



Universidade Federal de Pelotas
Centro de Desenvolvimento Tecnológico
Bacharelado em Ciência da Computação
Engenharia de Computação

Arquitetura e Organização de Computadores I

Aula 4

**Arquitetura do MIPS: formatos de instrução,
registradores, modos de endereçamento.
MIPS monociclo: regime de clock, construção do
bloco operativo.**

Prof. Guilherme Corrêa
gcorrea@inf.ufpel.edu.br

Arquitetura do MIPS

► Registradores

- 32 registradores de propósito geral de 32 bits
 - \$0, \$1, ..., \$31
 - operações inteiras
 - endereçamento
- \$0 tem sempre valor 0

Arquitetura do MIPS

► Tipos de Dados

- Sempre 32 bits na parte teórica da disciplina;
- Outros tamanhos são suportados e serão usados na parte prática:
 - bytes
 - meias-palavras de 16 bits
 - palavras de 32 bits

Arquitetura do MIPS

► Modos de Endereçamento

- **Acessos à memória devem ser alinhados**
 - Dados de 32 bits precisam iniciar em endereços múltiplos de 4
- **Modo registrador**
 - Para instruções aritméticas e lógicas: dado está em registrador

Arquitetura do MIPS

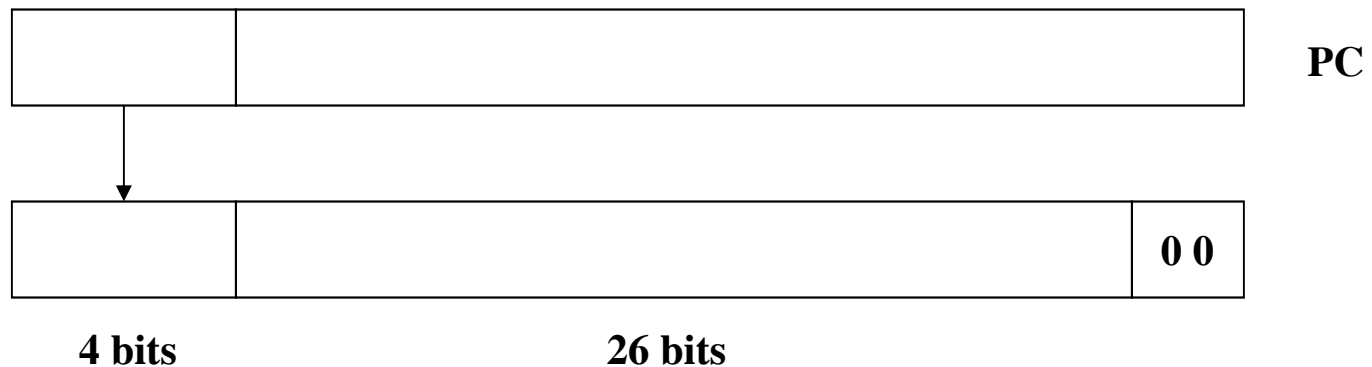
► Modos de Endereçamento

- **Modo base e deslocamento**
 - Para instruções load e store
 - Base é um registrador inteiro de 32 bits
 - Deslocamento de 16 bits contido na própria instrução
- **Modo relativo ao PC**
 - Para instruções de branch condicional
 - Endereço é a soma do PC com deslocamento contido na instrução
 - Deslocamento é dado em palavras e precisa ser multiplicado por 4

Arquitetura do MIPS

► Modos de Endereçamento

- **Modo absoluto**
 - Para instruções de desvio incondicional
 - Instrução tem campo com endereço de palavra com 26 bits
 - Endereço de byte obtido com dois bits menos significativos iguais a 0
 - 4 bits mais significativos obtidos do PC
 - Só permite desvios dentro de uma área de 256 Mbytes



Arquitetura do MIPS

► Instruções Suportadas na Parte Teórica da Disciplina

- Instruções de referência à memória (**tipo I**):

Load word (lw) e store word (sw)

- Instruções aritméticas e lógicas (**tipo R**):

(add, sub, and, or)

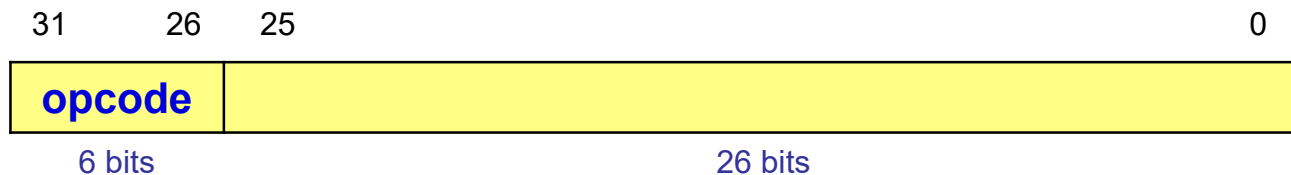
- Instruções de salto:

Branch on equal (beq) e jump (j)

Arquitetura do MIPS

► Instruções

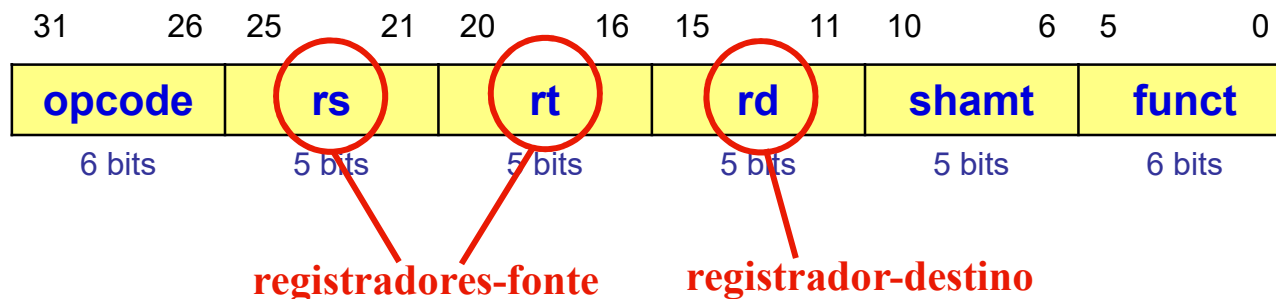
- Todas as instruções têm 32 bits
- Todas têm opcode de 6 bits
- O modo de endereçamento é codificado juntamente com o opcode



Arquitetura do MIPS

► Instruções formato R: add, sub, or, and

- Opcode = 0
- “funct” define a operação a ser feita pela ALU
- “shamt” (shift amount) é usado em instruções de deslocamento

