

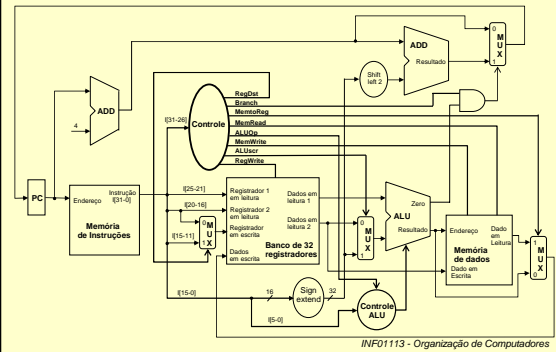
Organização de Computadores

Aula 06

Bloco operacional – versão multi-ciclo

INF01113 - Organização de Computadores

Todas as instruções usam todo HW?



INF01113 - Organização de Computadores

Máquina de ciclo único:

- Todas as instruções completam em um ciclo de relógio (CPI = 1)
- Algumas instruções usam mais HW que outras!
 - lw ocupa mais HW (5 módulos, vs. 4 para R-type e sw, 2 para beq)
- O relógio deve garantir a instrução mais longa \Rightarrow ineficiência!
 - Imagine que `mul` é incluída?

INF01113 - Organização de Computadores

Exemplo numérico

- Assuma 2ns para acesso as memórias, 1ns para leitura do banco de registradores, 2ns ALU e 1ns para escrita no banco.
- Período de relógio = 8 ns.
- Imaginemos que o instr. Mix seja de 24% loads, 12% stores, 44% R-format, 20% branches. Se tivéssemos um relógio variável, como seria o ganho?
 - Loads: 24%, levam 8ns; 1.92
 - stores: 12%, levam 5ns; 0.6
 - tipo-R: 44%, levam 6ns; 2.64
 - beq: 20%, levam 4ns; 0.8
- Se o relógio fosse variável, o ciclo médio de relógio seria 5.96ns, e a aceleração seria de 1.34 vezes
- Repita o problema caso incluíssemos um multiplicador, que aumentaria o tempo da ALU em 4ns.

INF01113 - Organização de Computadores

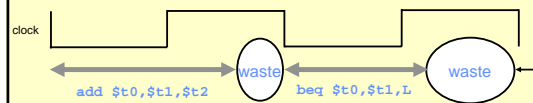
Multicycle Implementation (v.2)

- Want more efficient implementation
- Each step will take one clock cycle (not each instruction) [CPI > 1]
⇒ shorter clock cycle: cycle time **constrained by longest step, not longest instruction**
- simpler instructions take fewer cycles
⇒ higher overall performance
- complex control: [finite state machine](#)
- Versatile (can extend for new instructions: add3, swap, etc.)

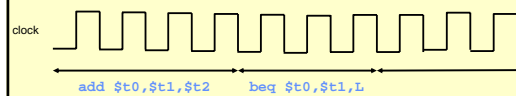
INF01113 - Organização de Computadores

Recap: Clocking: single-cycle vs. multicycle

Single-cycle Implementation



Multicycle Implementation



- Multicycle Implementation:
less waste=higher performance

INF01113 - Organização de Computadores

How fast can we run the clock?

- Depends on how much want done per clock cycle
 - Can do: [several](#) “inexpensive” datapath operations per clock
 - » simple gates (AND, OR, ...)
 - » single datapath registers (PC)
 - » sign extender, left shifter, multiplexor
 - PLUS: exactly [one](#) “expensive” datapath operation per clock
 - » ALU operation
 - » Register File access (2 reads, or 1 write)
 - » Memory access (read or write)

INF01113 - Organização de Computadores

Bloco operacional – versão multi-ciclo

1. Introdução
2. Ciclos das instruções
3. Bloco operacional completo
4. Execução das instruções
 - Busca de instrução
 - Decodificação de instrução
 - Instruções aritméticas
 - Instruções Load
 - Instruções Store
 - Instruções Branch
 - Instruções Jump
5. Cálculo de desempenho

