

Introducción

A quien no le ha pasado que cuando uno está acostumbrado al uso de nuestro smartphone y probamos otro mas “potente” a veces solemos sentirlo mas “lento”, es decir, que algunas aplicaciones se “traban” o se siente una sensación de lentitud en algunas funciones aún cuando el teléfono mas potente tiene una velocidad mayor y hasta mas núcleos.

Realmente no hay una manera fácil de decirlo sin entrar a muchos tecnicismos que aburrirían hasta el que crea los procesadores, pero vamos a tratar de explicarlo de una manera que sea lo *suficientemente técnica para aclararlo a la vez que fácil de entender para los usuarios que no les interesan datos técnicos*.

Cada procesador es diferente, y no solo a diferencias en como se hacen, si no a diferencias en personalizaciones como el pipeline, tamaño del bus, métodos de fabricación, materiales de fabricación, cache, set de instrucciones y todas esas cosas que el usuario final rara vez ve ya que usualmente, el marketing solo se enfoca en la velocidad de reloj (los miles de Megahertz) y el numero de núcleos (single core, dual-core, quad-core entre otros).

Ahora imagínense todas las combinaciones posible de entre esas diferencias, cada una generando un rendimiento diferente, por mínimo que sea, es diferente y muchas veces no siempre es para bien. Usualmente, lo que importa en el número de núcleos y la velocidad de los núcleos, pero a veces el hecho de tener 2 núcleos no resulta mejor que tener un núcleo, ya que la manera en que el chip fue fabricado no fue de la mejor manera para que la arquitectura aproveche la ventaja de tener 2 núcleos. Lo mismo pasa comparando un doble núcleo contra un cuádruple núcleo.

Esto se ve mas comúnmente en dispositivos de gama baja en donde los procesadores no son bien optimizados para ahorrar gastos, así que podemos ver un dual-core en un smartphone barato con un desempeño de risa, mientras que un smartphone de un solo núcleo hacer maravillas.

Pero lo anterior solo es una cara de la moneda, ya que también el software (es decir el S.O. y las apps) tiene que ser optimizado en sus lineas de programación para poder usar las capacidades de 2, 4, 8 o 16 núcleos según sea el caso (no falta mucho para que veamos procesadores de 8 núcleos). Muchas veces y sobre todo en el ambiente libre que tiene Android, vemos aplicaciones que no están bien optimizadas para el uso de varios núcleos y no es hasta que el desarrollador actualiza la aplicación con nuevas lineas de códigos de optimización para varios núcleos que cuando esa app se “siente” mas rápida en nuestro recién adquirido smartphone quad-core.

Esto es mas común en el ambiente Android en donde realmente Google no tiene un control muy definido de como las app son desarrolladas y subidas al mercado de apps de Android, caso contrario pasa con Apple y su iPhone en donde en la App Store, todas las apps que se suben pasan por un riguroso proceso de análisis, las cuales tienen que estar optimizadas para trabajar aprovechando la potencia del mas reciente iPhone, iPad o iPod.

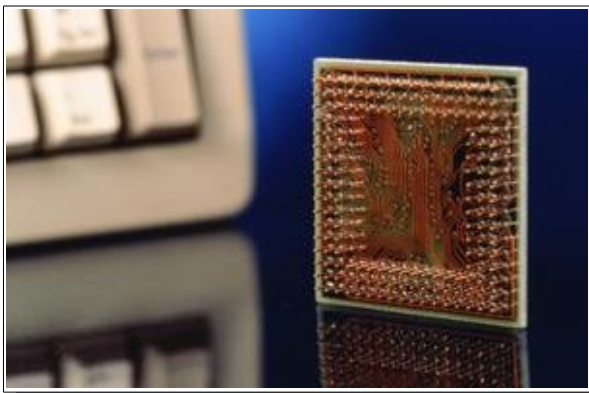
Como verán, es un conjunto de muchas variables el hecho de que un procesador pueda desempeñar

eficazmente el rendimiento que esperamos de éste, por eso siempre es bueno comparar, comparar y comparar hasta que estemos convencidos de que lo que vamos a comprar se ajusta realmente a nuestras necesidades y expectativas. Los famosos Benchmarks o pruebas de rendimiento que vemos en videos de Youtube no son realmente un indicador si el teléfono es bueno o no. Ayudan a ver solo la potencia de un procesador, pero recordemos que el verdadero rendimiento se ven cuando todos los factores que influyen en un dispositivos se ven todos unidos.