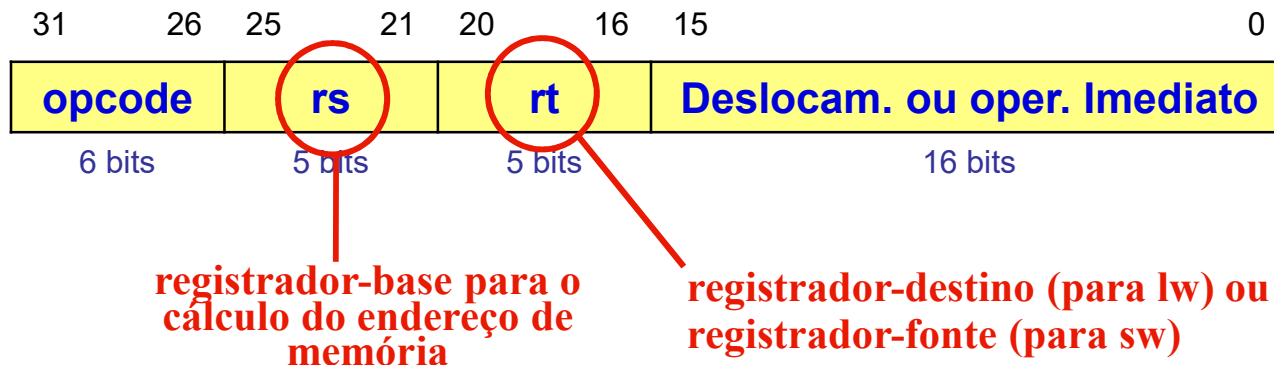


Simbólico (exemplo): add \$s1,\$s2,\$s3 ($\$s1 \leftarrow \$s2 + \$s3$)

Arquitetura do MIPS

► Instruções formato I: load word (lw) e store word (sw)

- load word (lw): opcode = 35
- store word (sw): opcode = 43



Simbólico

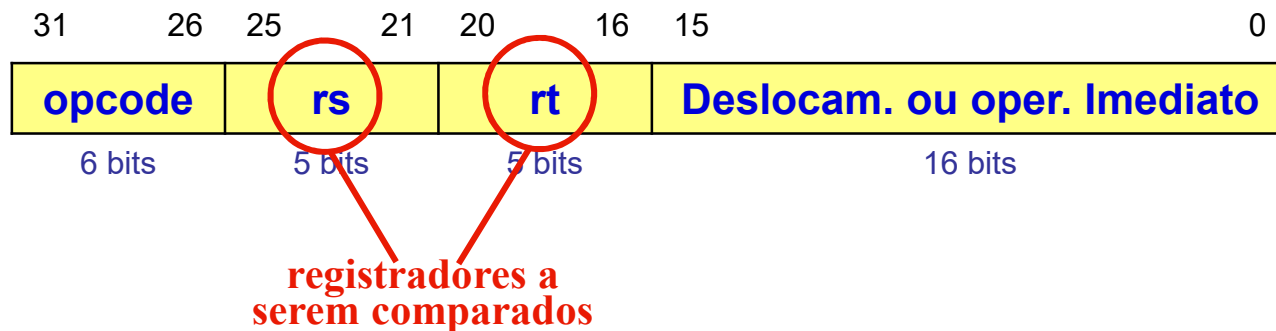
lw \$s1, deslocam(\$s2) ($\$s1 \leftarrow \text{Mem}[\$s2 + \text{deslocam}]$)

sw \$s1, deslocam(\$s2) ($\text{Mem}[\$s2 + \text{deslocam}] \leftarrow \$s1$)

Arquitetura do MIPS

► Instrução formato I: Desvio Condicional beq: branch on equal

- Opcode = 4
- Campo deslocamento usado para calcular o endereço-alvo
- Se o conteúdo do registrador cujo endereço está no campo rs for igual ao conteúdo do registrador cujo endereço está em rt, então salta para a posição endereço+PC+4



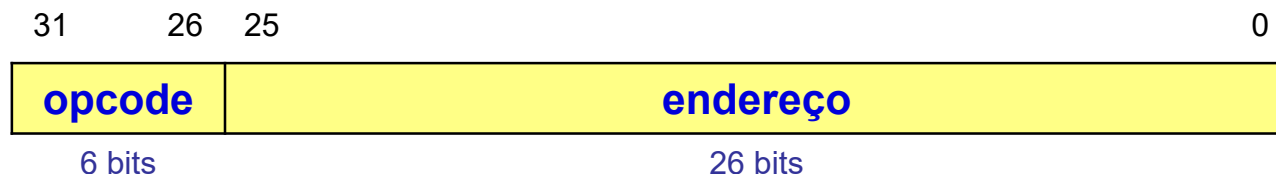
Simbólico

beq \$s1, \$s2, deslocam (if (\$s1 == \$s2) then PC ← PC+4+deslocam)

Arquitetura do MIPS

► Instrução formato J: Desvio Incondicional j: jump

- Opcode = 2
- Campo deslocamento usado para calcular o endereço-alvo



Simbólico

j endereço ($PC \leftarrow \text{endereço}$)

Organização do MIPS Monociclo

► Princípios do Projeto Eficiente

“Faça o caso comum executar mais rápido”

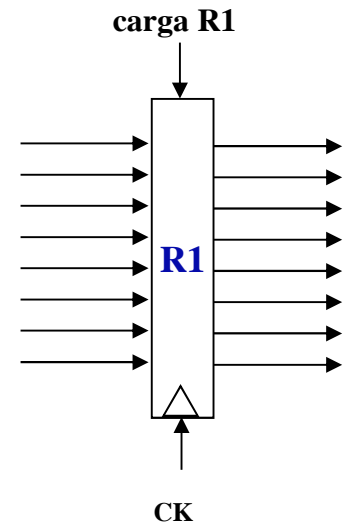
“A simplicidade favorece a regularidade”

Organização do MIPS Monociclo

► Regime de Clock (Temporização)

Iremos supor que:

- Cada registrador possui um **signal de carga** particular que é ativado com lógica direta
- Os registradores são “**disparados**” pela borda ascendente do relógio
- No desenho ao lado, quando $CK=\uparrow$, se $carga\ R1=1$, então $R1 \leftarrow entradas$



Todos lembram do comportamento de um registrador?

Como implementar um registrador a partir de um Flip-Flop D?

Organização do MIPS Monociclo

► Regime de Clock (Temporização)

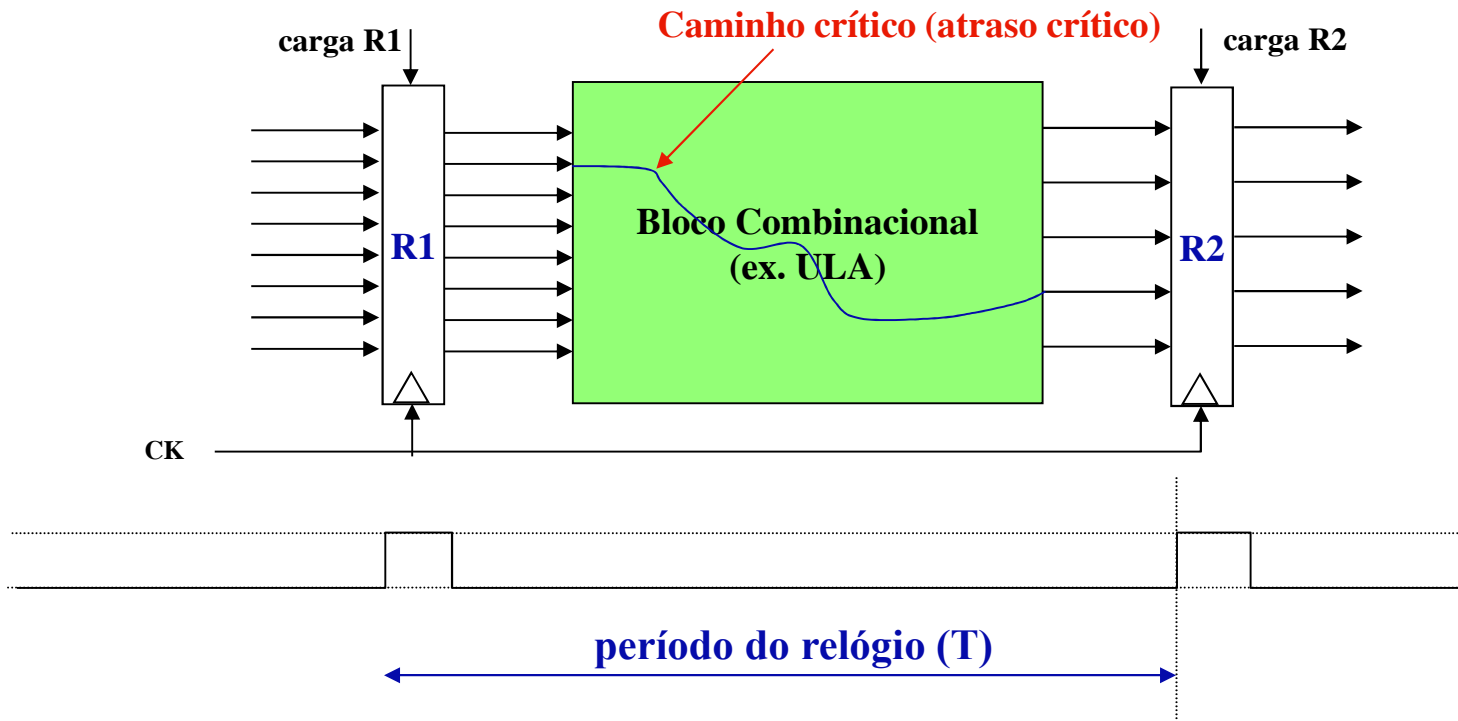
Condições para o Correto Funcionamento do Bloco Combinacional

