



Resumo – Aula 6

Processador Multiciclo

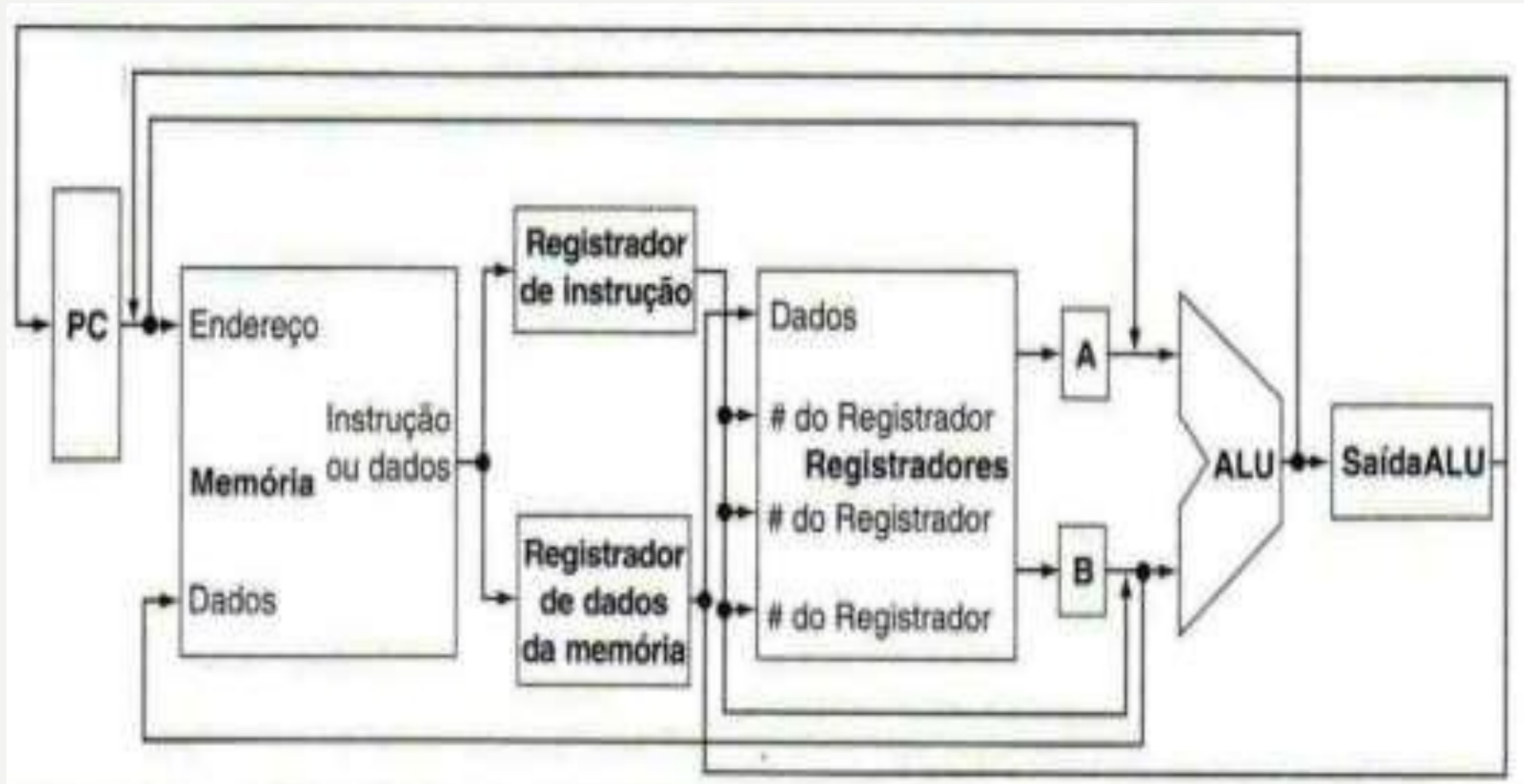
Prof. Sérgio Ronaldo

Sumário

- Uma Implementação Multiciclo (Patterson 3^o edição - cap. 5, seção 5.5).