Los Nucleos

Una de las primeras decisiones a la hora de diseñar un microprocesador es decidir cual será su juego de instrucciones. La decisión es trascendente por dos razones;

- El juego de instrucciones decide el diseño físico del conjunto;
- Cualquier operación que deba ejecutarse en el microprocesador deberá poder ser descrita en términos de un lenguaje de estas instrucciones. Frente a esta cuestión caben dos filosofías de diseño; máquinas denominadas CISC y máquinas denominadas RISC.



Cuando hablamos de microprocesadores CISC, *computadoras con un conjunto de instrucciones complejo*, del inglés *complex instruction set computer*, y procesadores RISC, *computadoras con un conjunto de instrucciones reducido*, del inglés *reduced instruction set computer*, se piensa que los atributos complejo y reducido describen las diferencias entre los dos modelos de arquitectura para microprocesadores. Esto es cierto solo de forma superficial, pues se requiere de muchas otras características esenciales para definir los RISC y los CISC. Aún más, existen diversos procesadores que no se pueden asignar con facilidad a ninguna categoría

determinada.

Hasta hace solo algunos años, la división era tajante: RISC se utilizaba para entornos de red, mientras que CISC se aplicaba en ordenadores domésticos. Pero en la actualidad se alzan voces que afirman que CISC está agotando sus posibilidades, mientras otras defienden fervientemente que CISC ya ha alcanzado a RISC, adoptando algunas de sus principales características.

Un Nucleo - Single Core

Hablemos un poco de los Single-Core o procesadores sencillos de 1 núcleo de procesamientos. Son buenos mientras que no sean de aplicaciones de alto performance... Como así? si estos procesadores son excelentes para: navegar en internet mientras escuchas música y editas Word, power point, etc... Más o menos con estos 3 programas simultáneos el PC funciona fluido claro que ya depende del procesador y la velocidad de este.

Dos Nucleos - Dual Core

Son los mejores a la hora de aplicaciones como juegos a pesar de que los demás procesadores de 3, 4 y mas núcleos sean mas rápidos y sofisticados puesto que la mayoría de juegos solo soporta dual core, esto lo afirman los grandes de los juegos de PC como Ea Microsoft Etc. porque él solo hecho de colocar a un procesador de 3 núcleo o más a que cada núcleo se encargue de cierta tarea haría que el juego fuera de que consuma excesivos recursos costaría 3 veces más y sería mucho más pesado la verdad los procesadores de 3 o mas cores si son buenos para juegos pero es una pérdida de plata comprar un I7 de \$999.999 pesos colombianos a un Intel Core 2 Dúo de \$249.000 P Col sabiendo que aunque el I7 es la mejor tecnología de Intel en un juego que utiliza un solo proceso en el PC el core 2 dúo y el I7 tienen poca diferencia en el juego mientras que el I7 se destaca por poder utilizar multiproceso jugar escuchar música y navegar en internet a la vez cosa que el core 2 dúo no puede allí está la diferencia los de mas de 2 cores pueden hacer muchísimos procesos al mismo tiempo!

Mas de 3 Cores

Estos últimos ofrecen un óptimo desempeño en tareas de muchísimos sub procesos es decir con estos puedes oír música jugar navegar en IE Editar Word PowerPoint y mas con una fluidez espectacular...

