriores a la de los bytes más significativos de un mismo dato.

La siguiente imagen ilustra el concepto cuando se quiere guardar por ejemplo el dato: **0x01234567**



15. ¿Cuál es la función de la memoria caché?

La caché es una unidad de memoria de acceso más rápido que la memoria principal y que se encuentra más cerca del procesador (en la misma pastilla). Esta unidad almacena de forma especulativa (usando técnicas estadísticas) instrucciones y datos. Es decir que almacena los datos y las instrucciones que son usadas con más frecuencia para que cuando sean requeridas, la CPU las busque en la caché en lugar de buscarla en la memoria principal ya que esta última es más lenta. Si la palabra buscada no está en la caché entonces se busca en memoria. Pero si una gran parte de las palabras requeridas por la CPU se encuentran en la caché el tiempo de acceso medio a los datos se reduce considerablemente.

16. ¿Cómo se conecta la CPU con la memoria principal y dispositivos de E/S?

El CPU se conecta con la memoria y los dispositivos de E/S a través de buses de datos (para transmitir datos), de direcciones (para acceder a ciertas direcciones de memoria) o de control (para transmitir señales de control).

Para la conexión entre la memoria principal y el CPU se utiliza un bus y para la interconexión de los periféricos se emplea otro bus. A su vez se pueden emplear distintos estándares de buses para la conexión de distintos tipos de periféricos. Estos buses se interconectan a través de puentes (bridges).

17. ¿Para qué sirve el registro denominado “Program Counter” (Contador de Programa)?

El PC es un registro que almacena la dirección en memoria de la próxima instrucción que debe ejecutarse.

Informática
Guía Teórica
Ingeniería en Mecatrónica

18. Defina y enuncie las diferencias entre compilación, ensamblado e interpretación.

La compilación es la conversión de código de un lenguaje a otro, en un paso previo a su ejecución.

En la forma más tangible la compilación da un código binario ejecutable como salida y se denomina compilación ahead-of-time (AOT) (que sería antes de tiempo). Este tipo de compilación permite llevar a cabo optimizaciones y adaptaciones complejas a la máquina donde se ejecute el programa. Una vez compilado el programa no es necesario volver a compilarlo a menos que se lleven a cabo modificaciones y se puede ejecutar tantas veces como se quiera. Una de las características que ofrece es la posibilidad de ocultar el código fuente entre diferentes plataformas aunque es más complicado asegurar la compatibilidad

La forma alternativa de ejecutar un programa a partir del código en un lenguaje de programación es analizar el código y realizar los cálculos que éste indique de forma secuencial. En este caso se dice que el programa es interpretado. El programa que lleva a cabo la interpretación se denomina intérprete y debe ejecutarse en el sistema mientras se ejecuta el programa. Esta alternativa no ofrece la posibilidad de ocultar el código fuente pero sí permite la compatibilidad del código entre plataformas que cuenten con el intérprete. Dado que la traducción del código debe hacerse cada vez que se ejecuta el programa y en la medida que es necesario, en este caso la ejecución suele tomar más recursos en tiempo de ejecución y ser más lenta.

Si las instrucciones del lenguaje fuente son esencialmente representaciones simbólicas entendibles por el usuario de las instrucciones en lenguaje numérico de las capas de más bajo nivel, entonces el traductor se denomina ensamblador (como también se denomina al lenguaje fuente del programa) y al proceso de traducción se le denomina ensamblado

Cuando el lenguaje fuente es de alto nivel y el lenguaje objetivo es un lenguaje de máquina numérico o una representación simbólica del mismo el traductor se denomina compilador.

Una de las ventajas del lenguaje ensamblador es que permite el acceso a ciertos recursos de la máquina objetivo (recursos de hardware) que no son accesibles a través de los lenguajes de más alto nivel. Sin embargo todas las características que ofrece un lenguaje de alto nivel pueden obtenerse también a través de un programa escrito en ensamblador.

Una desventaja del ensamblador respecto de los lenguajes de alto nivel es la falta de compatibilidad a través de distintas plataformas. Las instrucciones del lenguaje ensamblador, al ser representaciones simbólicas de las instrucciones de las máquinas virtuales de más bajo nivel (y la máquina real), están en estrecha relación con la propia arquitectura de una familia de computadoras en específico.