

Arquitetura e Organização de Computadores - 5cop090

Paralelismo em nível de instruções e processadores superescalares

Paralelismo em nível de instruções e processadores superescalares

Objetivos do módulo

- Explicar a diferença entre abordagens superescalares e de superpipelines.
- Definir o paralelismo em nível de instrução.
- Discutir dependências e conflitos de recursos como limitações ao paralelismo em nível de instrução.

Paralelismo em nível de instruções e processadores superescalares

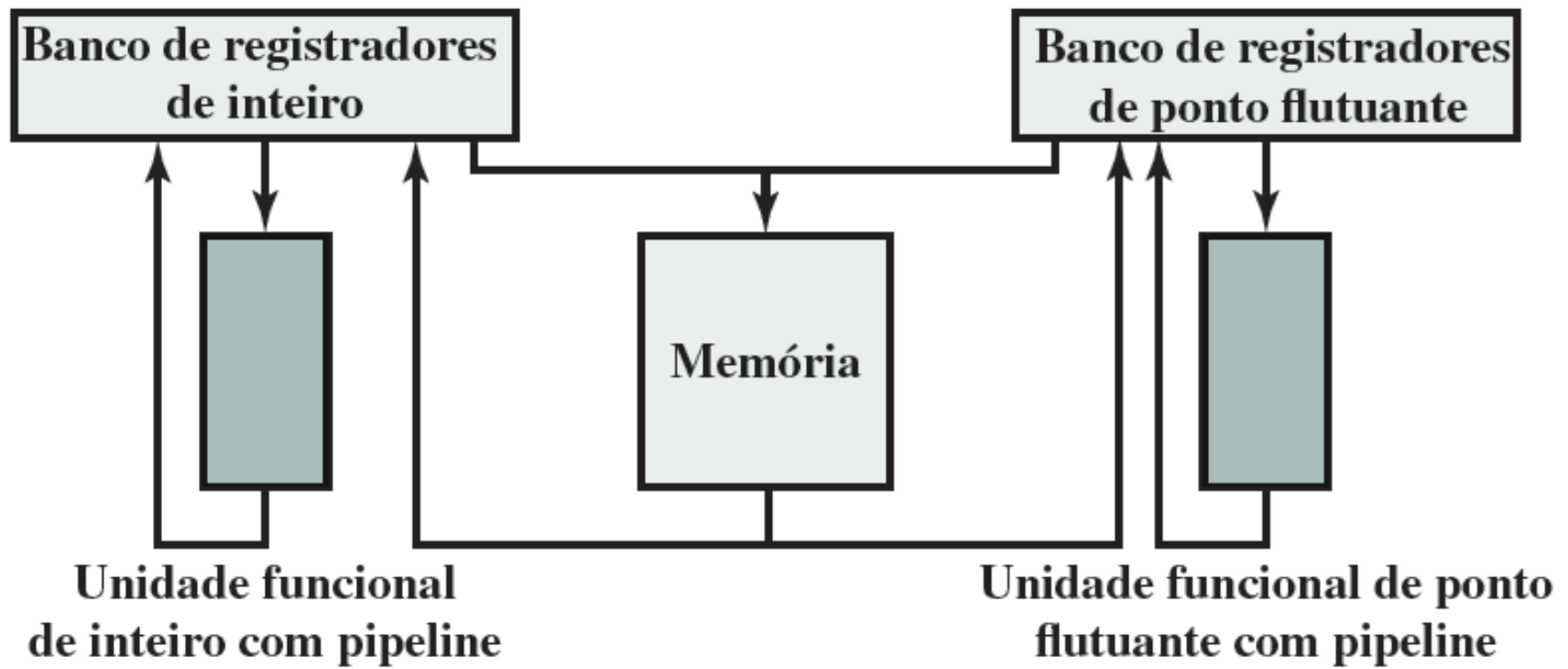
Visão geral

- A essência da abordagem superescalar é a habilidade de executar instruções **independente e concorrentemente** em pipelines diferentes.
- O conceito pode ser explorado permitindo que **as instruções sejam executadas em uma ordem diferente da do programa**.
- As figuras que seguem comparam, em termos gerais, as abordagens **escalares e superescalares**.
- Muitos pesquisadores investigaram processadores do tipo superescalar e suas pesquisas indicam que algum grau de melhoria de desempenho é possível.

