

Universidade Federal do Rio de Janeiro
Informática DCC/IM

Arquitetura de Computadores II

Microarquiteturas Avançadas

Gabriel P. Silva

Introdução

- **As arquiteturas dos processadores têm evoluído ao longo dos anos, e junto com ela o conceito de arquitetura avançada tem se modificado.**
- **Nos anos 60 os computadores ditos de grande porte (IBM, CDC) eram considerados exemplos de alto desempenho.**
- **Um pouco depois surgiram os supercomputadores, arquiteturas específicas para a realização de cálculos com valores reais dispostos em arranjos de matrizes e vetores (Cray).**

Introdução

- **No início dos anos 80 surgiram os primeiros microcomputadores, com desempenho muito longe dos computadores de grande porte e supercomputadores.**
- **A evolução das tecnologias de integração de componentes em larga escala permitiu que esses processadores evoluíssem em desempenho, atingindo um amplo espectro de aplicações comerciais.**
- **Em nosso curso iremos estudar as principais características da arquitetura dos microprocessadores em sua evolução até os dias de hoje.**

Arquiteturas CISC

- **Características:**

- **Instruções complexas demandando um número grande e variável de ciclos de máquina para sua execução.**
- **Uso de diversos modos de endereçamento de operandos.**
- **Instruções com formato muito variável.**
- **Diferentes tipos de instruções podem referenciar operandos na memória principal.**
- **Cada fase do processamento da instrução pode ter duração variável em função da complexidade.**

Arquiteturas CISC

- **Conseqüências:**
 - **Implementação com uso de pipeline é difícil.**
 - **A taxa média de execução das instruções por ciclo tende a ser bastante inferior a 1 CPI.**
 - **A unidade de controle é em geral microprogramada.**
 - **Códigos compactos podem ser gerados pelos compiladores.**

Arquiteturas RISC

- **Características:**

- **Instruções mais simples demandando um número fixo de ciclos de máquina para sua execução;**
- **Uso de poucos modos simples de endereçamento de operandos;**
- **Poucos formatos diferentes de instruções**
- **Apenas as instruções de “load” e “store” referenciam operandos na memória principal;**
- **Cada fase de processamento da instrução tem a duração fixa igual a um ciclo de máquina.**

Arquiteturas RISC

- **Conseqüências:**
 - **Implementadas com o uso do pipeline;**
 - **A taxa média de execução de instruções por ciclo de máquina é próxima de 1 CPI;**
 - **A unidade de controle é em geral “hardwired”;**
 - **Processo de compilação é complexo e requer cuidados especiais para otimização do desempenho do código gerado.**

Arquiteturas com Desempenho Superior a uma Instrução por Ciclo

- **ARQUITETURAS SUPERESCALARES**
 - Execução de múltiplas instruções, escalonadas por “hardware” e/ou “software”, concorrentemente.
- **ARQUITETURAS VLIW (Very Long Instruction Word)**
 - Execução de múltiplas operações, escalonadas por “software”, concorrentemente.
- **ARQUITETURAS Multithreading e SMT**
 - Permitem a execução simultânea de mais de uma thread no processador.
- **ARQUITETURAS MULTICORE**
 - Uso de vários processadores em um mesmo encapsulamento, compartilhando ou não a cache de nível 2.

Arquiteturas Superescalares

- **As máquinas super escalares são conhecidas por possuírem mais de um pipeline em paralelo em sua arquitetura.**
- **Isso gera problemas de acesso aos recursos comuns, que precisam ser duplicados em sua maioria.**
- **Os barramentos devem largura suficiente para a busca de mais de uma instrução por ciclo.**
- **O uso de caches separadas para dados e instruções é uma solução para permitir leitura e escrita simultâneas à memória.**
- **É comum termos várias portas de leitura e de escrita no banco de registradores.**

Arquiteturas Superescalares

- **Devido a problemas de dependências de dados e controle pode acontecer que em alguns ciclos não haja instruções prontas para serem executadas.**
- **Esquemas mais simples de despacho, simplesmente congelam o *pipeline* nesse caso.**
- **Esquemas mais sofisticados continuam a busca e tentam executar as instruções fora-de-ordem, com mecanismos que posteriormente permitam a retirada dessas instruções, e atualização do estado da máquina, na mesma ordem estabelecida pelo programa objeto.**

Arquiteturas VLIW

- **VLIW** é acrônimo para **“Very Long Instruction Word”**. São máquinas que exploram o paralelismo no nível das instruções.
- Várias operações são executadas em paralelo em diferentes unidades funcionais, tais como em máquinas superescalares, a diferença está no controle do despacho e na terminação das operações.
- Nas máquinas superescalares as dependências são resolvidas em tempo de execução por um hardware dedicado, nas máquinas VLIW as dependências são resolvidas em tempo de compilação pelo compilador.