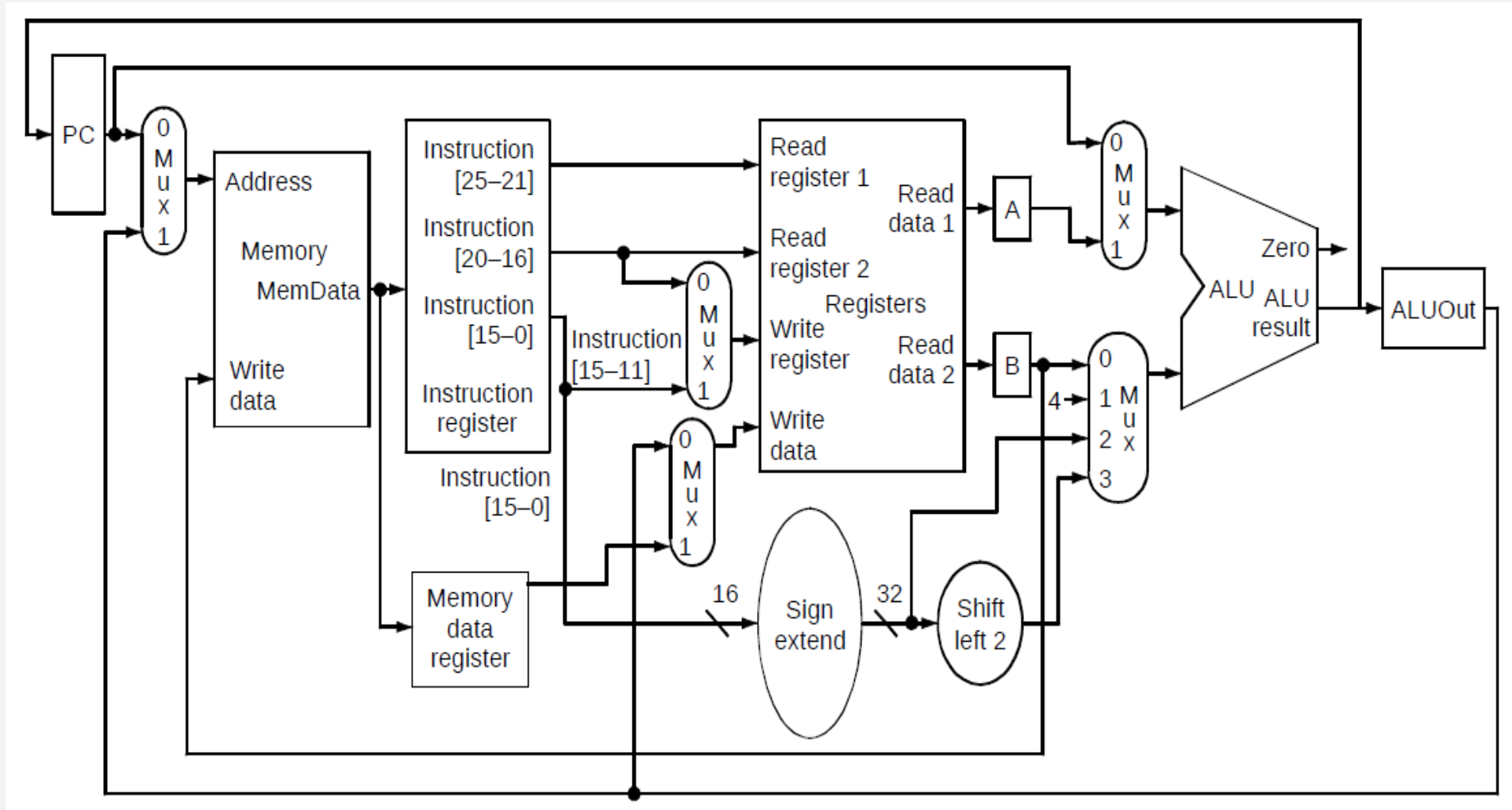
Principais unidades funcionais

- Visão de alto nível do caminho de dados do processador multiciclo.