Paralelismo em nível de instruções e processadores superescalares

Visão geral

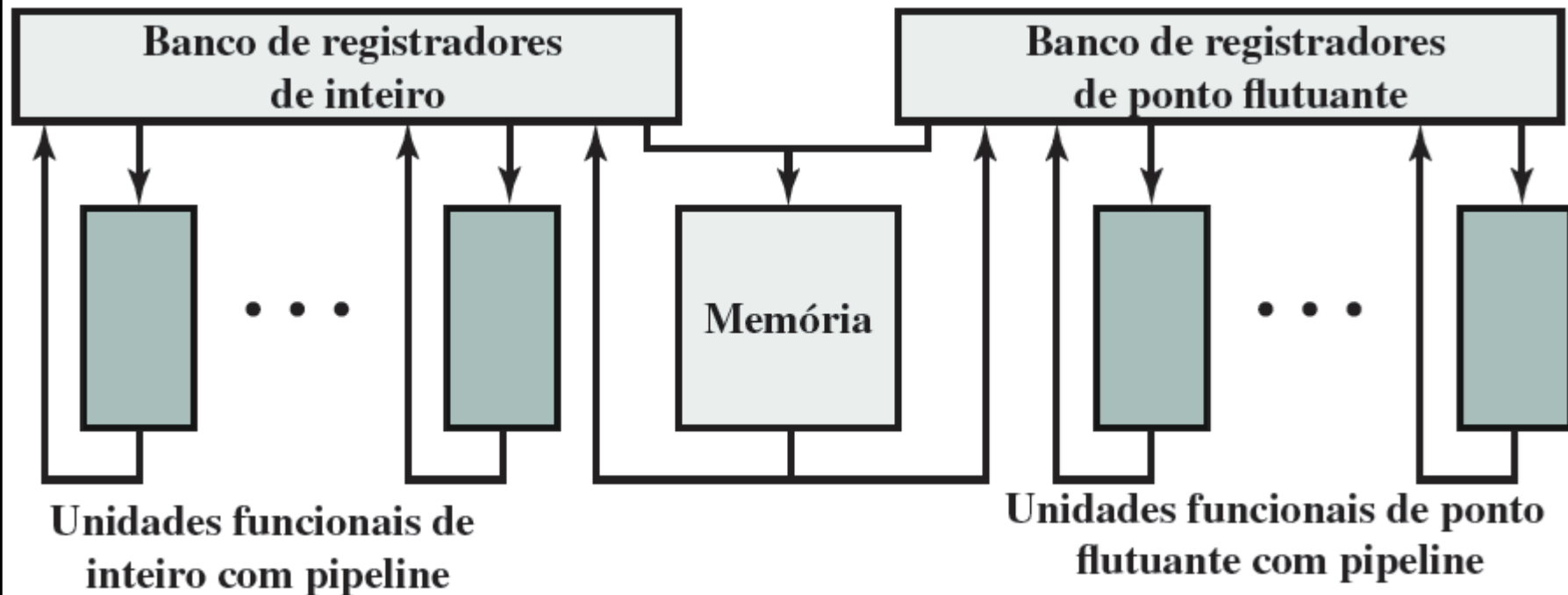
- Organização escalar:



Paralelismo em nível de instruções e processadores superescalares

Visão geral

- Organização superescalar:



Paralelismo em nível de instruções e processadores superescalares

Visão geral

- A abordagem superescalar depende da habilidade de executar **múltiplas instruções em paralelo**.
- O termo **paralelismo em nível de instruções** refere-se ao grau em que, em média, **as instruções de um programa podem ser executadas em paralelo**.
- Uma combinação de otimização **baseada em compilador e técnicas de hardware** pode ser usada para maximizar o paralelismo em nível de instruções.