Arquiteturas VLIW

- Em uma máquina VLIW, várias operações são codificadas em uma mesma instrução. A palavra de instrução é bastante longa, podendo conter várias operações (que operam sobre vários operandos) independentes.
- A posição de cada operação dentro da palavra VLIW determina a unidade funcional que será usada. Deste modo, o hardware de despacho é simples.
- Em uma máquina VLIW o compilador é responsável por determinar as operações que serão executadas em paralelo.

Arquiteturas VLIW

- Operações que possam ser executadas em paralelo são atribuídas à uma mesma palavra de instrução.
- Dentro de um mesmo bloco básico (seqüencial) as instruções podem ser escalonadas com base na dependência de dados.
- Já o escalonamento que ultrapasse as fronteiras entre os blocos básicos exige técnicas mais elaboradas como “trace scheduling” e “list scheduling”.

Arquiteturas SMT

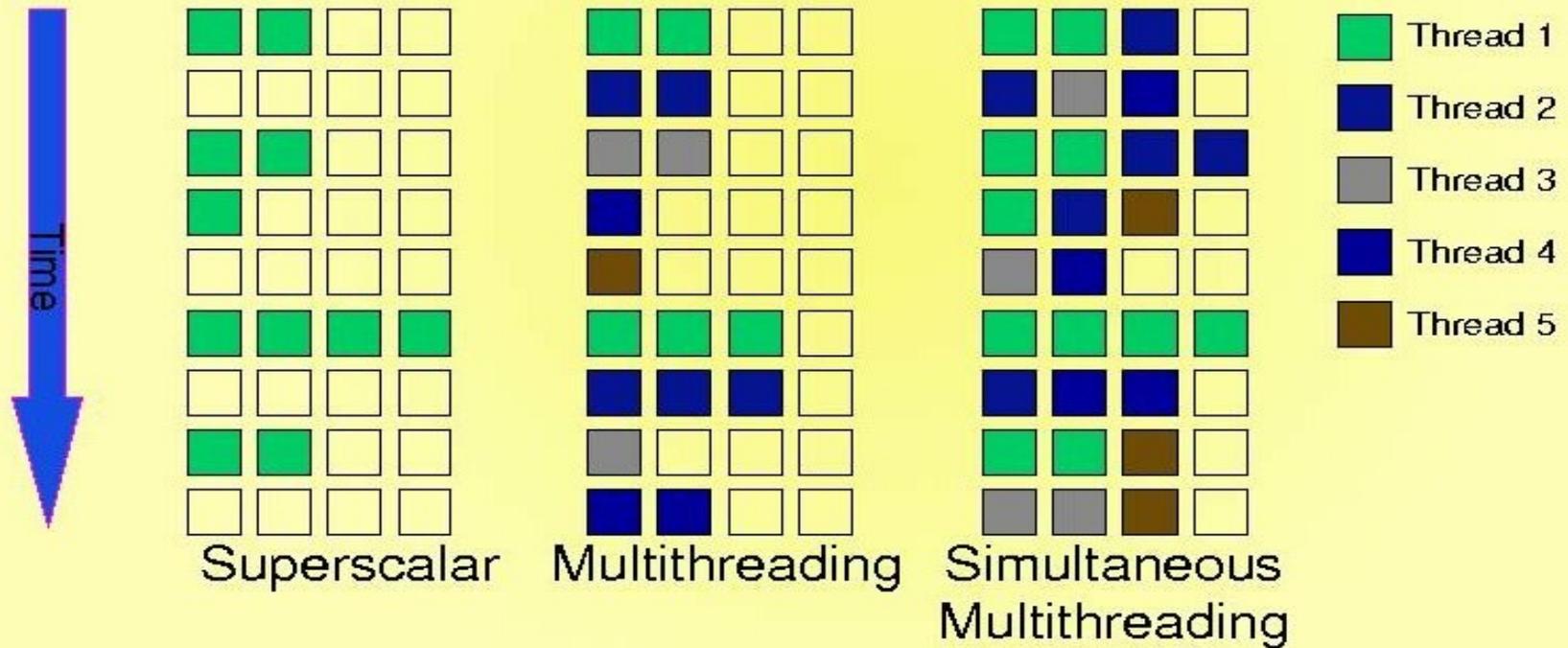
- SMT é uma técnica que permite múltiplas *threads* despacharem múltiplas instruções a cada ciclo para unidades funcionais de um processador superescalar.
- SMT combina a capacidade de despacho de múltiplas instruções das arquiteturas superescalares, com a habilidade de esconder latência das arquiteturas *multithreading*.
- A cada instante de tempo instruções de diferentes *threads* podem estar sendo executadas simultaneamente.
- Busca reduzir o número de *slots* de despacho não ocupados a cada ciclo (elevado em arquiteturas *multithreading*) e o número de ciclos em que nenhuma instrução é despachada (elevado em arquiteturas superescalares).