Principais unidades funcionais

- Comparando com o caminho de dados para a versão de ciclo único poderemos ver as seguintes diferenças.



Implementação Multiciclo

- Comparando com o caminho de dados para a versão de ciclo único poderemos ver as seguintes diferenças.

- Uma única unidade de memória é usada para instruções e para dados.
- Existe uma única ALU, em vez de uma ALU e dois somadores.
- Um ou mais registradores são adicionados após cada unidade funcional para conter a saída dessa unidade até o valor ser usado em um ciclo de clock subsequente.

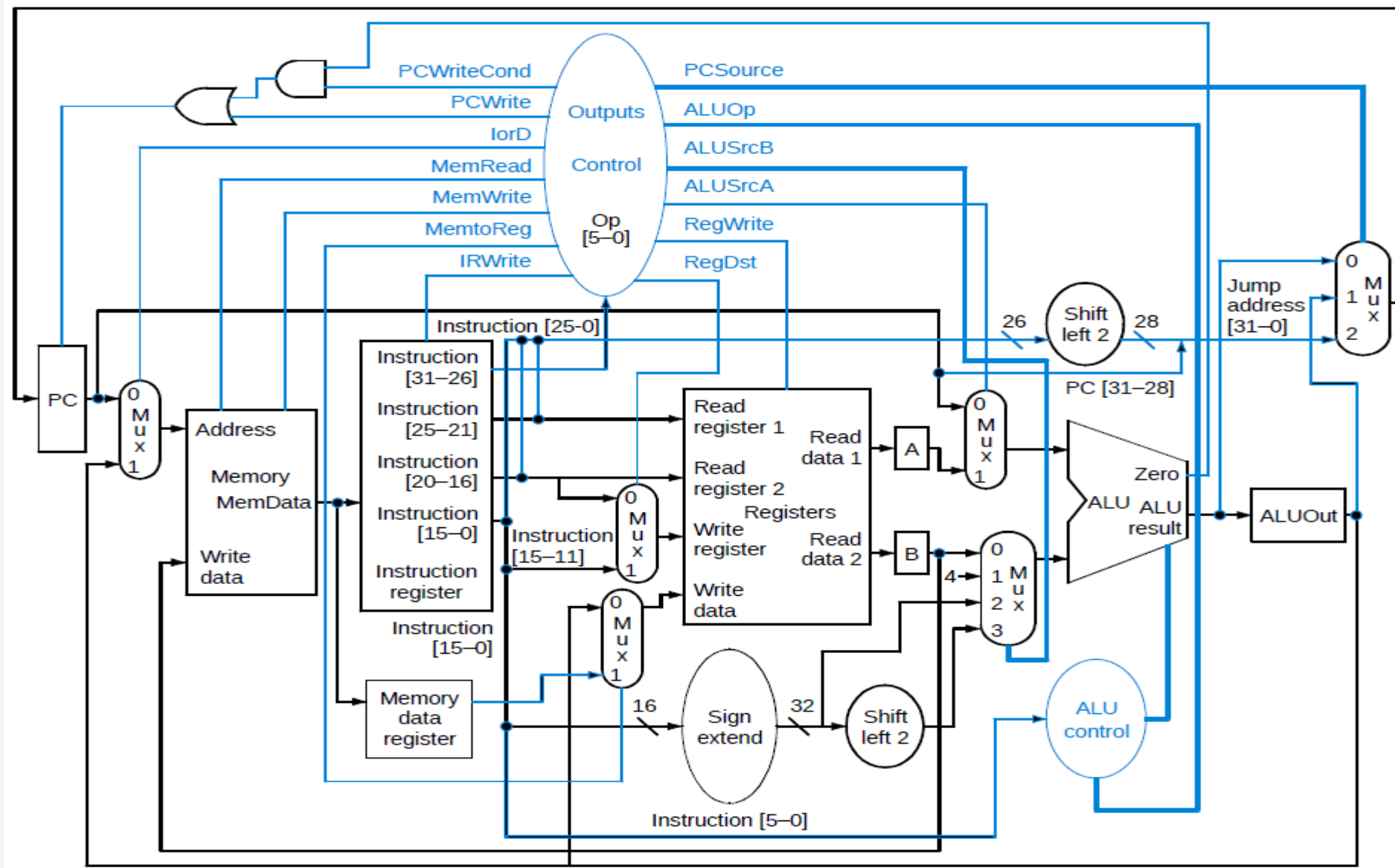
Implementação Multiciclo

Os seguintes registradores temporários são acrescentados para atender a esses requisitos:

- O registrador de instrução (IR) e o registrador de dados da memória (MDR) são incluídos para salvar a saída da memória para uma leitura de instrução e uma leitura de dados, respectivamente. Dois registradores separados são usados, já que, como ficará claro em breve, ambos os valores são necessários durante o mesmo ciclo de clock.
- Os registradores A e B são usados para conter os valores dos registradores operandos lidos do banco de registradores.
- O registrador SaídaALU contém a saída da ALU.

Todos os registradores exceto o IR contêm dados apenas entre um par de ciclos de clock adjacentes e, portanto, não precisarão de um sinal de controle de escrita. O IR precisa conter a instrução até o fim da execução dessa instrução e, então, exigirá um sinal de controle de escrita.

Caminho de dados para o Multiciclo



Dividindo a execução da instrução em ciclos de clock

- 1) Busca da instrução.
- 2) Decodificação da instrução e busca dos registradores.
- 3) Execução, cálculo do endereço de memória ou conclusão do desvio.
- 4) Acesso à memória ou conclusão de instrução do tipo R.
- 5) Etapa de conclusão da leitura da memória.