INF01113 - Organização de Computadores

1. Introdução

- máquina mono-ciclo
 - todas as operações devem ser feitas em um só ciclo
 - duração do ciclo calculada pelo pior caso
 - leitura da instrução e acesso à memória no mesmo ciclo: duas memórias
 - cálculos de endereço e operações aritméticas no mesmo ciclo: três unidades funcionais (ALU, somadores)
- máquina multi-ciclo
 - vários ciclos por instrução
 - cada instrução pode ser executada num número diferente de ciclos
 - unidades funcionais podem ser reutilizadas em ciclos distintos
 - » pequeno acréscimo de multiplexadores e registradores
- compromisso no desempenho
 - CPI aumenta => desempenho cai
 - período do relógio diminui => desempenho sobe

INF01113 - Organização de Computadores

Observação histórica

- As primeiras máquinas que surgiram foram multiciclo. Por que?
- Há uma tendência de se aumentar o uso de máquinas ciclo-único: novas mudanças tecnológicas!

INF01113 - Organização de Computadores

Introdução

- quando é necessário inserir registradores?
 - quando valor é computado num ciclo e utilizado em outro ciclo
 - quando entradas de unidade funcional podem mudar antes que a saída seja salva em outro registrador ou memória
 - exemplo: Instruction Register
 - » memória vai mudar saída devido à atualização do PC e campos da instrução precisam se manter estáveis nas entradas do banco de registradores durante todos os ciclos
- instrução deve ser dividida em passos de duração similar

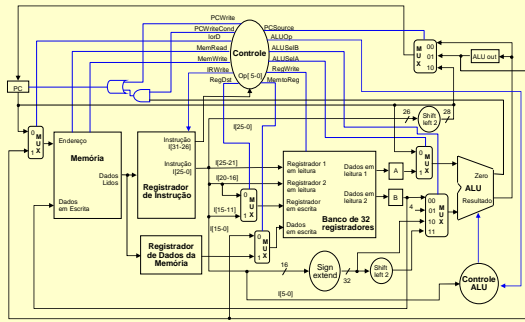
INF01113 - Organização de Computadores

2. Ciclos das instruções

1. Busca da instrução
2. Decodificação da instrução
 - Leitura dos registradores – mesmo que não sejam utilizados
 - Cálculo do endereço do branch – mesmo que instrução não seja branch
3. Execução da operação – instruções tipo R
 - Cálculo do endereço efetivo do operando – instruções load e store
 - Determinar se branch deve ser executado – instruções branch
4. Acesso à memória – instruções load e store
 - Escrita de registrador – instruções tipo R
5. Escrita de registrador – instruções load

INF01113 - Organização de Computadores

3. Bloco operacional completo



INFO1113 - Organização de Computadores

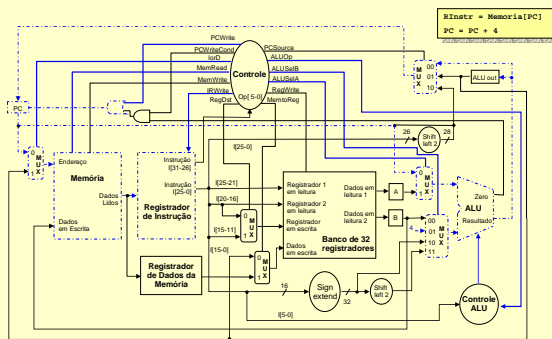
Comparando com BO mono-ciclo

- uma única memória para dados e instruções
- uma única unidade lógica e aritmética para todas as operações
 - operações das instruções tipo R
 - cálculo de $PC = PC + 4$
 - cálculo de endereço efetivo de memória: base + deslocamento
 - cálculo de endereço de desvio: $PC + \text{deslocamento}$
- novos registradores
 - Registrador de Instruções
 - Registrador de Dados da Memória (MDR)
 - A e B: guardam valores dos operandos do banco de registradores
 - ALU out: guarda valor da saída da ALU
- novos multiplexadores ou extensão dos já existentes

