

# Arquitetura e Organização de Computadores

Introdução ao Pipeline de Instruções

**Professor: André Luiz Tinassi D'Amato**  
**Disciplina: Arquitetura de Computadores**

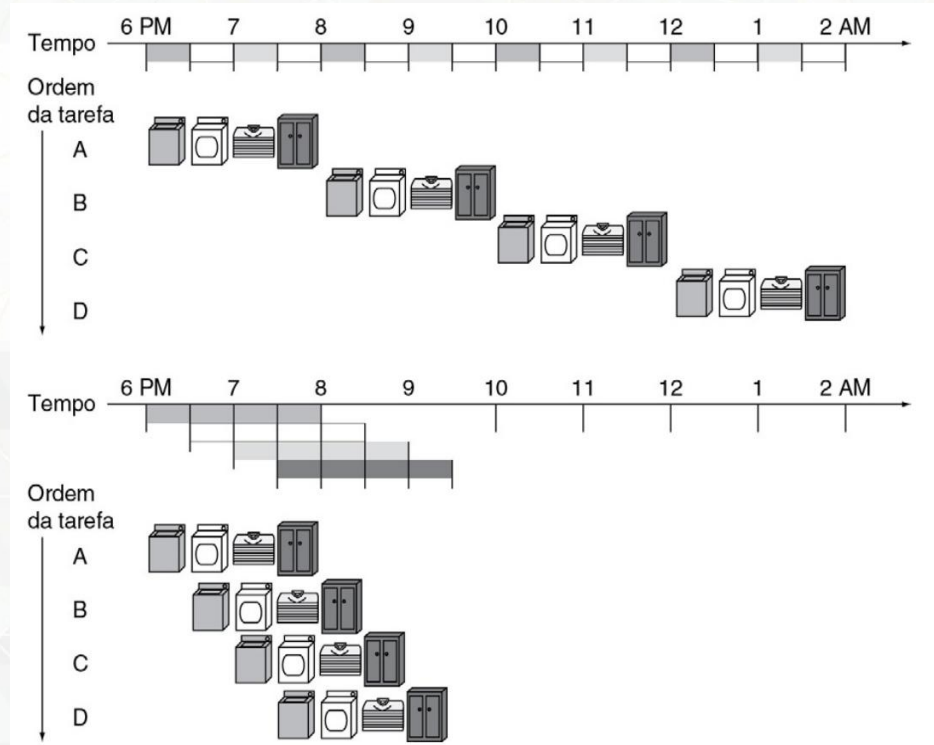
# Conceitos

- Arquiteturas monociclo
  - Desempenho do ciclo de instruções limitado pela sequência de operações
  - Solução: Buscar paralelismo na execução de instruções
  - Pipeline de instruções
    - Técnica que explora o paralelismo entre as instruções em um fluxo de instruções sequenciais
    - Novas instruções admitidas, antes que instruções previamente aceitas tenham terminado



# Visão geral de pipelining

- Colocar a trouxa de roupa suja na lavadora
- Quando a lavadora terminar, colocar a trouxa de roupa molhada na secadora (se houver)
- Quando a secadora terminar, colocar a trouxa de roupa seca na mesa e passar
- Quando terminar de passar, pedir ao seu colega de quarto para guardar as roupa