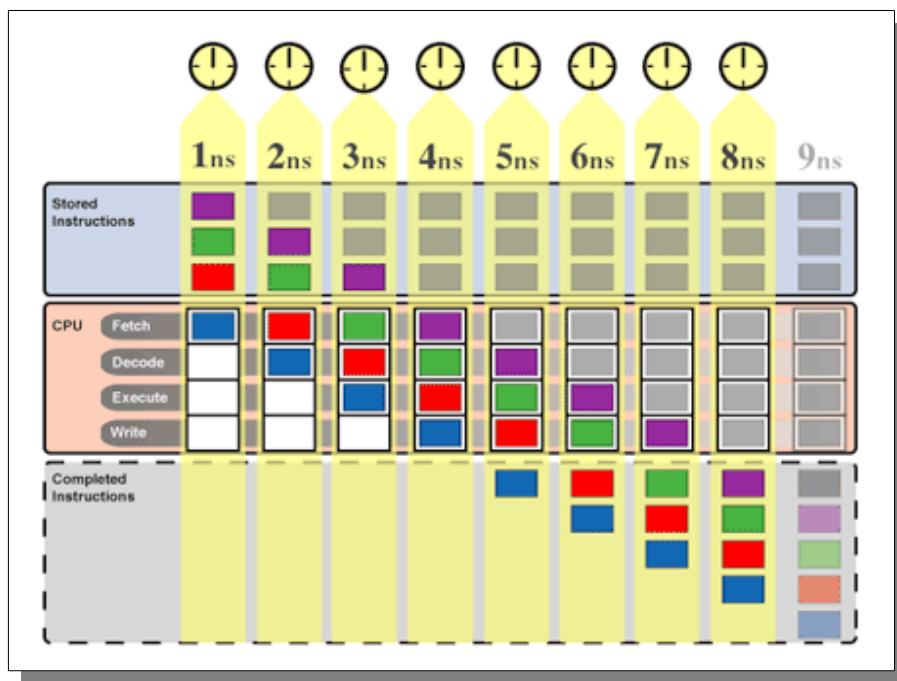
No todas las aplicaciones de software están diseñadas para tomar ventaja de las capacidades del procesador de cuatro núcleos. Si la mayor parte del software que se utiliza ha sido diseñado para funcionar con procesadores de núcleo único, no puedes observar un gran aumento de rendimiento al cambiar a un procesador de cuatro núcleos. Las aplicaciones que están diseñadas para ejecutar tareas al mismo tiempo serán capaz de ejecutarlas incluso con núcleos individuales en el procesador.

Que es el Pipeline?

Pipeline es un término perteneciente a la ingeniería de software, y consiste en una cadena de elementos de procesamiento ordenados de tal manera que la salida de cada elemento es la entrada del siguiente, con almacenamiento temporal de datos o buffering entre los procesos.

Suena complicado pero no lo es; el nombre quiere decir en español "tuberías", y el sistema es básicamente como el agua que circula por cañerías o tubos. En este caso el agua vendría a ser la información o los procesos.

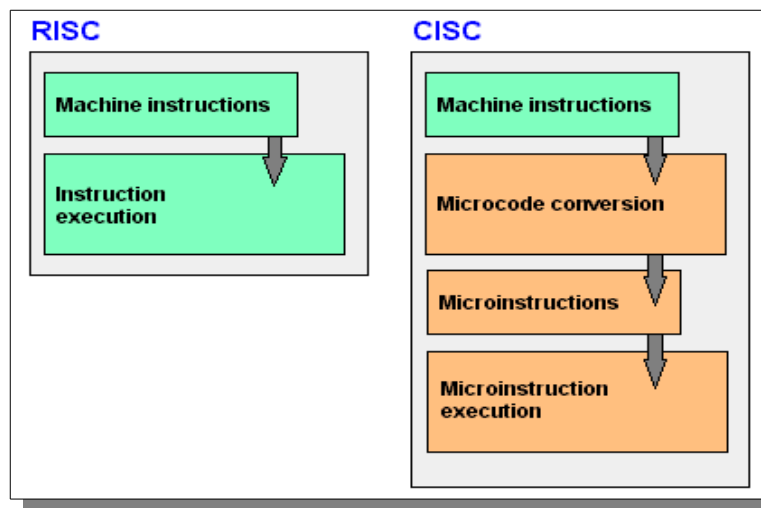
El pipeline es común verlo en sistemas operativos multitarea, como los que empleamos hoy en día en nuestras computadoras; se ejecutan una serie de procesos de manera simultánea, que son ejecutados luego de manera secuencial mediante un administrador de tareas dándoles diferente prioridad y capacidad de procesamiento, alternando entre un "pipeline" y los demás.