Obs: Cada etapa de execução leva 1 ciclo de clock.

Dividindo a execução da instrução em ciclos de clock

1º Passo: busca da instrução

$IR = \text{Memória}[PC]$

$PC = PC + 4$

- Envio do PC para a memória como endereço, realização da leitura e escrita da instrução no registrador de instruções.

Dividindo a execução da instrução em ciclos de clock

2º Passo: decodificação da instrução e busca do registrador

- a) Executa algumas operações que podem depois ser descartadas após a decodificação da instrução (ainda não se sabe qual instrução está no IR)
- b) São executadas para evitar a perda de tempo durante a execução (ler rs e rt)
- c) Carga de:
 - Registradores de entrada da ULA
 - Endereço de desvio condicional (salvo em UALSaída)
- d) $A = \text{Reg}[\text{IR}[25-21]];$
- e) $B = \text{Reg}[\text{IR}[20-16]];$
- f) $\text{UALSaída} = \text{PC} + \text{extensão de sinal } (\text{IR}[15-0] \ll 2);$

Dividindo a execução da instrução em ciclos de clock

3º Passo: Execução, cálculo do endereço de memória ou efetivação do desvio condicional

a) Referência à memória

$$\text{UALSaída} = A + \text{extensão de sinal IR}[15-0]$$

b) Instrução aritmética ou lógica (tipo R)

$$\text{UALSaída} = A \text{ op } B$$

c) Desvio condicional

$$\text{Se } (A == B) \text{ PC} = \text{UALSaída}$$

d) Desvio incondicional

$$\text{PC} = \text{PC} [31-28] \parallel (\text{IR}[25-0] \ll 2)$$

Dividindo a execução da instrução em ciclos de clock

4º Passo: Final da execução das instruções de acesso à memória e do tipo R

a) Referência à memória

MDR = Memória [UALSaída];

ou

Memória = [UALSaída] = B;

b) Instruções aritméticas ou lógicas (tipo R)

Reg[IR[15-11]] = UALSaída;

Dividindo a execução da instrução em ciclos de clock

5° Passo: Final de leitura da memória

- Load word:
 - $\text{Reg}[\text{IR}[20-16]] = \text{MDR}$