- Quais são as características de um bloco combinacional?
- Quando as entradas de um bloco combinacional podem ser alteradas sem que haja inconsistência nos resultados de saída?

Organização do MIPS Monociclo

► Regime de Clock (Temporização)

Condições para o Correto Funcionamento do Bloco Combinacional

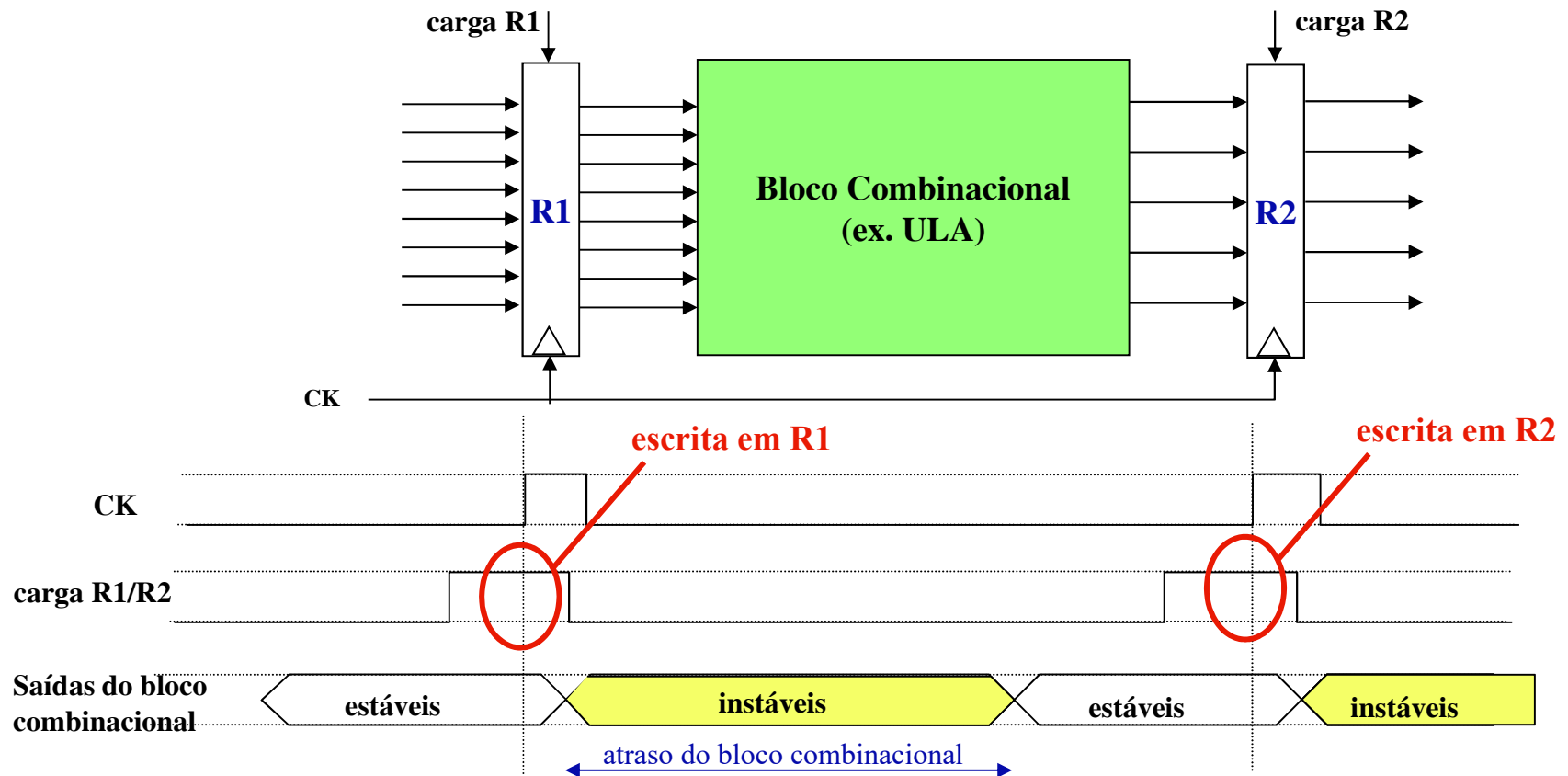


O atraso crítico do bloco combinacional deve ser menor que o período do relógio!

Organização do MIPS Monociclo

► Regime de Clock (Temporização)

Temporização para o Correto Funcionamento do Bloco Comb.



Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Nesta primeira implementação iremos considerar que:

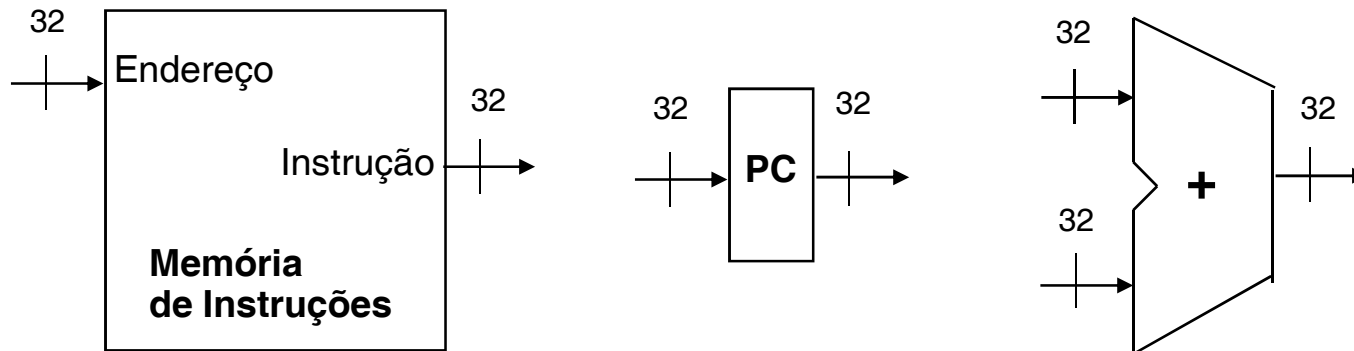
- Qualquer instrução é executada em um único ciclo de relógio
- O período do relógio será longo o suficiente para acomodar qualquer instrução
 - Na verdade, o período do relógio será função da instrução mais demorada

Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Elementos Necessários Para a Busca da Instrução:

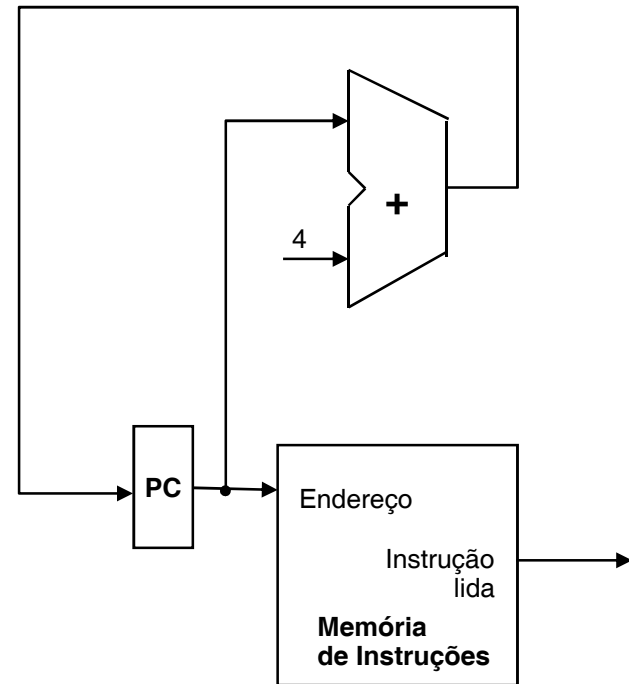
- A memória onde estão armazenadas as instruções
- O contador de programa (PC) para armazenar o endereço da instrução
- Um somador para calcular o endereço da próxima instrução



Organização do MIPS Monociclo

► Bloco Operativo Monociclo

- O contador de programa contém o endereço da instrução em execução
- O endereço da próxima instrução é obtido pela soma de 4 posições ao contador de programa
- A instrução lida é usada por outras porções do bloco operativo

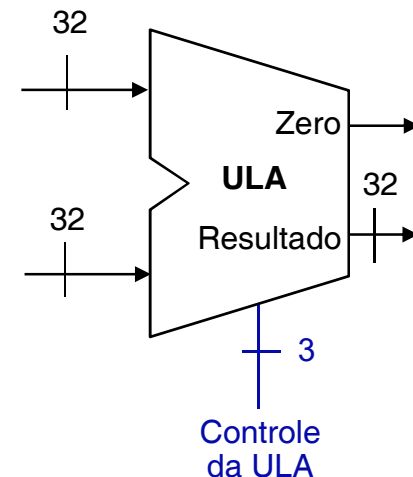
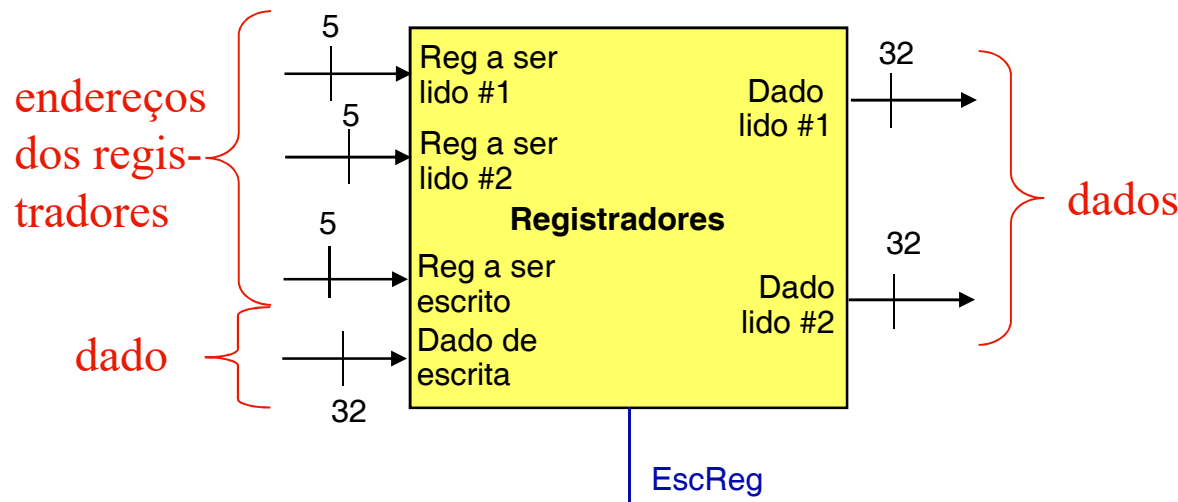


Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Elementos Necessários para Execução de Instruções tipo R:

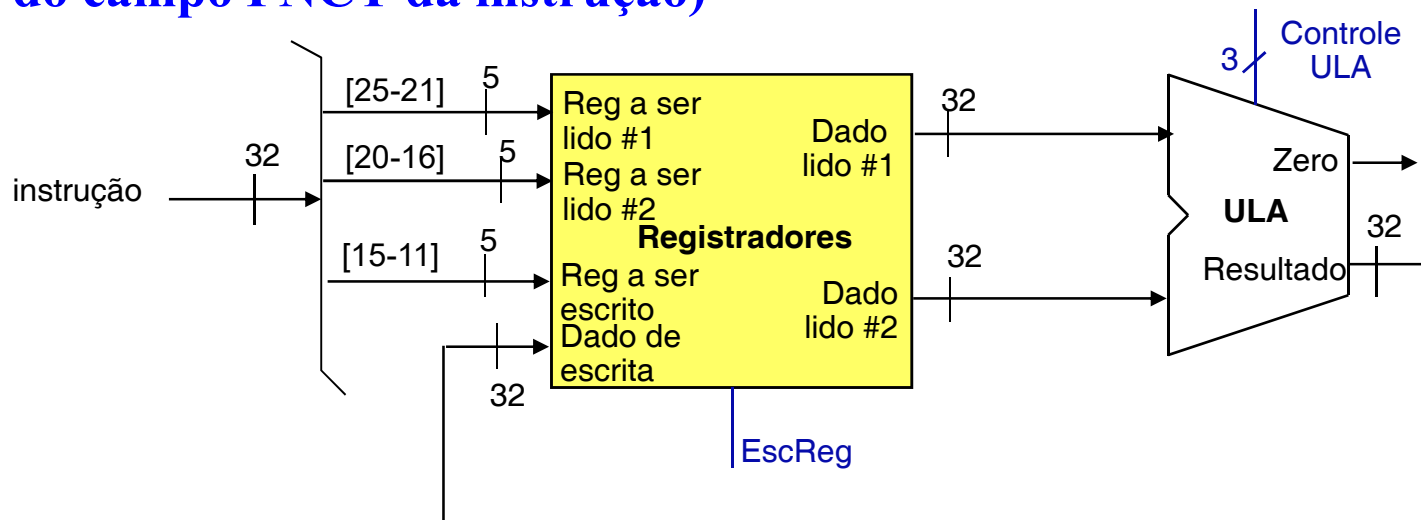
- Um banco de registradores para armazenar os operandos e o resultado das operações
- Uma Unidade Lógica/Aritmética (ULA) que será utilizada para realizar as operações