Al comienzo del primer nanosegundo, la instrucción azul entra en la etapa de fetch (traer). Comienza el segundo nanosegundo y la instrucción azul se mueve a la etapa de decodificación mientras la siguiente instrucción, la roja, se abre camino desde el almacén de código al procesador (esto es, entra a la etapa de fetch). Al comienzo del tercer nanosegundo, la instrucción azul avanza a la etapa de ejecución, la instrucción roja a la etapa de decodificación, y la instrucción verde entra a la etapa de fetch. En el cuarto, la instrucción azul avanza a la etapa de escritura, la roja a la de ejecución, la verde a la decodificación, y la violeta a la de fetch. Cuando se ha completado el cuarto nanosegundo y comienza el quinto, la instrucción azul ha pasado a través del pipeline y ha terminado su ejecución. Así podemos decir que al final de cuatro nanosegundos (= a cuatro ciclos de reloj) el procesador con pipeline ha completado una instrucción.

Al comienzo del quinto nanosegundo, el pipeline está completo y el procesador puede empezar a completar instrucciones al ritmo de una instrucción por nanosegundo. Esta tasa de ejecución de 1 instrucción/ns es un incremento de 4x sobre la tasa del procesador de ciclo simple, que era 0.25 instrucciones/ns (o 4 instrucciones cada 16 nanosegundos)

Tecnología RISC y CISC



Tecnología CISC

La tecnología CISC (Complex Instruction Set Computer) nació de la mano de Intel, creador en 1971 del primer microchip que permitiría el nacimiento de la informática personal. Más concretamente, sería en 1972 cuando aparecería el 8080, primer chip capaz de procesar 8 bits, suficiente para representar números y letras. Con la posibilidad de colocar todos los circuitos en un solo chip y la capacidad de manejar número y letras nacería la cuarta generación de ordenadores, la de los conocidos como PC u ordenadores personales.

Los microprocesadores CISC tienen un conjunto de instrucciones que se caracteriza por ser muy amplio y permitir operaciones complejas entre operandos situados en la memoria o en los registros internos.

Este tipo de arquitectura dificulta el paralelismo entre instrucciones, por lo que en la actualidad la mayoría de los sistemas CISC de alto rendimiento implementan un sistema que convierte dichas instrucciones complejas en varias instrucciones simples, llamadas generalmente microinstrucciones.

La microprogramación es una característica importante y esencial de casi todas las arquitecturas CISC. La microprogramación significa que cada instrucción de máquina es interpretada por un microprograma localizado en una memoria en el circuito integrado del procesador. Las instrucciones compuestas son decodificadas internamente y ejecutadas con una serie de microinstrucciones almacenadas en una ROM interna. Para esto se requieren de varios ciclos de

reloj, al menos uno por microinstrucción. Es así entonces como los chips CISC utilizan comandos que incorporan una gran diversidad de pequeñas instrucciones para realizar una única operación.

Cuando el sistema operativo o una aplicación requiere de una de estas acciones, envía al procesador el nombre del comando para realizarla junto con el resto de información complementaria que se necesite. Pero cada uno de estos comandos de la ROM del CISC varían de tamaño y, por lo tanto, el chip debe en primer lugar verificar cuanto espacio requiere el comando para ejecutarse y poder así reservárselo en la memoria interna. Además, el procesador debe determinar la forma correcta de cargar y almacenar el comando, procesos ambos que ralentizan el rendimiento del sistema.

El procesador envía entonces el comando solicitado a una unidad que lo descodifica en instrucciones más pequeñas que podrán ser ejecutadas por un nanoprocesador, una especie de procesador dentro del procesador. Y al no ser las instrucciones independientes, pues son instrucciones menores procedentes de la descodificación de una instrucción mayor, sólo puede realizarse una instrucción cada vez.

A través de la compleja circuitería del chip, el nanoprocesador ejecuta cada una de las instrucciones del comando. El desplazamiento por esta circuitería también ralentiza el proceso. Para realizar una sola instrucción un chip CISC requiere de cuatro a diez ciclos de reloj.

