

## ***Principios de diseño Risc (para Aplicaciones Especificas)***

Se intentara presentar de manera general la filosofía básica de diseño de estas máquinas, teniendo en cuenta que dicha filosofía puede presentar variantes. Es muy importante conocer estos principios básicos, pues de éstos se desprenden algunas características importantes de los sistemas basados en microprocesadores RISC.

En el diseño de maquinas Risc se tienen cinco pasos importantes:

1. Analizar las aplicaciones para encontrar las operaciones claves.
2. Diseñar un bus de datos que sea óptimo para esas operaciones claves.
3. Diseñar instrucciones que realicen las operaciones clave utilizando el bus de datos.
4. Agregar nuevas instrucciones sólo si no hacen más lenta a la máquina.
5. Repetir el proceso para otros recursos.

El primer punto se refiere a que el diseñador deberá encontrar qué es lo que hacen en realidad los programas que se pretender ejecutar. Ya sea que los programas a ejecutar sean del tipo algorítmicos tradicionales o estén dirigidos a robótica o al diseño asistido por computadora.

La parte medular de cualquier sistema es la que contiene los registros, el ALU y los “buses” que los conectan. Se debe optimizar este circuito para el lenguaje o aplicación en cuestión. El tiempo requerido (denominado tiempo del ciclo del bus de datos) para extraer los operandos de sus registros, mover los datos a través de la ALU y almacenar el resultado de nuevo en un registro, deberá hacerse en el tiempo más corto posible.

El siguiente punto a tratar es diseñar instrucciones de máquina que hagan un buen uso del bus de datos. Por lo general se necesitan solo unas cuantas instrucciones y modos de direccionamiento; solo se deben colocar instrucciones adicionales si serán *usadas* con frecuencia y *no* reducen el desempeño de las más importantes.

Siempre que aparezca una nueva y atractiva característica, deberá analizarse y ver la forma en que se afecta al ciclo de bus de datos. Si se incrementa el tiempo del ciclo, probablemente no vale la pena tenerla.

Por ultimo, el proceso anterior debe repetirse para otros recursos dentro del sistema, tales como memoria cache, administración de memoria, etc. Una vez planteadas las características principales de la arquitectura Risc así como la filosofía de su diseño, podríamos extender el análisis y estudio de cada una de las características importantes de las arquitecturas Risc y las implicaciones que estas tienen.

### ***Propiedades de un sistema tipo RISC.***

1. Estándar, tamaño arreglado de instrucción, igual para la computadora la longitud de la palabra y para el ancho del bus de datos (con la estipulación algunos nuevos sistemas la longitud de la palabra y del bus de datos en un entero múltiple del tamaño de la instrucción como en estos nuevos sistemas de 64bits y en algunos sistemas de 32 bits pero con un bus de datos de 64bits).
2. El tiempo de la ejecución es estándar para todas las instrucciones, preferiblemente dentro de un ciclo singular de CPU (con la estipulación esta minoría de instrucciones igual como divide tendrá que ser ejecutado en mas de un solo ciclo).
3. La implementación del pipeline deberá ser eficiente.
4. Las instrucciones en una primera consideración tienen un formato fijo de 32 bits, y utilizan los tres formatos estándar: el Tipo R, Tipo I y Tipo J.
5. poseen 32 registros de uso general (aunque algunos sistemas nuevos poseen hasta 64 registros de uso general, especialmente para operaciones aritméticas, lógicas, etc.).
6. Utilizan generalmente mas de 3 modos de direccionamiento, con frecuencia son usados los modos registro, relativo al PC, Directo e Inmediato, pero podrían usarse cualquier otro tipo dependiendo del análisis hecho por los diseñadores.