Organização do MIPS Monociclo

► Bloco Operativo Monociclo

- A instrução (fornecida pelo hardware de busca de instruções) contém o endereço de três registradores
- Dois destes registradores são lidos e passados para a ULA realizar a operação
- O resultado é armazenado em um terceiro registrador
- O controle da ULA determina a operação que será realizada (a partir do campo FNCT da instrução)

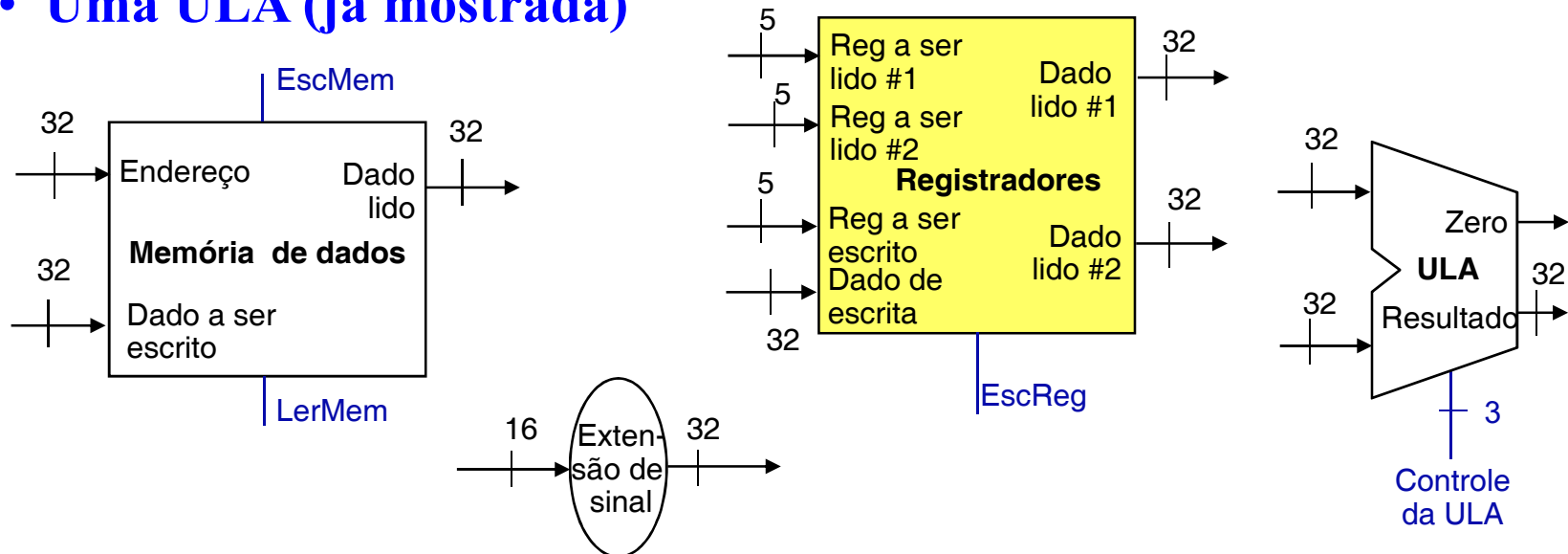


Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Elementos Necessários para Executar lw e sw

- Uma memória de dados
- Um módulo de extensão de sinal
- Um banco de registradores (já mostrado)
- Uma ULA (já mostrada)

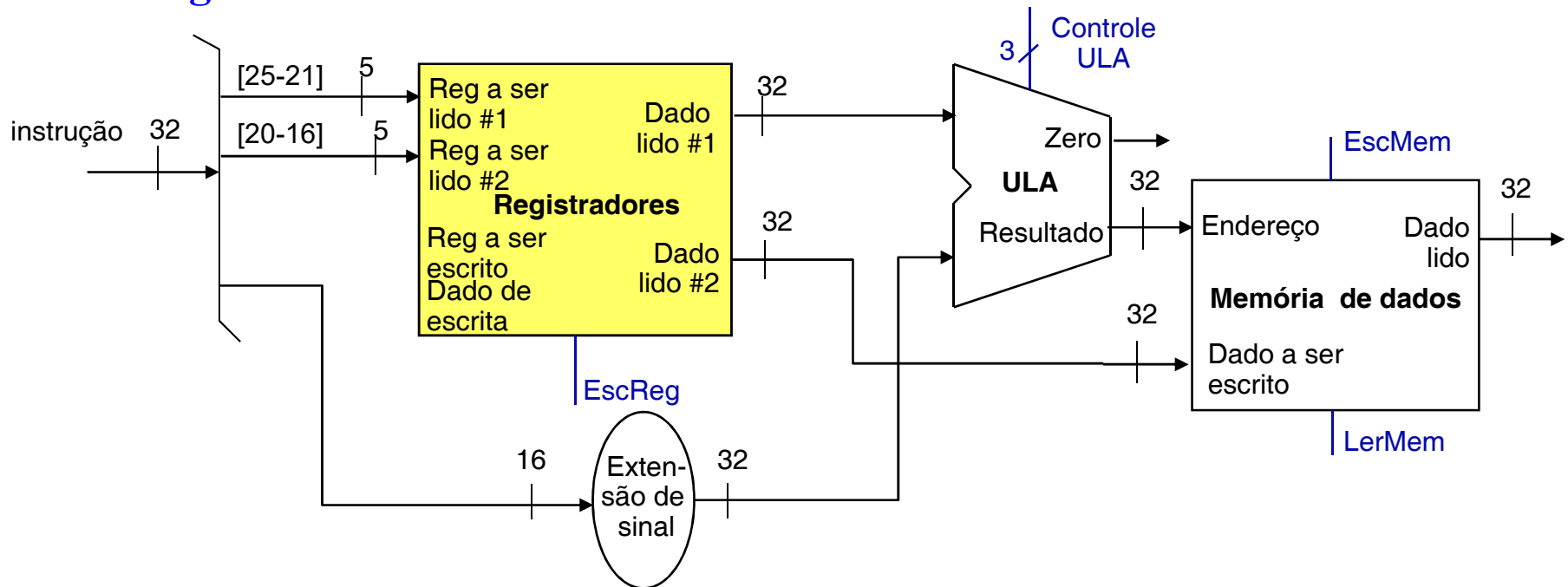


Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Combinando os Elementos para uma Escrita na Memória (sw)

- O endereço de escrita é obtido pela soma de um registrador de base (registrador 1) com um deslocamento de 16 bits estendido para 32 bits
- O registrador 2 é escrito na memória