Paralelismo em nível de instruções e processadores superescalares

Aspectos de projeto

- O **paralelismo em nível de instrução** existe quando as instruções de uma sequência são independentes e, assim, podem ser executadas em paralelo por sobreposição.
- Como um exemplo do conceito de paralelismo, considere dois pedaços de código a seguir (JOUPI, 1989b):

Load R1 ← R2	Add R3 ← R3, "1"
Add R3 ← R3, "1"	Add R4 ← R3, R2
Add R4 ← R4, R2	Store [R4] ← R0

Paralelismo em nível de instruções e processadores superescalares

Aspectos de projeto

- O **paralelismo de máquina** é uma medida da habilidade do processador para obter vantagem do paralelismo em nível de instruções.
- Ele é determinado pelo número de instruções que podem ser obtidas e executadas ao mesmo tempo (o número de pipelines paralelos) e pela velocidade e sofisticação dos mecanismos que o processador usa para localizar instruções independentes.
- O paralelismo em nível de instruções e o de máquina são fatores importantes para melhorar o desempenho.

Paralelismo em nível de instruções e processadores superescalares

Política sobre emissão de instruções

- Em geral, podemos dizer que a emissão da instrução ocorre quando a instrução é movida do estágio de decodificação para o primeiro estágio de execução do pipeline.
- Essencialmente, o processador está tentando olhar para a frente do ponto atual de execução para localizar instruções que podem ser trazidas para o pipeline e executadas.
- Três tipos de ordenação são importantes nessa consideração:
 1. A ordem em que as instruções são lidas.
 2. A ordem em que as instruções são executadas.

Paralelismo em nível de instruções e processadores superescalares

Política sobre emissão de instruções

3. A ordem em que as instruções atualizam o conteúdo dos registradores e as posições de memória.
 - Quanto mais sofisticado for o processador, menos ele estará sujeito a esta estrita relação entre essas ordens.
 - Em termos gerais, podemos agrupar políticas de emissão de instruções superescalares nas seguintes categorias:
 1. Emissão em ordem com conclusão em ordem.
 2. Emissão em ordem com conclusão fora de ordem.
 3. Emissão fora de ordem com conclusão fora de ordem.

Paralelismo em nível de instruções e processadores superescalares

Renomeação de registradores

- Uma maneira de enfrentar os conflitos de armazenamento é baseada em uma solução tradicional de conflitos de recursos: duplicação de recursos.
- Nesse contexto, a técnica é conhecida como renomeação de registradores.
- Basicamente, registradores são alocados dinamicamente pelo hardware do processador e são associados com os valores usados pelas instruções em vários pontos do tempo.
- Quando um novo valor de registrador é criado, um novo registrador é alocado para esse valor.