# Pipeline MIPS

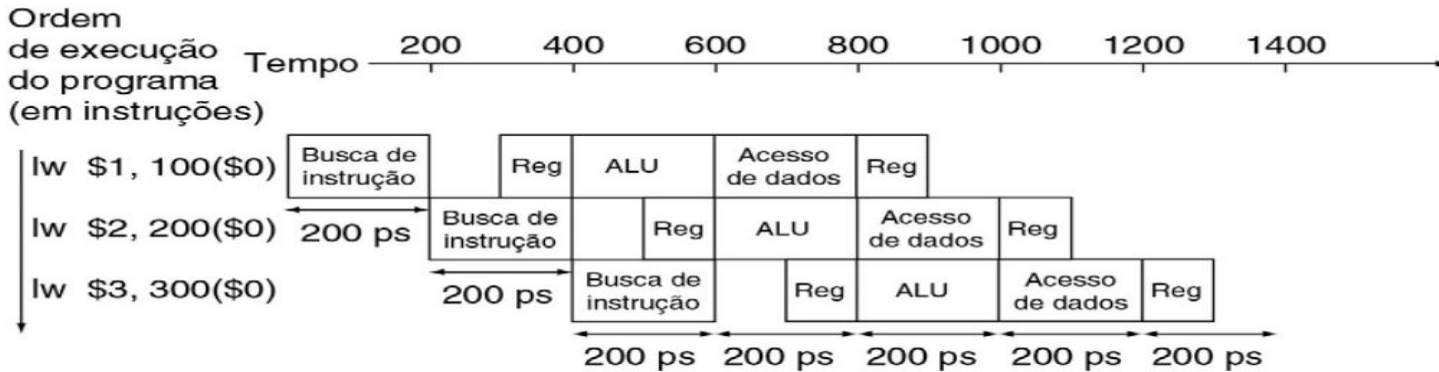
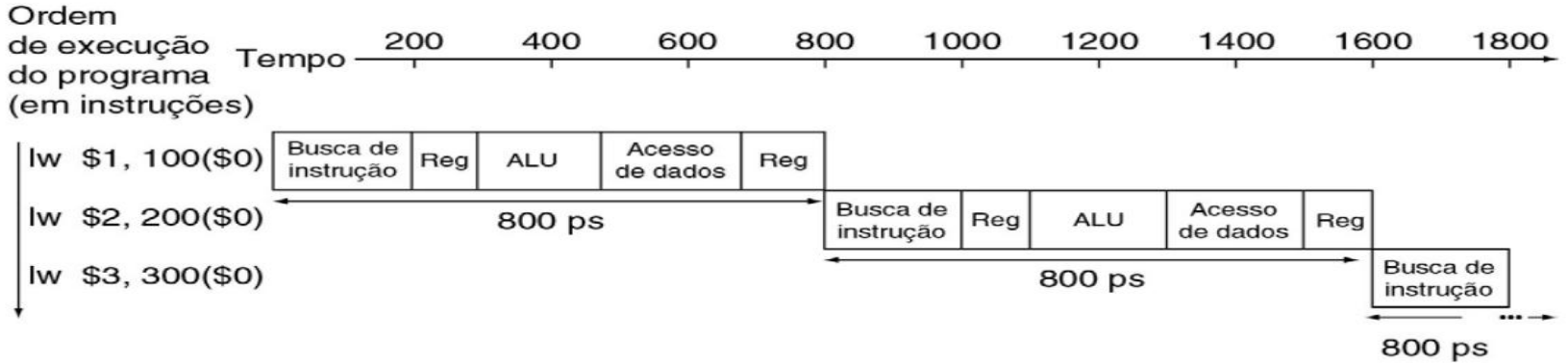
- IF: Buscar instrução da memória
- ID: Ler registradores enquanto a instrução é decodificada. O formato das instruções MIPS permite que a leitura e a decodificação ocorram simultaneamente
- EX: Executar a operação ou calcular um endereço
- MEM: Acessar um operando na memória de dados
- WB: Escrever o resultado em um registrador



# Pipeline MIPS

Classe de instrução	Busca de instrução	Leitura do registrador	Operação ALU	Acesso de dados	Escrita do registrador	Tempo Total
Load word (lw)	200 ps	100 ps	200 ps	200 ps	100 ps	800 ps
Store word (sw)	200 ps	100 ps	200 ps	200 ps		700 ps
R-format (add, sub, AND, OR, slt)	200 ps	100 ps	200 ps		100 ps	600 ps
Branch (beq)	200 ps	100 ps	200 ps			500 ps

# Desempenh



# Exercício Introdutório: Pipeline

Disponível no moodle



# Hazards de Pipeline (Conflitos)

- Conflitos podem ser
  - Estruturais
  - De dados
  - De controle

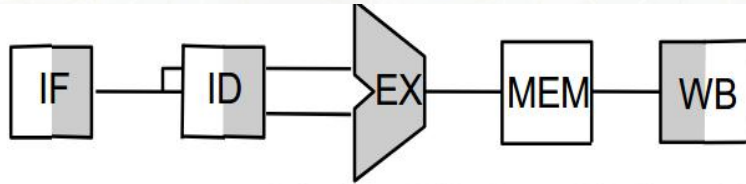


# Conflito de Dados

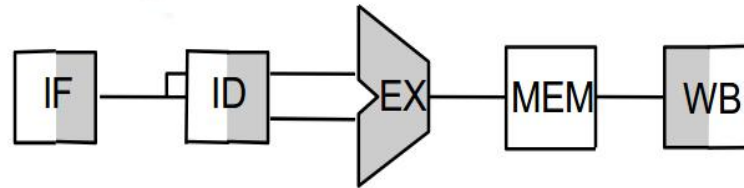
- Ocorre quando há dependência de dados
  - Possíveis soluções
    - Inserção de operações NOP (Bolhas)
    - Reordenação de código pelo compilador
      - Inserção de instruções independentes entre instruções dependentes
    - Inserção de curto circuito (Forwarding)

