

Organización de la CPU - Arquitectura Von Neumann

(Extractos del capítulo 4 de Pat & Patel)

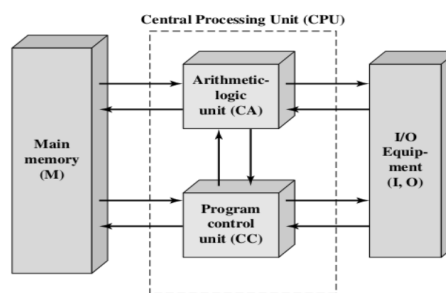
Para que una computadora pueda completar una tarea, se necesitan dos cosas:

1. Un programa que especifique lo que la computadora debe hacer para completar la tarea
2. La computadora misma que sea capaz de llevar a cabo la tarea

El programa consiste de un set de instrucciones, cada una especifica una parte bien definida del trabajo que la computadora debe hacer.

La instrucción es la representación más pequeña de un programa. Una computadora solo tiene dos opciones, sabe ejecutar la instrucción o no.

En 1946, John Von Neumann propuso un modelo fundamental de una computadora diseñada para ejecutar programas. La siguiente figura muestra sus componentes básicos.



El modelo consiste de 3 partes:

1. Memoria
2. CPU
 - a. Unidad de procesamiento
 - b. Unidad de Control
3. Entradas / Salidas

Memoria

La memoria es la parte de la computadora en la que se almacenan programas y datos. Un computador de programa almacenado es el que almacena las instrucciones del programa en la memoria electrónica y es característico (nació) de las arquitecturas Von Neumann.

¿Que entienden cuando se habla de una memoria de 256 megas (2^{28}) por 8 bits? Esto quiere decir que se tiene una memoria que consiste de 2^{28} diferentes posiciones de memoria, cada una capaz de almacenar 8 bits de información.

Cuando se tiene esto se dice que la memoria tiene un espacio de direcciones ("address space") de 2^{28} posiciones y una direccionabilidad ("addressability") de 8 bits.

Es importante resaltar que si se tiene una memoria de 2^k posiciones de memoria, se necesitan direcciones ("addresses") de al menos k bits.

000	
001	
010	
011	
100	00000110
101	
110	00000100
111	

Unidad Central de Procesamiento

La CPU (Unidad Central de Procesamiento o procesador) es el “cerebro” de la computadora. Su función es ejecutar programas almacenados en la memoria principal. Para esto la CPU necesita buscar (fetch) y decodificar (decode) instrucciones para después ejecutarlas (execute) una tras otra. Esta sucesión de pasos se conoce como el ciclo de **búsqueda-decodificación-ejecución** y es fundamental para el funcionamiento de todas las computadoras.

Los procesadores de computadoras modernas pueden consistir en múltiples y complejas unidades funcionales, cada una capaz de ejecutar operaciones aritméticas básicas (suma, resta), complejas (punto flotante) y operaciones lógicas básicas (AND, OR).

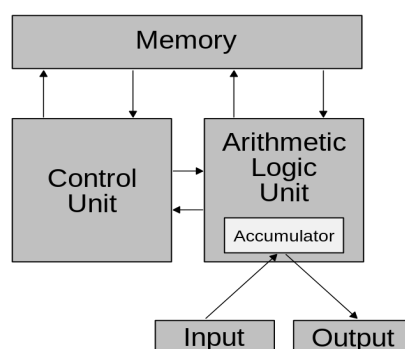
La unidad de procesamiento más sencilla, y en la que normalmente se piensa cuando se habla de la arquitectura de Von Neumann, es la ALU. ALU significa unidad aritmético-lógica y es una unidad funcional capaz de realizar operaciones sencillas como suma, resta y operaciones booleanas.

Normalmente el tamaño de los datos que puede procesar la ALU se conoce como *tamaño de palabra* de la computadora y cada elemento se conoce como palabra (“word”). Por ejemplo, en una computadora con un ALU capaz de procesar datos de 16-bits se dice que tiene palabras de 16 bits.

La unidad de control se encarga de buscar instrucciones de la memoria principal y determinar su tipo. Además, es la que se encarga de llevar el control sobre dos aspectos principales:

- En qué punto se encuentra en el proceso de ejecución de un programa
- En qué punto se encuentra en el proceso de ejecución de cada instrucción

La CPU también contiene una memoria pequeña de alta velocidad que sirve para almacenar resultados temporales. Esta memoria se compone de varios registros, cada uno de los cuales tiene cierto tamaño y función. Por lo general, todos los registros tienen el mismo tamaño.



E/S

Antes de que una computadora pueda procesar información, está información debe primero llegar hasta la computadora.

Para poder utilizar los resultados de ese procesamiento, los mismos deben mostrarse de alguna forma fuera de la computadora.

Muchos dispositivos existen con estos propósitos. Generalmente se conocen como dispositivos periféricos porque son accesorios al proceso de información, pero no son menos importantes.