INFO1113 - Organização de Computadores

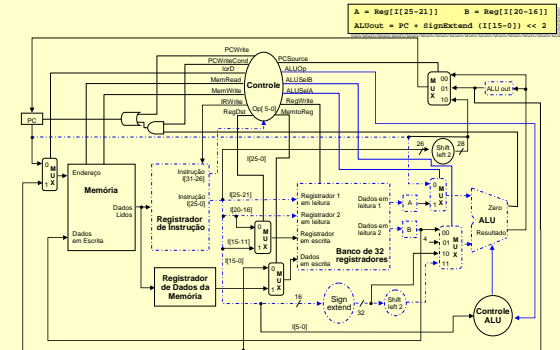
4. Execução das instruções

1º ciclo- Busca de instrução



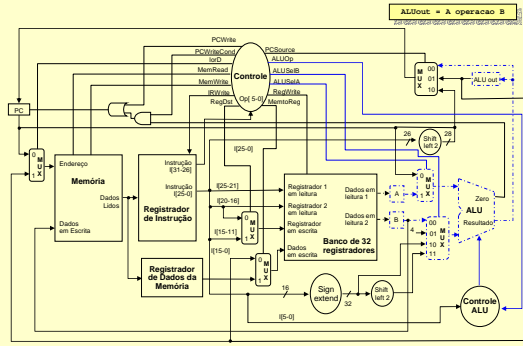
INFO1113 - Organização de Computadores

2º ciclo - Decodificação de instrução (+ leitura de registradores, + cálculo de end. branch)