# Conflito de Dados

Add **r1**, r2, r3



Sub r4, **r1**, r5



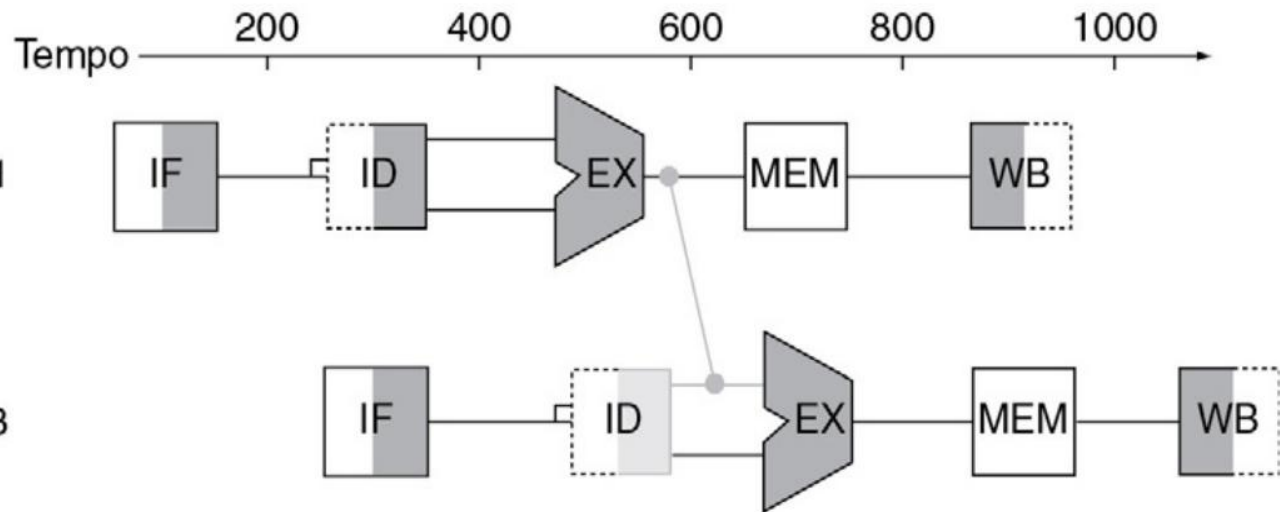
# Encaminhamento Forwarding

By-passing: Ir por uma rota alternativa

Ordem  
de execução  
do programa  
(em instruções)