Entre las bondades de CISC destacan las siguientes:

- Reduce la dificultad de crear compiladores.
- Permite reducir el costo total del sistema.
- Reduce los costos de creación de software.
- Mejora la compactación de código.
- Facilita la depuración de errores.

Ejemplo de microprocesadores basados en la tecnología CISC:

- Intel 8086, 8088, 80286, 80386, 80486.
- Motorola 68000, 68010, 68020, 68030, 6840.

Tecnología RISC

Buscando aumentar la velocidad del procesamiento se descubrió en base a experimentos que, con una determinada arquitectura de base, la ejecución de programas compilados directamente con microinstrucciones y residentes en memoria externa al circuito integrado resultaban ser mas eficientes, gracias a que el tiempo de acceso de las memorias se fue decrementando conforme se mejoraba su tecnología de encapsulado.

La idea estuvo inspirada también por el hecho de que muchas de las características que eran incluidas en los diseños tradicionales de CPU para aumentar la velocidad estaban siendo ignoradas por los programas que eran ejecutados en ellas. Además, la velocidad del procesador en relación con la memoria de la computadora que accedía era cada vez más alta.

Debido a que se tiene un conjunto de instrucciones simplificado, éstas se pueden implantar por hardware directamente en la CPU, lo cual elimina el microcódigo y la necesidad de decodificar instrucciones complejas.

La arquitectura RISC funciona de modo muy diferente a la CISC, su objetivo no es ahorrar esfuerzos externos por parte del software con sus accesos a la RAM, sino facilitar que las instrucciones sean ejecutadas lo más rápidamente posible. La forma de conseguirlo es simplificando el tipo de instrucciones que ejecuta el procesador. Así, las instrucciones más breves y sencillas de un procesador RISC son capaces de ejecutarse mucho más aprisa que las instrucciones más largas y complejas de un chip CISC. Sin embargo, este diseño requiere de mucha más RAM y de una tecnología de compilador más avanzada.

La relativa sencillez de la arquitectura de los procesadores RISC conduce a ciclos de diseño más cortos cuando se desarrollan nuevas versiones, lo que posibilita siempre la aplicación de las más recientes tecnologías de semiconductores. Por ello, los procesadores RISC no solo tienden a ofrecer una capacidad de procesamiento del sistema de 2 a 4 veces mayor, sino que los saltos de capacidad que se producen de generación en generación son mucho mayores que en los CISC.

Los comandos que incorpora el chip RISC en su ROM constan de varias instrucciones pequeñas que realizan una sola tarea. Las aplicaciones son aquí las encargadas de indicar al procesador qué combinación de estas instrucciones debe ejecutar para completar una operación mayor.

Además, los comandos de RISC son todos del mismo tamaño y se cargan y almacenan del mismo modo. Al ser estas instrucciones pequeñas y sencillas, no necesitan ser descodificadas en instrucciones menores como en el caso de los chips CISC, pues ya constituyen en sí unidades

descodificadas. Por ello, el procesador RISC no gasta tiempo verificando el tamaño del comando, en descodificarlo ni en averiguar cómo cargarlo y guardarlo.

El procesador RISC puede además ejecutar hasta 10 comandos a la vez pues el compilador del software es el que determina qué comandos son independientes y por ello es posible ejecutar varios a la vez. Y al ser los comandos del RISC más sencillos, la circuitería por la que pasan también es más sencilla. Estos comandos pasan por menos transistores, de forma que se ejecutan con más rapidez. Para ejecutar una sola instrucción normalmente les basta con un ciclo de reloj.

Entre las ventajas de RISC tenemos las siguientes:

- La CPU trabaja mas rápido al utilizar menos ciclos de reloj para ejecutar instrucciones.
- Utiliza un sistema de direcciones no destructivas en RAM. Eso significa que a diferencia de CISC, RISC conserva después de realizar sus operaciones en memoria los dos operandos y su resultado, reduciendo la ejecución de nuevas operaciones.
- Cada instrucción puede ser ejecutada en un solo ciclo del CPU

Ejemplo de microprocesadores basados en la tecnología CISC:

- MIPS, Millions Instruction Per Second.
- PA-RISC, Hewlett Packard.
- SPARC, Scalable Processor Architecture, Sun Microsystems.
- POWER PC, Apple, Motorola e IBM.

