

Revisão Prova 1 de Organização de Computadores Luciano L. Caimi

Erickson Giesel Müller

Conteúdos

1. Arquitetura Von Neumann
2. Arquitetura Harvard
3. Arquitetura RISC
4. Arquitetura CISC
5. Pipeline
6. Hierarquia de memória
 - (a) Cache
 - (b) Memória Principal RAM
 - (c) Memória Virtual
 - (d) Armazenamento de Massa

1 Arquitetura CISC

A arquitetura CISC apresenta instruções complexas que demandam um **grande número de ciclos** para serem executadas. Dezenas de modos de endereçamento, instruções de tamanhos variados e referência a operandos na memória principal. Nas arquiteturas CISC fica mais difícil implementar o pipeline. A unidade de controle é microprogramada. Menos instruções por ciclo.

2 Arquitetura RISC

Instruções mais simples, demandando um número **fixo** de ciclos de máquina para a sua execução. Uso de poucos e simples modos de endereçamento. Poucos formatos de instruções.

Apenas instruções de load/store referenciam operandos na memória principal.

Cada fase de processamento da instrução tem a duração fixa igual a um ciclo de máquina. O formato fixo das instruções facilita o pipeline. As instruções são executadas, na sua maioria, em apenas um ciclo de máquina.

A unidade de controle é em geral hardwired, ou seja, não há microprograma para interpretar as instruções. Arquitetura orientada a registrador, todas as operações aritméticas são realizadas entre registradores.

A desvantagem de usar uma arquitetura com um único ciclo é que o ciclo do clock se torna ineficiente, pois este deve ser do tamanho da instrução mais lenta. Isso é um grande problema quando se considera instruções mais complexas como por exemplo multiplicação de ponto flutuante.

3 Pipeline

4 Memória Cache

4.1 Cache com Mapeamento Direto

Para calcular a linha da memória cache:

$$i = j \mod m$$

onde:

i é o número da linha na memória cache

j é o número de blocos na memória RAM

m é o número de blocos na memória cache

Todas as relações de endereço são em potência de 2. Assim, conseguimos encontrar a linha na memória cache, mas, em uma operação de busca, ainda precisamos saber em qual bloco está a informação, para isso usamos duas informações de controle:

4.2 Bit de Válido

Indica se a entrada do cache está escrita com alguma informação.

4.3 Bit de Tag

Complementa o cálculo de endereço com a sintaxe $[tag]j \mod m$. Os bits de tag informam em qual parte da memória cache está a informação de acordo com a linha calculada.