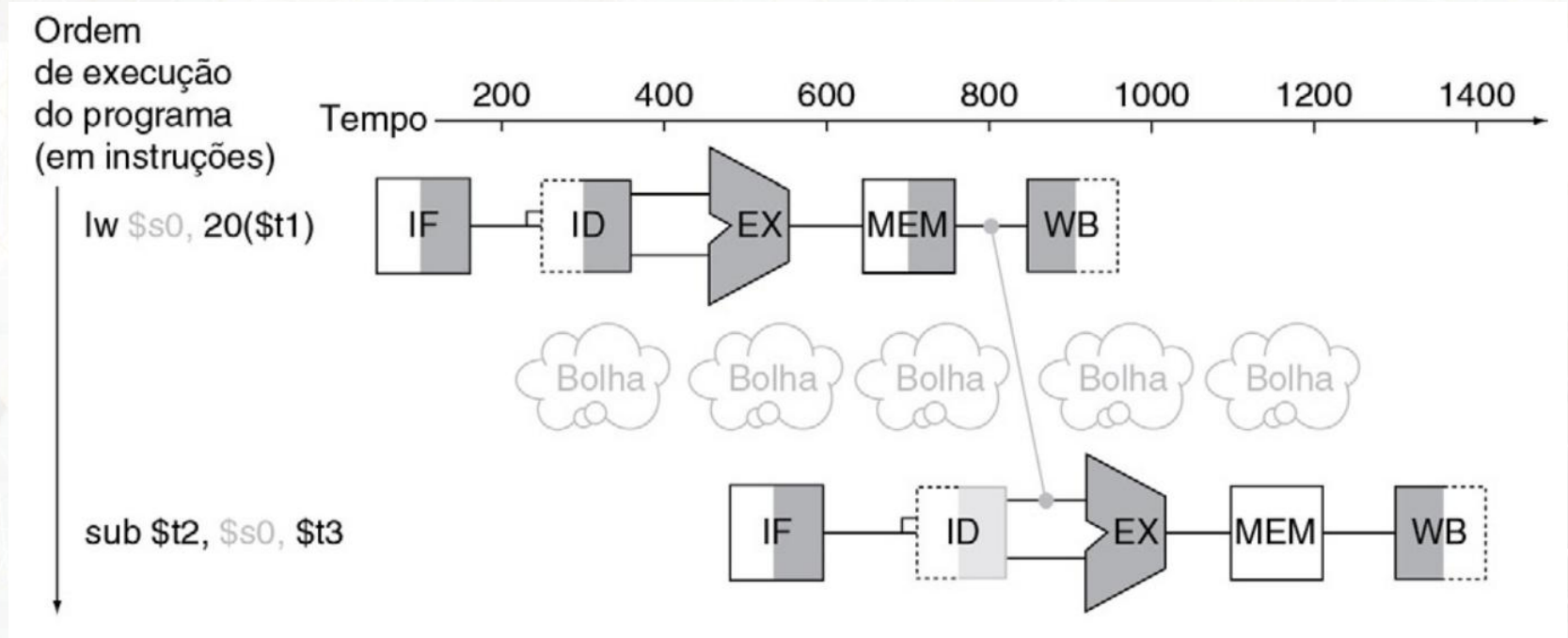
add \$s0, \$t0, \$t1

sub \$t2, \$s0, \$t3





# Forwarding (Caso LW)



# Exercício Introdutório: Hazard Dados

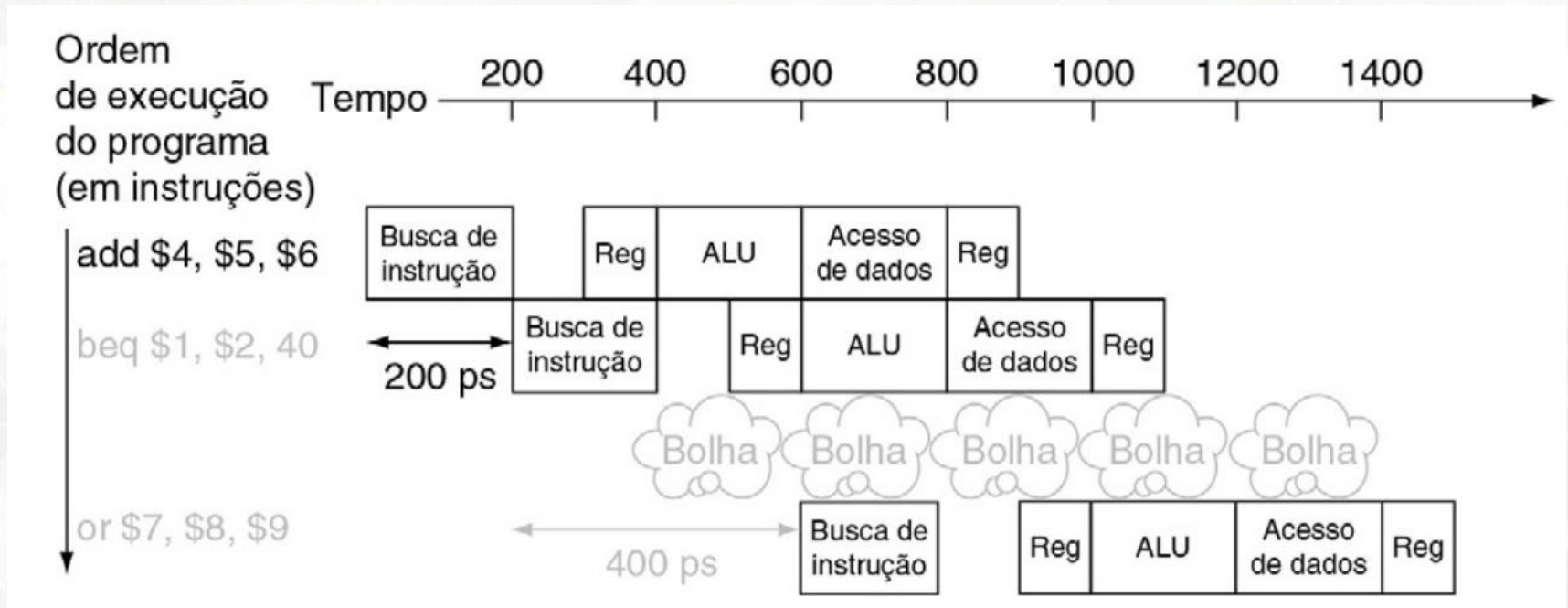
Disponível no moodle

# Conflito de Controle

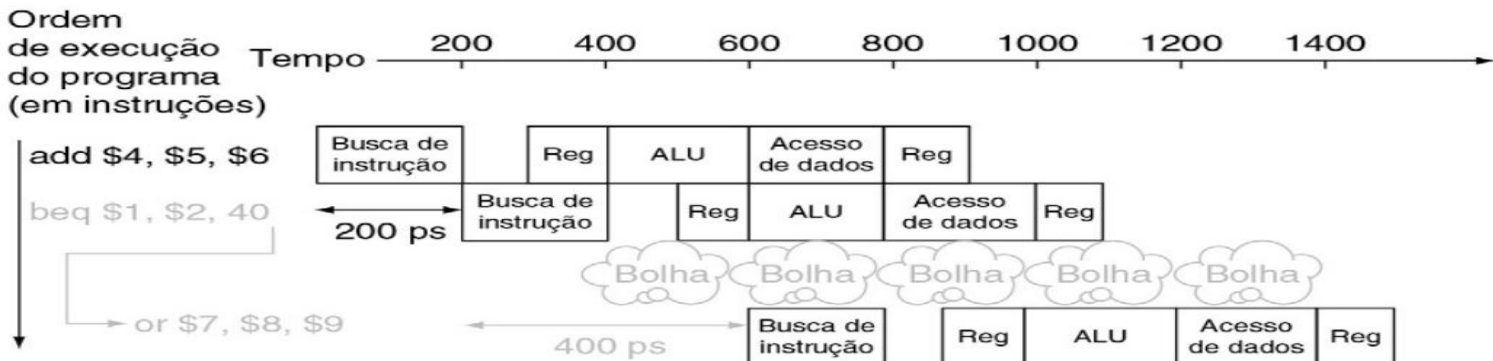
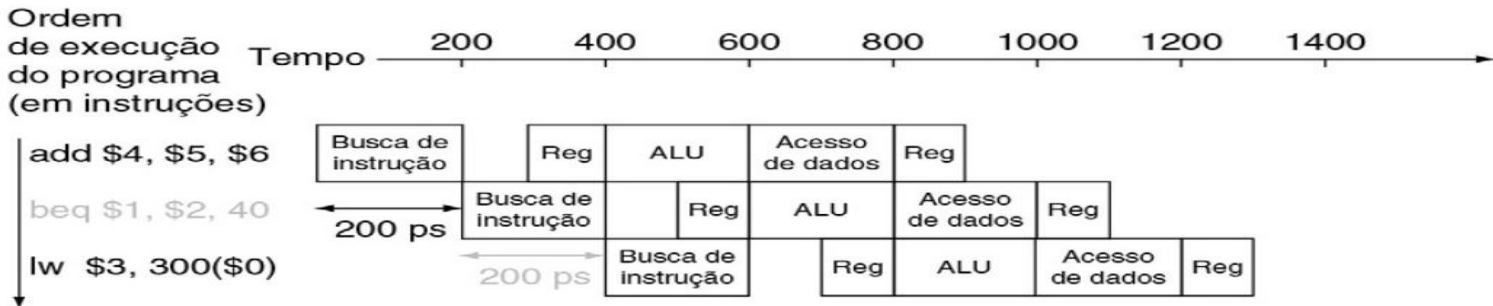
- Ocorre em instruções de desvio
  - Ex: j, jal, beq
  - Soluções
    - Bolhas
    - Previsão de desvio estático
    - Previsão de desvio dinâmico



# Conflito de Controle (Nop's)



# Previsão de Desvio Estático



Exemplo: Prever que o desvio nunca irá ocorrer



# Previsão de Desvio Dinâmico

- As decisões de predição podem ser alteradas durante a execução do programa
  - Implementação em hardware
  - Baseada em ocorrências passadas
  - Histórico (buffer de previsão)
  - Próxima aula...