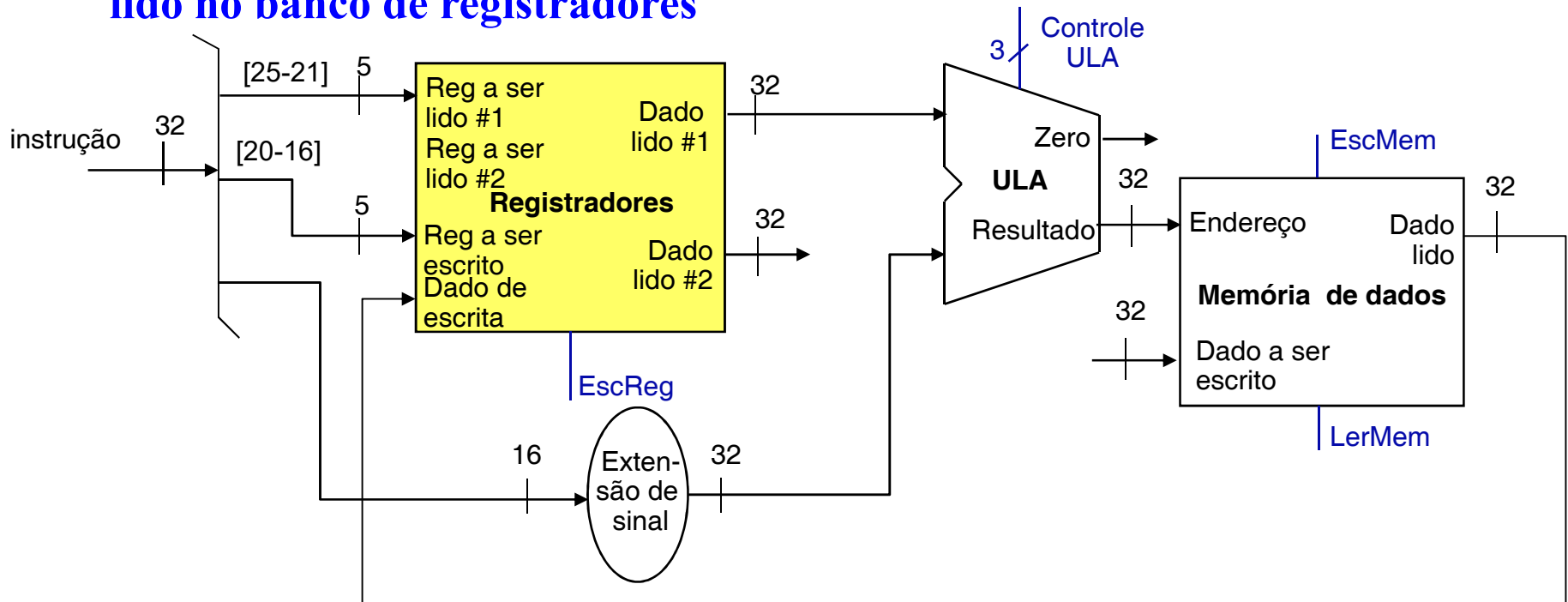


Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Combinando os Elementos para uma Leitura da Memória (lw)

- O processo de leitura é semelhante ao de escrita
- A diferença básica é a existência de um caminho para escrever o valor lido no banco de registradores

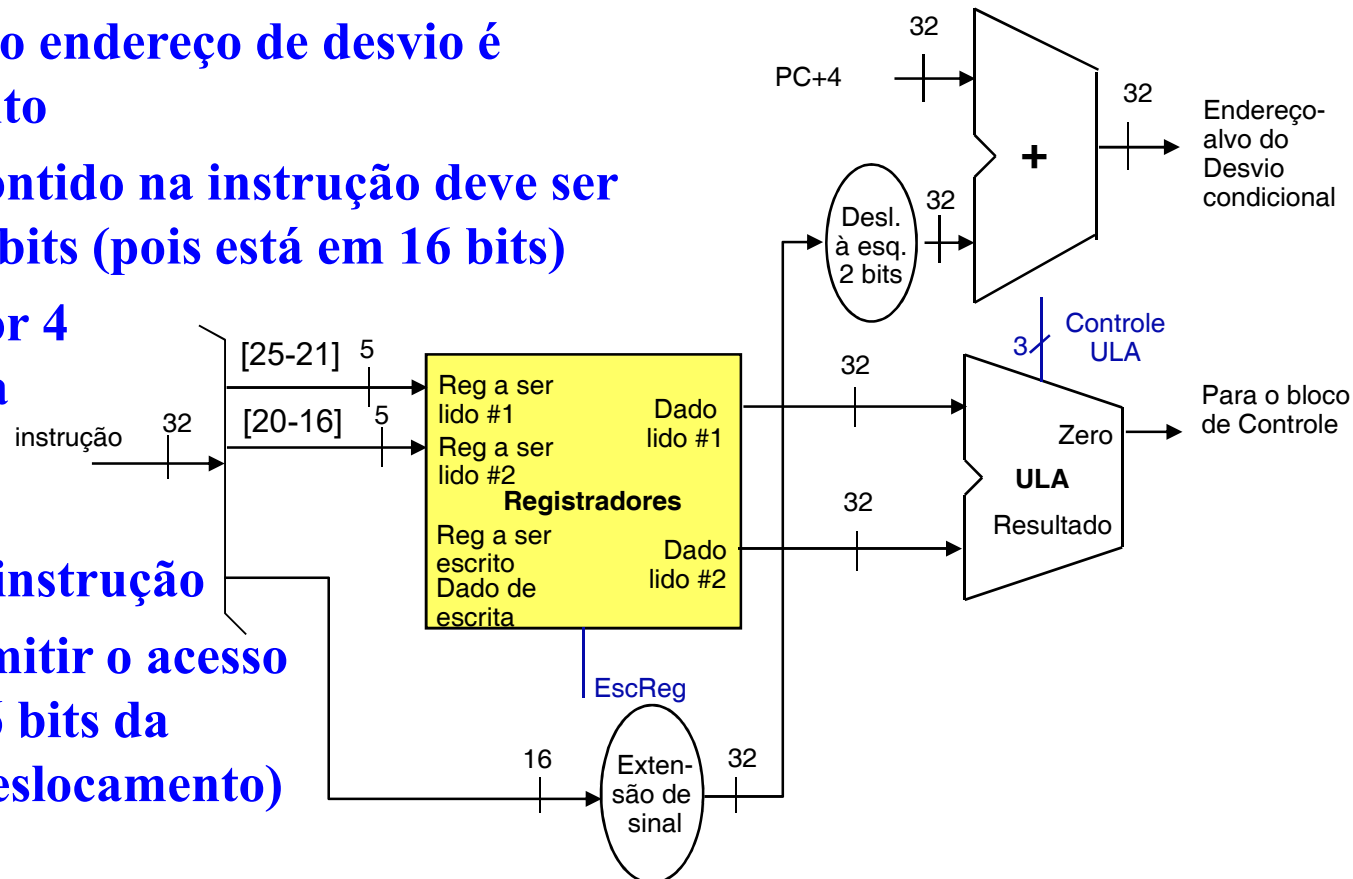


Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Elementos Necessários Implementar um Branch on Equal

- Comparar dois registradores usando a ULA para fazer uma subtração
- Se ocorrer desvio, o endereço de desvio é $PC+4+\text{deslocamento}$
- O deslocamento contido na instrução deve ser estendido para 32 bits (pois está em 16 bits)
- A multiplicação por 4 é necessária, pois a memória de instruções utiliza 4 bytes para cada instrução
- A vantagem é permitir o acesso a 2^{18} endereços (16 bits da instrução + 2 de deslocamento)