Arquitectura Von Newman y Harvard

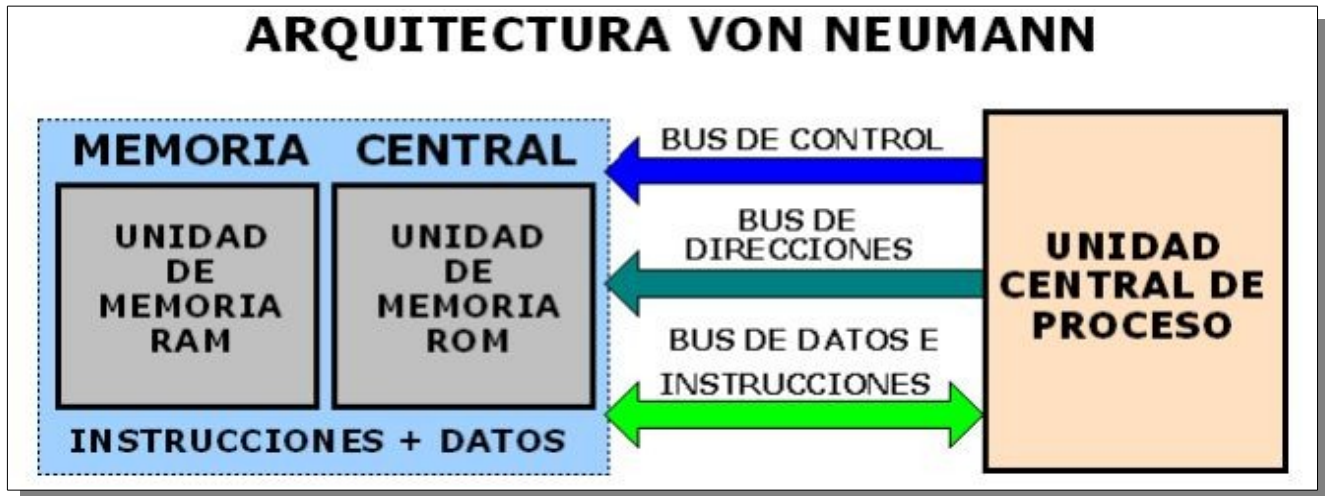
Arquitectura de von Neumann

Tradicionalmente los sistemas con microprocesadores se basan en esta arquitectura, en la cual la unidad central de proceso (CPU), está conectada a una memoria principal única (casi siempre sólo RAM) donde se guardan las instrucciones del programa y los datos. A dicha memoria se accede a través de un sistema de buses único (control, direcciones y datos).

En un sistema con arquitectura Von Neumann el tamaño de la unidad de datos o instrucciones está fijado por el ancho del bus que comunica la memoria con la CPU. Así un microprocesador de 8 bits con un bus de 8 bits, tendrá que manejar datos e instrucciones de una o más unidades de 8 bits (bytes) de

longitud. Si tiene que acceder a una instrucción o dato de más de un byte de longitud, tendrá que realizar más de un acceso a la memoria.

El tener un único bus hace que el microprocesador sea más lento en su respuesta, ya que no puede buscar en memoria una nueva instrucción mientras no finalicen las transferencias de datos de la instrucción anterior.