Características más importantes

Las computadoras son máquinas de arquitectura von Neumann cuando:

1. Tanto los programas como los datos se almacenan en una memoria en común. Esto hace posible la ejecución de instrucciones de la misma forma que los datos.
2. Cada celda de memoria de la máquina se identifica con un número único, llamado dirección.
3. Cada programa se ejecuta de forma secuencial que, en el caso de que no haya instrucciones especiales, comienza con la primera instrucción.

Organización de la CPU - Arquitectura Harvard

Es una arquitectura de computadora con pistas de almacenamiento y de señal físicamente separadas para las instrucciones y para los datos. El término proviene de la computadora Harvard Mark I basada en relés, que almacenaba las instrucciones sobre cintas perforadas (de 24 bits de ancho) y los datos en interruptores electromecánicos. Estas primeras máquinas tenían almacenamiento de datos totalmente contenido dentro la unidad central de proceso, y no proporcionaban acceso al almacenamiento de instrucciones como datos. Los programas necesitaban ser cargados por un operador; el procesador no podría arrancar por sí mismo.

CISC vs RISC

Las primeras computadoras tenían un conjunto de instrucciones pequeños y sencillos las cuales podían ejecutar, pero la búsqueda de computadoras más potentes llevo, entre otras cosas, a instrucciones individuales más potentes.

Casi desde el principio se descubrió que con instrucciones individuales más complejas a menudo era posible acelerar la ejecución de los programas, aunque las instrucciones individuales tardaran más en ejecutarse.

A veces era tan sencillo como observar que con frecuencia ocurrían las mismas dos instrucciones en sucesión, de modo que una sola instrucción podía realizar el trabajo de ambas.

Las instrucciones más complejas eran mejores porque a veces hacían posible traslapar la ejecución de operaciones individuales o ejecutarlas en paralelo empleando diferente hardware.

En las costosas computadoras de alto rendimiento el costo de este hardware adicional podía justificarse fácilmente. Fue así como las computadoras de alto precio y mejor rendimiento comenzaron a tener muchas más instrucciones que las de bajo costo.

Sin embargo, el creciente costo de la creación de software y las necesidades de compatibilidad de las instrucciones hicieron necesario implementar instrucciones complejas incluso en las computadoras de bajo costo en las que el precio era más importante que la velocidad.

Hacia finales de los 50's, IBM se dio cuenta de las ventajas, tanto para ellos como para los clientes, que tenía manejar una sola familia de máquinas, todas las cuales ejecutaban las mismas instrucciones.

IBM introdujo el termino arquitectura para describir este nivel de compatibilidad. Una nueva familia de computadoras tenía una sola arquitectura, pero muchas implementaciones diferentes, todas las cuales podían ejecutar el mismo programa, la única diferencia era el precio y la velocidad.

El resultado fue la arquitectura IBM System/360, una familia de computadoras compatibles que abarcaba casi dos órdenes de magnitud, tanto en precio como en velocidad. Solo se usaba una implementación directa en hardware (es decir, no interpretada) en los modelos más caros.

En 1980 un grupo de Berkeley dirigido por David Patterson y Carlos Sequin comenzó a diseñar chips de CPU VLSI que no utilizaban interpretación. Ellos usaron el termino RISC para este concepto y llamaron a su chip de CPU RISA I.

En 1981 en Stanford, John Hennessy diseño y fabrico un chip un tanto diferente al que llamo MIPS.

Estos nuevos procesadores tenían diferencias significativas respecto a los procesadores comerciales de la época. Dado que estas nuevas CPU no tenían que ser compatibles con productos existentes, sus diseñadores estaban en libertad de escoger nuevos conjuntos de instrucciones que maximizaran el rendimiento total del sistema. Pronto se dieron cuenta que era más importante la cantidad de instrucciones que podían emitirse al mismo tiempo, que lo que duraba cada instrucción individual en ejecutarse.

La característica que llamó la atención de estos procesadores era el número relativamente pequeño de instrucciones disponibles, por lo regular unas 50 instrucciones (mucho menor que las 200 del DEC). De hecho, el acrónimo RISC significa Reduced Instruction Set Computer, en contraste con CISC Complex Instruction Set Computer.

Esto comenzó una discusión entre los partidarios de RISC y los de CISC (VAX, Intel, IBM). Los partidarios de RISC argumentaban que incluso si una maquina RISC requería 4 o 5 instrucciones para hacer lo que una instrucción CISC, si las instrucciones RISC son 10 veces más rápidas (ya que CISC tiene que interpretar), RISC gana.

Referencias:

1. Yale N. Patt & Sanjay J. Patel (2005)

Introduction to computing systems [Cap 4]

2. Andrew S. Tanenbaum (2000)

Organización de computadoras - Un enfoque estructurado. [Cap 2]