Organização do MIPS Monociclo

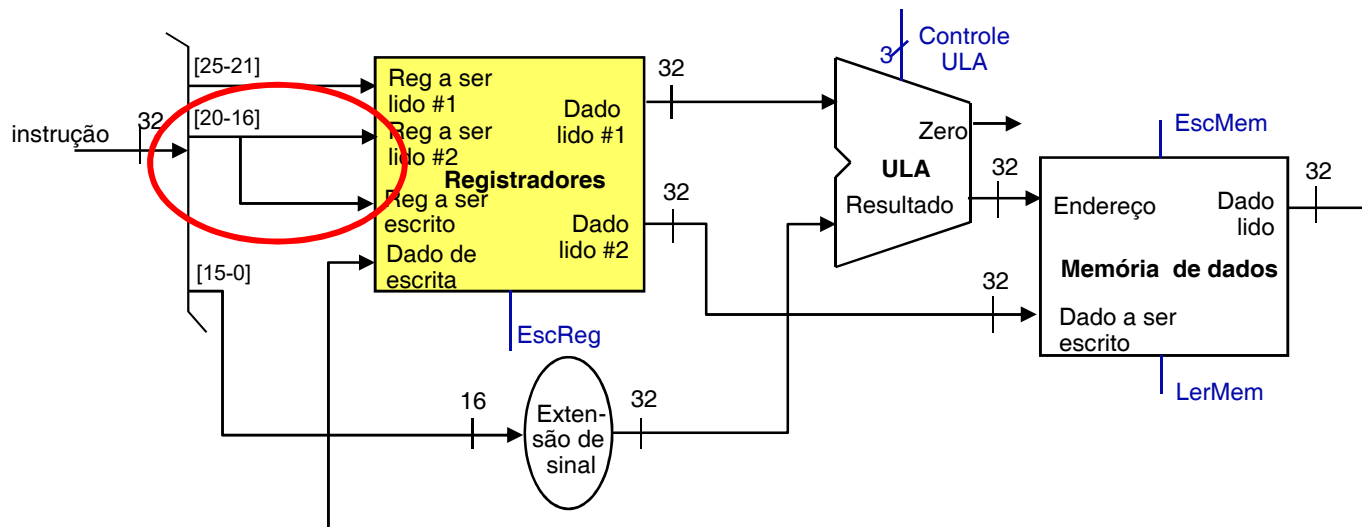
► Compondo o Bloco Operativo Monociclo

- Nesta versão de bloco operativo qualquer instrução executa em um ciclo de relógio
- Isto significa que o período do relógio deverá ser suficientemente longo para acomodar a instrução mais demorada
- **Durante a execução de uma instrução qualquer, cada unidade funcional só pode ser usada uma única vez**
- Por isso necessitamos de uma memória de instruções e outra de dados
- Ao combinarmos as porções de bloco operativo vistas anteriormente, veremos que muitas unidades funcionais podem ser compartilhadas

Organização do MIPS Monociclo

► Compondo o Bloco Operativo Monociclo

Recursos para executar instruções lw ou sw

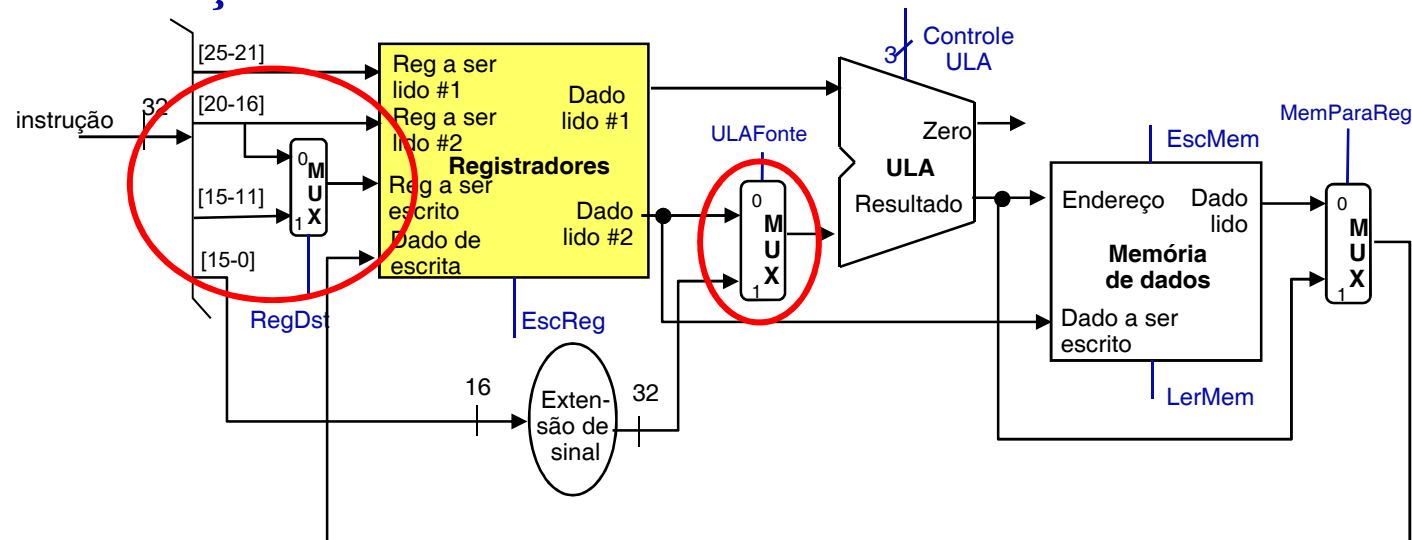


- Para sw, campo Rt (bits 20-16) designam registrador cujo conteúdo será escrito na memória de dados
- Para lw, Rt (bits 20-16) designam registrador que será carregado com valor lido da memória de dados

Organização do MIPS Monociclo

► Compondo o Bloco Operativo Monociclo

Combinando os recursos para executar instruções tipo R ou instruções lw e sw...