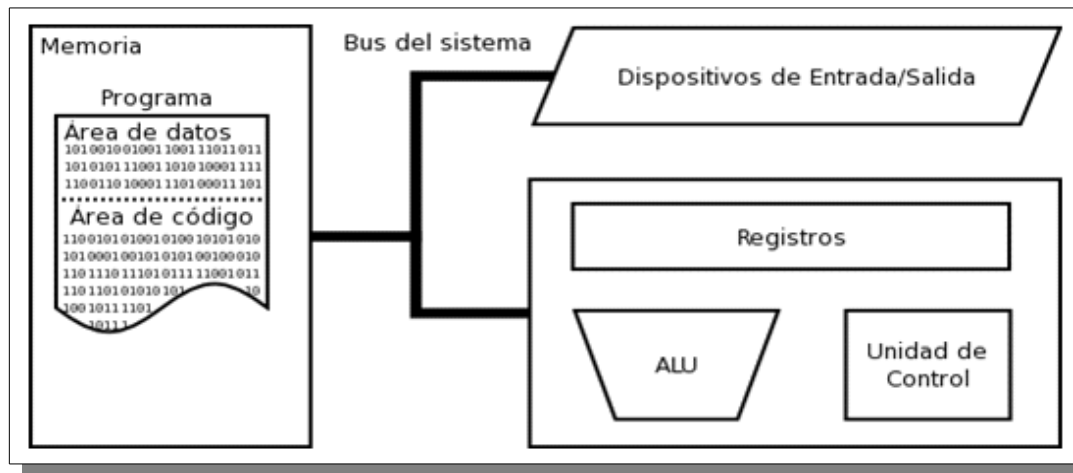


Las principales limitaciones que nos encontramos con la arquitectura Von Neumann son:

- La limitación de la longitud de las instrucciones por el bus de datos, que hace que el microprocesador tenga que realizar varios accesos a memoria para buscar instrucciones complejas.
- La limitación de la velocidad de operación a causa del bus único para datos e instrucciones que no deja acceder simultáneamente a unos y otras, lo cual impide superponer ambos tiempos de acceso

La arquitectura Von Neumann describe a la computadora con 4 secciones principales: la unidad lógica y aritmética (ALU), la unidad de control, la memoria, y los dispositivos de entrada y salida (E/S).

Los ordenadores con arquitectura Von Neumann constan de las siguientes partes:



La arquitectura Von Neumann realiza o emula los siguientes pasos secuencialmente:

1. Obtiene la siguiente instrucción desde la memoria en la dirección indicada por el contador de programa y la guarda en el registro de instrucción.
2. Aumenta el contador de programa en la longitud de la instrucción para apuntar a la siguiente.
3. Descodifica la instrucción mediante la unidad de control. Ésta se encarga de coordinar el resto de componentes del ordenador para realizar una función determinada.
4. Se ejecuta la instrucción. Ésta puede cambiar el valor del contador del programa, permitiendo así operaciones repetitivas.
5. Regresa al paso N° 1.

La mayoría de las computadoras todavía utilizan la arquitectura Von Neumann, propuesta a principios de los años 40 por John Von Neumann.

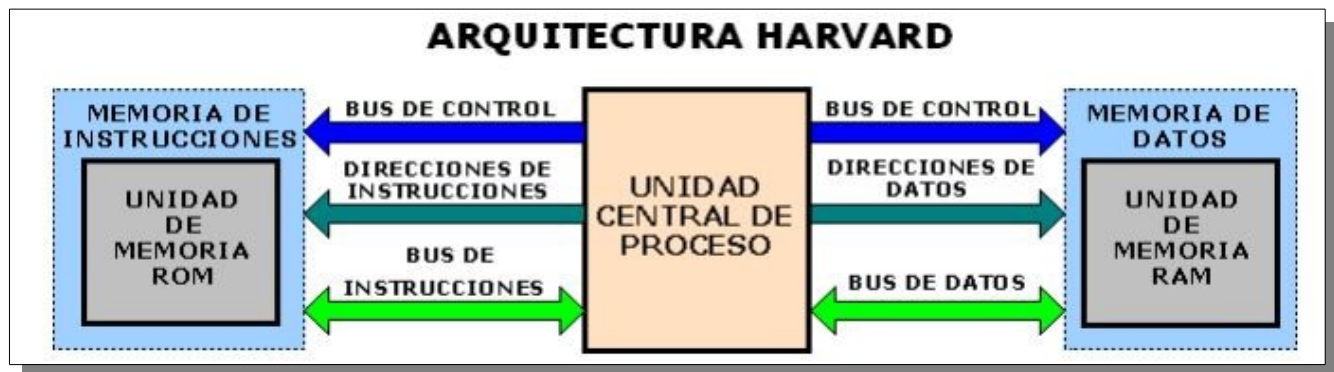
Arquitectura Harvard

Este modelo, que utilizan los Microcontroladores PIC, tiene la unidad central de proceso (CPU) conectada a dos memorias (una con las instrucciones y otra con los datos) por medio de dos buses diferentes.