Arquiteturas SMT

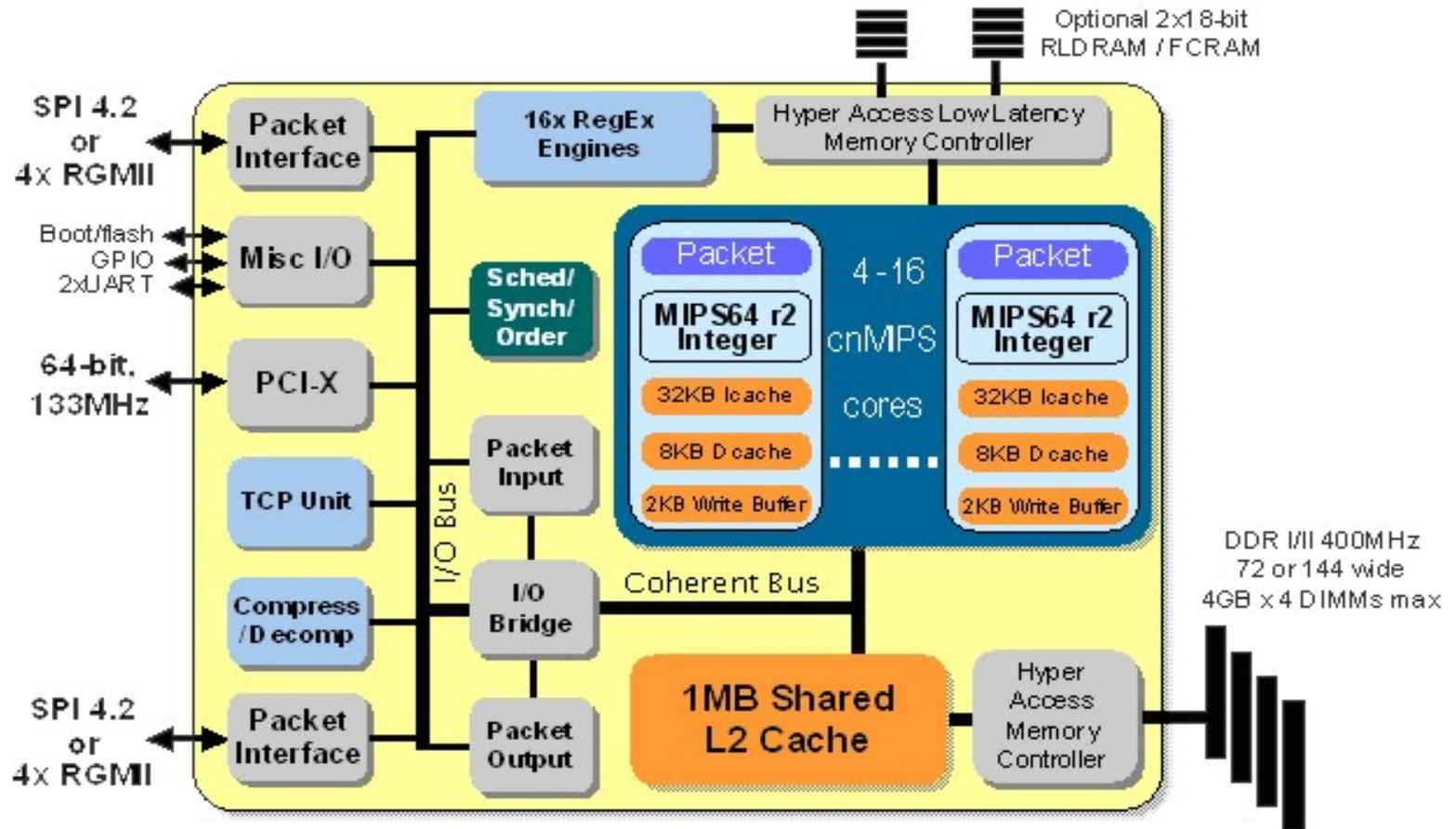
- As arquiteturas SMT podem ser de três tipos básicos (Tullsen, ISCA95):
 - Full Simultaneous issue: modelo ideal, mas de difícil realização, onde não há restrição sobre o número de instruções de cada thread que pode ser despachada a cada ciclo.
 - Limited Simultaneous Issue: apenas um número limitado (1 a 4 tipicamente) de instruções de cada thread pode ser despachado a cada ciclo.
 - Limited Connection: restringem o número de unidades funcionais de cada tipo que podem estar conectadas a cada contexto.

Arquiteturas SMT

Simultaneous multithreading



Arquiteturas SMT



Arquiteturas Multicore

- **Vários processadores encapsulados em uma mesma pastilha, podendo compartilhar ou não a cache de nível 2.**
- **Normalmente são processadores idênticos, tanto em termos de arquitetura quanto em velocidade de processamento.**
- **As tarefas podem ser executadas indistintamente em cada um dos processadores.**
- **O sistema operacional é responsável pelo escalonamento das tarefas (processos e/ou threads) para cada um dos processadores.**
- **É uma alternativa muito adotada recentemente devido aos ganhos de consumo de energia que podem ser obtidos, sem perda de desempenho quando há um número adequado de tarefas para serem executadas em paralelo.**