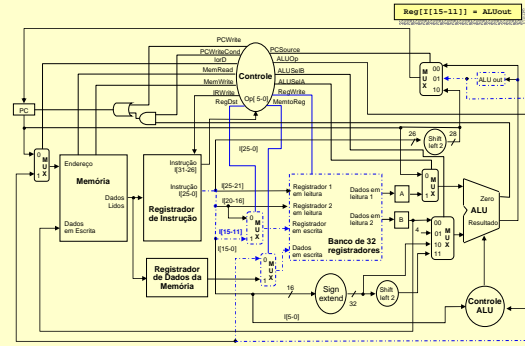


INFO1113 - Organização de Computadores

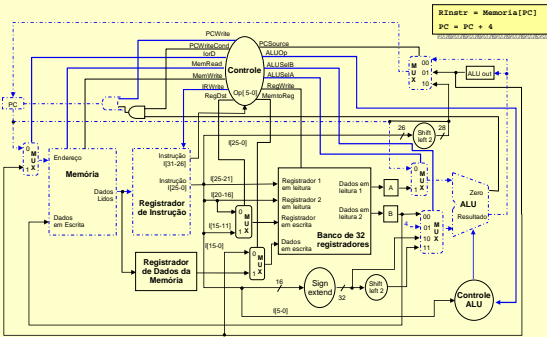
Instruções aritméticas – 3º ciclo



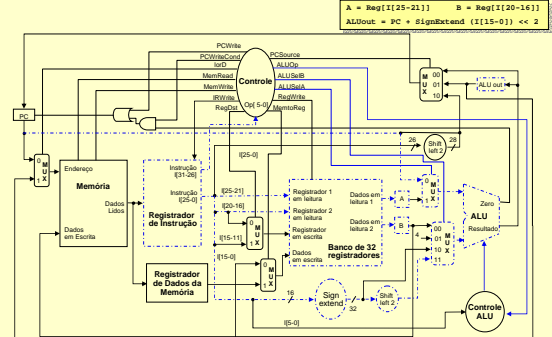
Instruções aritméticas – 4º ciclo

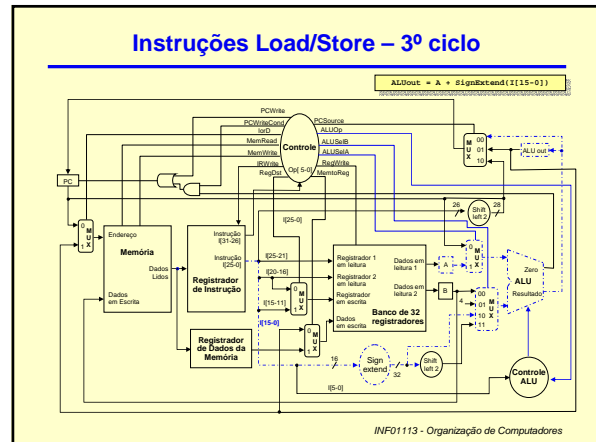
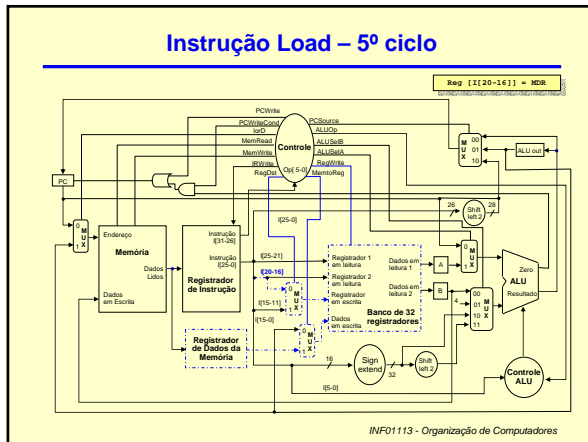
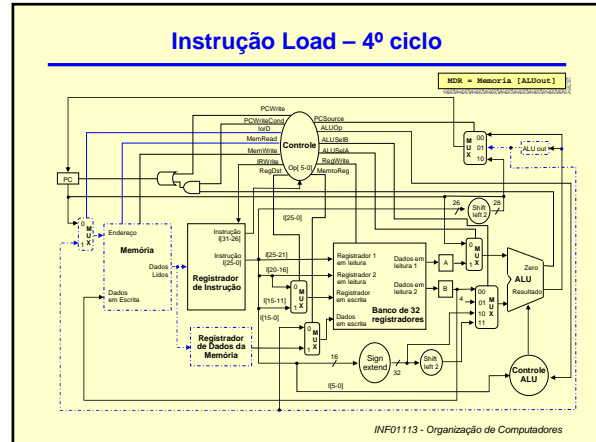
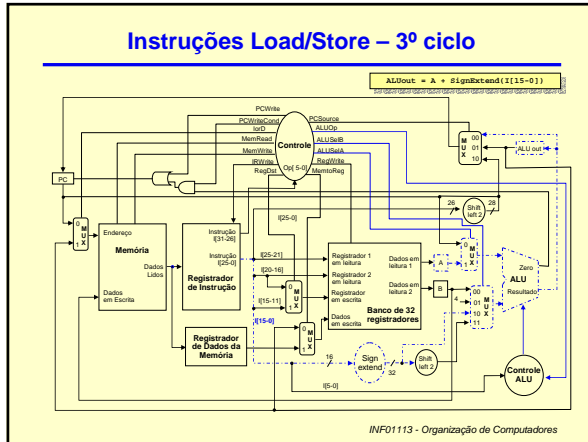


Instrução Load: 1º ciclo- Busca de instrução

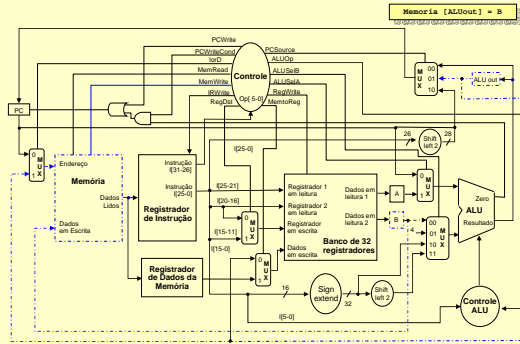


2º ciclo - Decodificação de instrução (+ leitura de registradores, + cálculo de end. branch)