Una de las memorias contiene solamente las instrucciones del programa (Memoria de Programa), y la otra sólo almacena datos (Memoria de Datos).

Ambos buses son totalmente independientes lo que permite que la CPU pueda acceder de forma independiente y simultánea a la memoria de datos y a la de instrucciones. Como los buses son independientes estos pueden tener distintos contenidos en la misma dirección y también distinta longitud.

También la longitud de los datos y las instrucciones puede ser distinta, lo que optimiza el uso de la memoria en general.



Para un procesador de Set de Instrucciones Reducido, o RISC (Reduced Instrucción Set Computer), el set de instrucciones y el bus de memoria de programa pueden diseñarse de tal manera que todas las instrucciones tengan una sola posición de memoria de programa de longitud.

Además, al ser los buses independientes, la CPU puede acceder a los datos para completar la ejecución de una instrucción, y al mismo tiempo leer la siguiente instrucción a ejecutar.

Ventajas de esta arquitectura:

- El tamaño de las instrucciones no está relacionado con el de los datos, y por lo tanto puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa, logrando así mayor velocidad y menor longitud de programa.
- El tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una mayor velocidad en cada operación.

Recibe el nombre por el ordenador Harvard Mark I, desarrollado en esa universidad de Massachusetts por Howard Aiken

Conclusiones

- Hoy en día, los programas cada vez más grandes y complejos demandan mayor velocidad en el procesamiento de información, lo que implica la búsqueda de microprocesadores más rápidos y eficientes.
- Los avances y progresos en la tecnología de semiconductores han reducido las diferencias en las velocidades de procesamiento de los microprocesadores con las velocidades de las memorias, lo que ha repercutido en nuevas tecnologías en el desarrollo de microprocesadores. Hay quienes consideran que en breve los microprocesadores RISC sustituirán a los CISC, pero existe el hecho que los microprocesadores CISC tienen un mercado de software muy difundido.
- En la década de los sesentas, la microprogramación era la técnica más apropiada para la tecnología de memorias existentes. En consecuencia, los procesadores se dotaron de poderosos conjuntos de instrucciones, dando surgimiento a la arquitectura CISC.
- Las arquitecturas CISC utilizadas desde hace 15 años han permitido desarrollar un gran número

de productos de software. Sin embargo, simultáneamente aumentan las aplicaciones en las cuales la capacidad de procesamiento que se pueda obtener del sistema es más importante que la compatibilidad con el hardware y el software anteriores. Por ello, todos los productores de estaciones de trabajo de renombre, han pasado en pocos años, de los procesadores CISC a los RISC, lo cual se refleja en el fuerte incremento anual del número de procesadores RISC.

- Cada usuario debe decidirse a favor o en contra de determinada arquitectura de procesador en función de la aplicación concreta que quiera realizar. Nunca será decisiva únicamente la capacidad de procesamiento del microprocesador; se debe considerar por igual la capacidad real que puede alcanzar el sistema en su conjunto.
- Si bien el campo de aplicaciones de la arquitectura RISC crece con fuerza, esto no equivale al fin de la arquitectura CISC, que también seguirá perfeccionándose adoptando técnicas típicas de los procesadores RISC, a fin de encontrar nuevas rutas para el incremento de sus capacidades.
- Cuando en 1990 Apple lanza su primer Power Macintosh y anuncia que se pasa al RISC, muchos auguraron que la tecnología de instrucciones complejas tenía los días contados. Pero desde entonces han pasado 17 años y los fabricantes de procesadores CISC han seguido aumentando el rendimiento, y lo que es más importante, han conseguido mantener los precios de los chips muy bajos.
- Intel ha mantenido hasta ahora que la arquitectura CISC puede estar perfectamente a la altura de la RISC. Pero la progresiva asimilación de técnicas propias del RISC en sus chips parecen revelar un abandono progresivo de la arquitectura de instrucciones complejas por la de instrucciones más reducidas
- AMD e Intel siguen caminos distintos en el rendimiento de sus microprocesadores. Intel se decidió por aumentar los MHz para aumentar el rendimiento mientras que AMD se decantó por aumentar el número de instrucciones que se pueden realizar por ciclo de reloj.