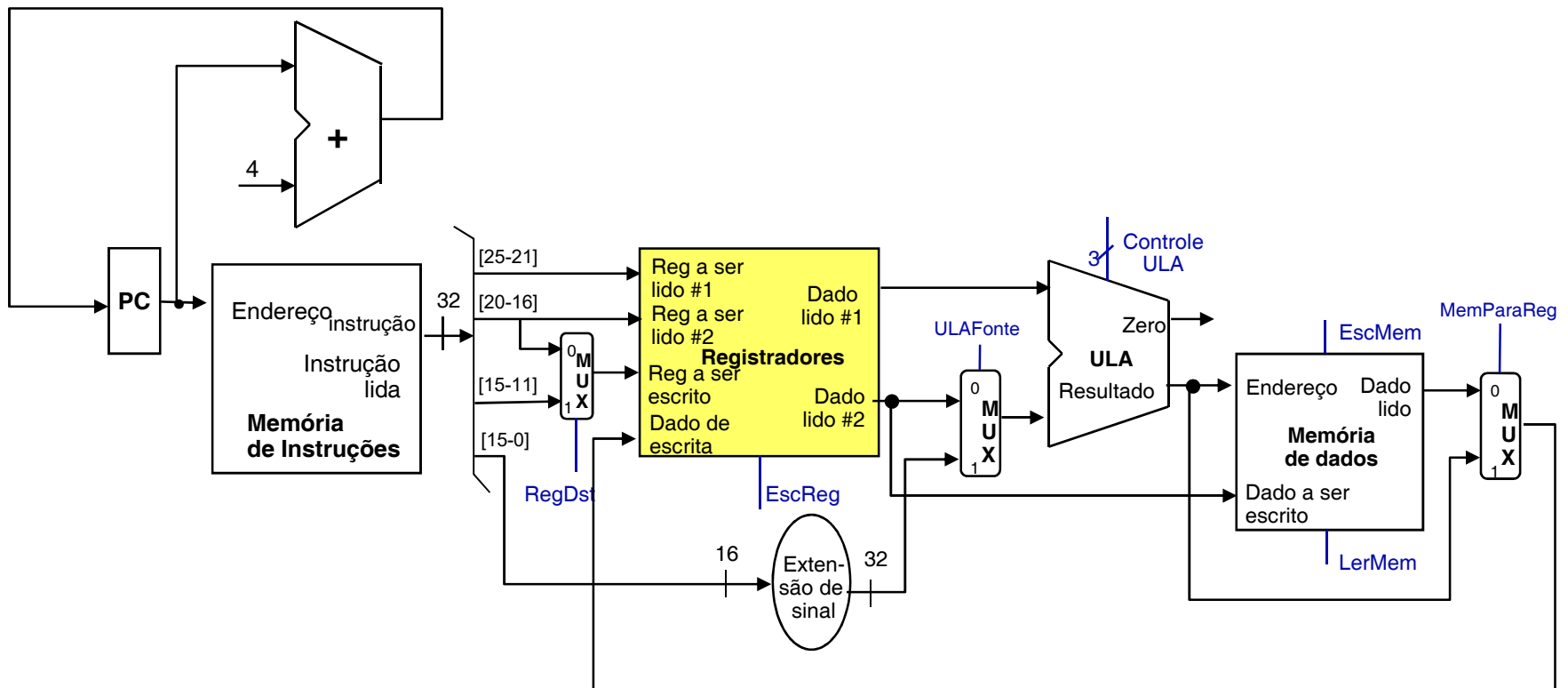


- Para **lw**, o endereço do registrador a ser escrito está no campo Rt (bits 20-16)
- Para **instruções tipo R**, o endereço do registrador a ser escrito está no campo Rd (bits 15-11)

Organização do MIPS Monociclo

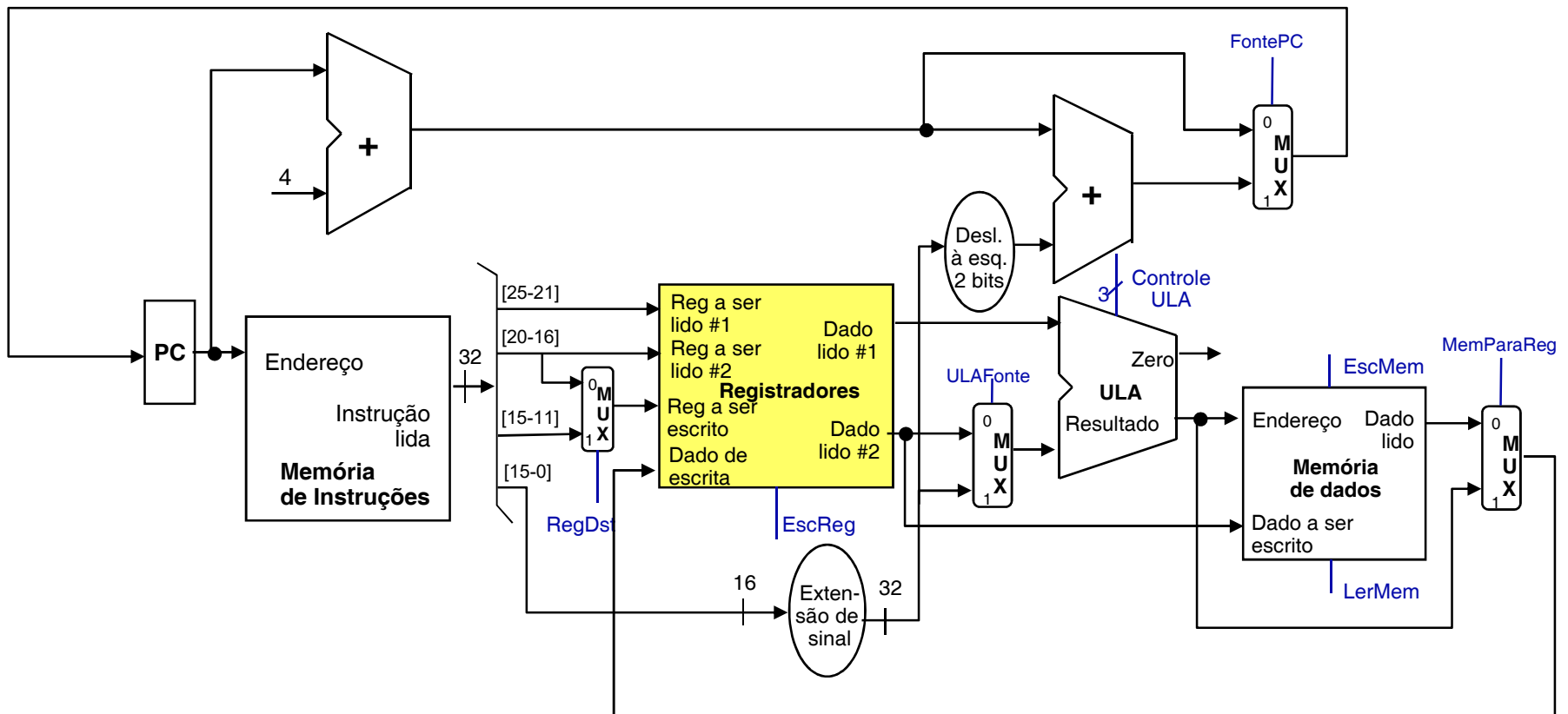
► Compondo o Bloco Operativo Monociclo

Acrescentando os Recursos para a Busca da Instrução e o Cálculo do Próximo Endereço (exceto em desvios)



Organização do MIPS Monociclo

► Compondo o Bloco Operativo Monociclo Acrescentando os Recursos para Executar Desvio Condicional (Branch on equal)



Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Estimativa da Frequência Máxima de Funcionamento

- ao final de cada ciclo de relógio o PC é carregado com um novo valor
- mudança no valor do PC se propaga através de uma grande lógica combinacional
 - memória de instruções => banco de registradores => ULA => memória de dados => banco de registradores
- período do ciclo de relógio deve ser maior do que máximo atraso de propagação através desta lógica combinacional

Organização do MIPS Monociclo

► Bloco Operativo Monociclo

Exemplo de Cálculo

- supondo os seguintes atrasos:
 - memórias: 4 ns
 - ULA: 2 ns
 - banco de registradores: 1 ns
 - somadores: 1 ns
 - demais componentes: atraso desprezível

| instrução | busca | Lê registradores | Cálculo na ULA | Acessa memória de dados | Escreve em registrador | Total |
|-------------------|-------|------------------|----------------|-------------------------|------------------------|-------|
| add, sub, and, or | 4ns | 1ns | 2ns | -- | 1ns | 8ns |
| beq | 4ns | 1ns | 2ns | -- | -- | 7ns |
| sw | 4ns | 1ns | 2ns | 4ns | -- | 11ns |
| lw | 4ns | 1ns | 2ns | 4ns | 1ns | 12ns |

período do ciclo de relógio deve ser maior do que 12 ns (cerca de 83 MHz)

Organização do MIPS Monociclo

► Bibliografia recomendada

- PATTERSON, David A.; HENESSY, John L. **Organização e Projeto de Computadores: a interface hardware/software**. 2^a.ed. Rio de Janeiro: LTC, 2000.