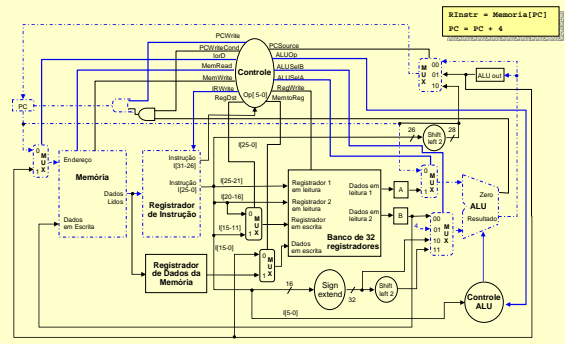


Instrução Store – 4º ciclo



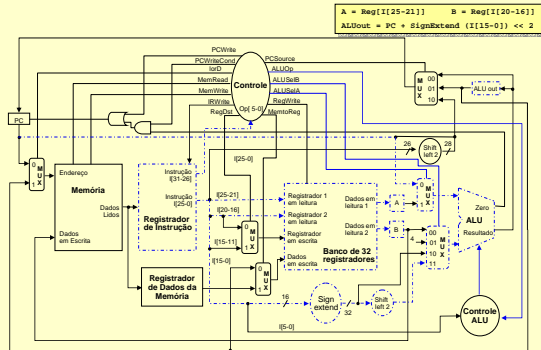
INF01113 - Organização de Computadores

Instrução Branch 1º ciclo- Busca de instrução



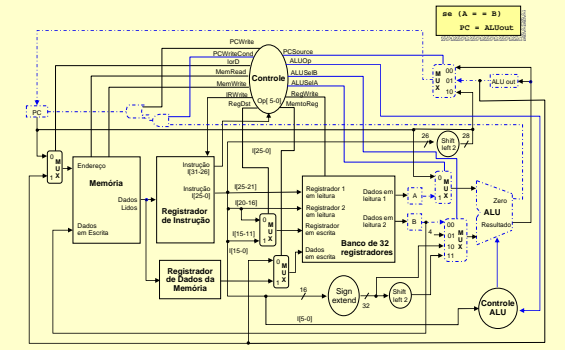
INF01113 - Organização de Computadores

2º ciclo - Decodificação de instrução (+ leitura de registradores, + cálculo de end. branch)



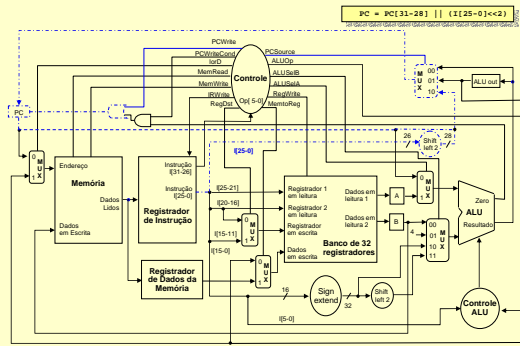
INF01113 - Organização de Computadores

Instrução Branch – 3º ciclo



INF01113 - Organização de Computadores

Instrução Jump – 3º ciclo



5. Cálculo de desempenho

- clock pode ter período de 1 ns (1 GHz), considerando ...
 - 1 ns para acessos à memória
 - 0.5 ns para acessos (escrita ou leitura) ao banco de registradores
 - 0.5 ns para operações na ALU
 - 0 ns para demais blocos (muxs, portas, ...)
- com estes valores, período da versão mono-ciclo seria de 3.5 ns
- clock da máquina multi-ciclo poderia ser duplicado (2 GHz) se acessos à memória fossem realizados em 2 ciclos de 0.5 ns cada
- CPI para cada tipo de instrução

	versão 1 GHz	versão 2 GHz
load:	5	7
store:	4	6
tipo R:	4	5
branch:	3	4
jump:	3	4

INFO1113 - Organização de Computadores

Cálculo de desempenho

- mix de instruções do compilador gcc
 - 22% loads, 11% stores, 49% tipo R, 16% branches, 2% jumps
- CPI médio na máquina multi-ciclo
 - $= 0.22 \times 5 + 0.11 \times 4 + 0.49 \times 4 + 0.18 \times 3 = 4.04$ na versão 1 GHz
 - $= 0.22 \times 7 + 0.11 \times 6 + 0.49 \times 5 + 0.18 \times 4 = 5.37$ na versão 2 GHz
- tempo total de execução do programa gcc no MIPS mono-ciclo
 - $= N \times \text{CPI} \times \text{período do clock}$
 - $= N \times 1 \times 3.5 \text{ ns} = 3.5 N \times 10^{-9}$
- tempo total de execução do programa gcc no MIPS multi-ciclo
 - $= N \times \text{CPI médio} \times \text{período do clock}$
 - $= N \times 4.04 \times 1 \text{ ns} = 4.04 N \times 10^{-9}$ na versão 1 GHz
 - $= N \times 5.37 \times 0.5 \text{ ns} = 2.685 N \times 10^{-9}$ na versão 2 GHz

INFO1113 - Organização de Computadores

Para os famintos por Saber

- O que deve ser alterado no datapath multiciclo para realizar a instrução LUI?
- O que deve ser alterado no datapath multiciclo para realizar a instrução de load com pós-incremento?

INFO1113 - Organização de Computadores