Paralelismo em nível de instruções e processadores superescalares

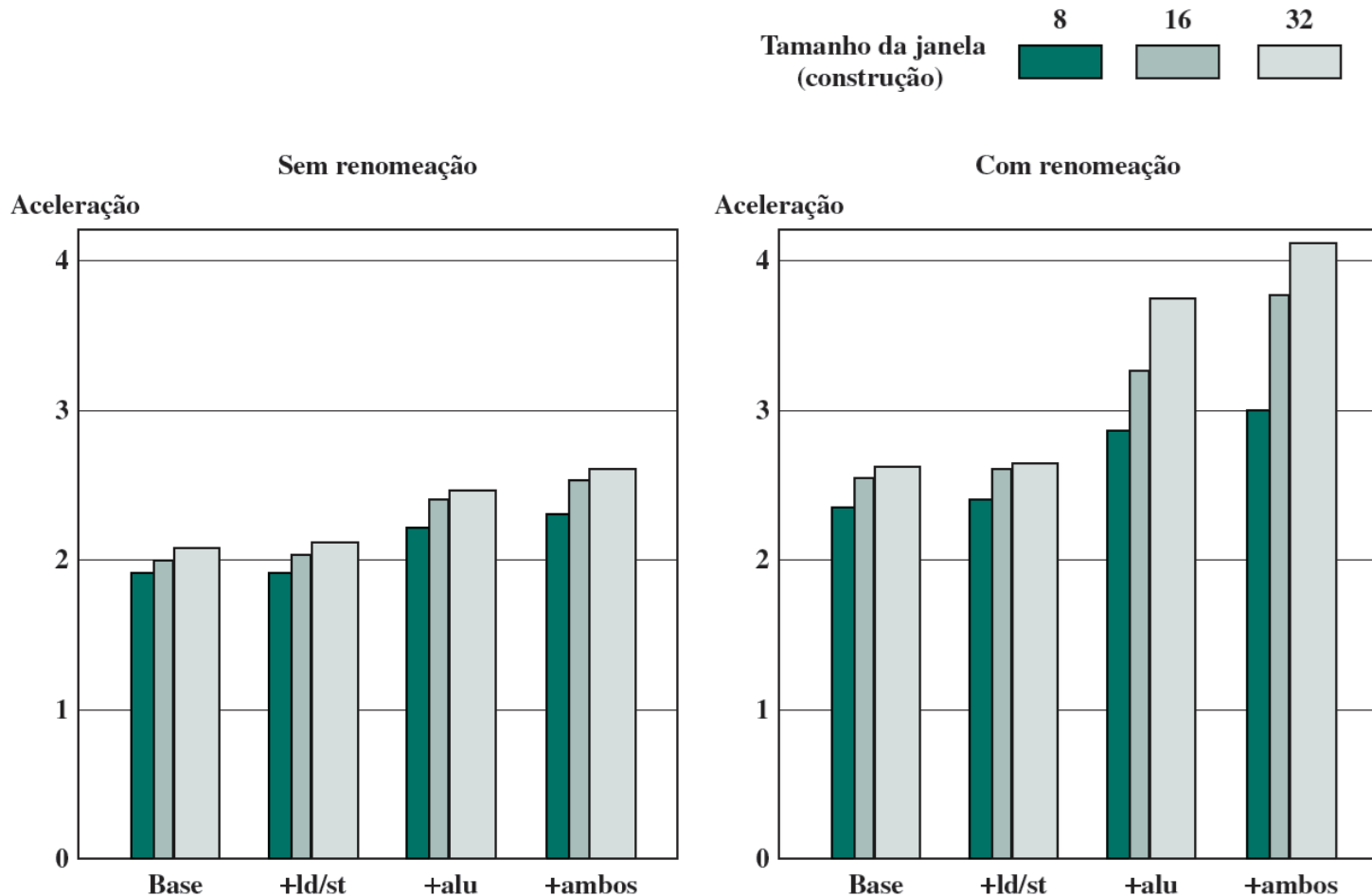
Renomeação de registradores

- As instruções subsequentes que acessam esse valor como operando de origem nesse registrador têm de passar pelo processo de renomeação:
- as referências de registradores nessas instruções precisam ser revisadas para que se refiram ao registrador que contém o valor necessário.
- Desse modo, a mesma referência do registrador original em várias instruções diferentes pode se referir a registradores reais diferentes, se valores diferentes são pretendidos.

Paralelismo em nível de instruções e processadores superescalares

Paralelismo de máquina

- Acelerações de várias organizações de máquinas sem dependências procedurais:



Paralelismo em nível de instruções e processadores superescalares

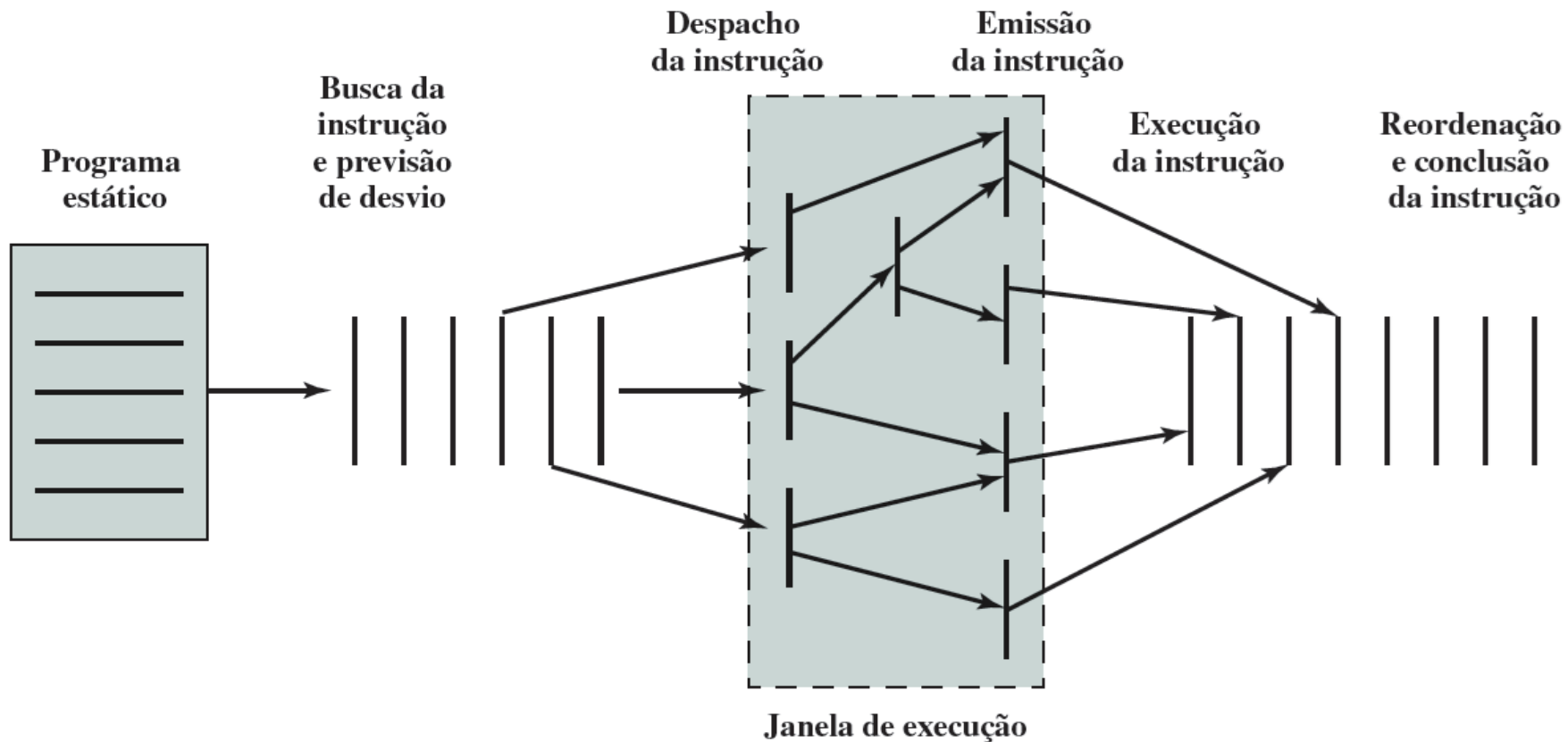
Previsão de desvio

- Com a chegada das máquinas RISC, a estratégia de desvio atrasado (*delayed branch*) foi explorada.
- Com o desenvolvimento de máquinas superescalares, a estratégia de desvio atrasado tem menos apelo.
- O motivo é que múltiplas instruções precisam ser executadas no slot de atraso (*delay slot*), trazendo vários problemas relacionados com as dependências das instruções.
- Desse modo, as máquinas superescalares retornaram às técnicas de previsão de desvios pré-RISC.

Paralelismo em nível de instruções e processadores superescalares

Execução superescalar

- Ilustração conceitual de processamento superescalar:



Paralelismo em nível de instruções e processadores superescalares

Execução superescalar

- O programa a ser executado consiste em uma sequência linear de instruções.
- Esse é o programa estático conforme escrito pelo programador ou gerado pelo compilador.
- O processo de obter instrução, o que inclui a previsão de desvio, é usado para formar um fluxo dinâmico de instruções.
- Esse fluxo é examinado para dependências e o processador pode remover as dependências artificiais.
- Ele então despacha as instruções para uma janela de execução.

Paralelismo em nível de instruções e processadores superescalares

Execução superescalar

- Nessa janela, as instruções não formam mais um fluxo sequencial, mas são estruturadas de acordo com suas dependências verdadeiras de dados.
- O processador efetua o estágio de execução de cada instrução numa ordem determinada pelas dependências verdadeiras de dados e pela disponibilidade de recursos de hardware.
- Finalmente, as instruções são conceitualmente colocadas de volta na ordem sequencial e seus resultados são reordenados.
- O passo final mencionado no parágrafo anterior é conhecido como concluir, ou retirar, a instrução.