Universidade Federal de Santa Catarina

Centro Tecnológico

Departamento de Informática e Estatística

Ciências da Computação & Engenharia Eletrônica

Sistemas Digitais

INE 5406

Aula 11-T

4. O Processador MIPS multiciclo: execução das instruções e construção do bloco de controle.

Profs. José Luís Güntzel e Cristina Meinhardt

{j.guntzel, cristina.meinhardt}@ufsc.br

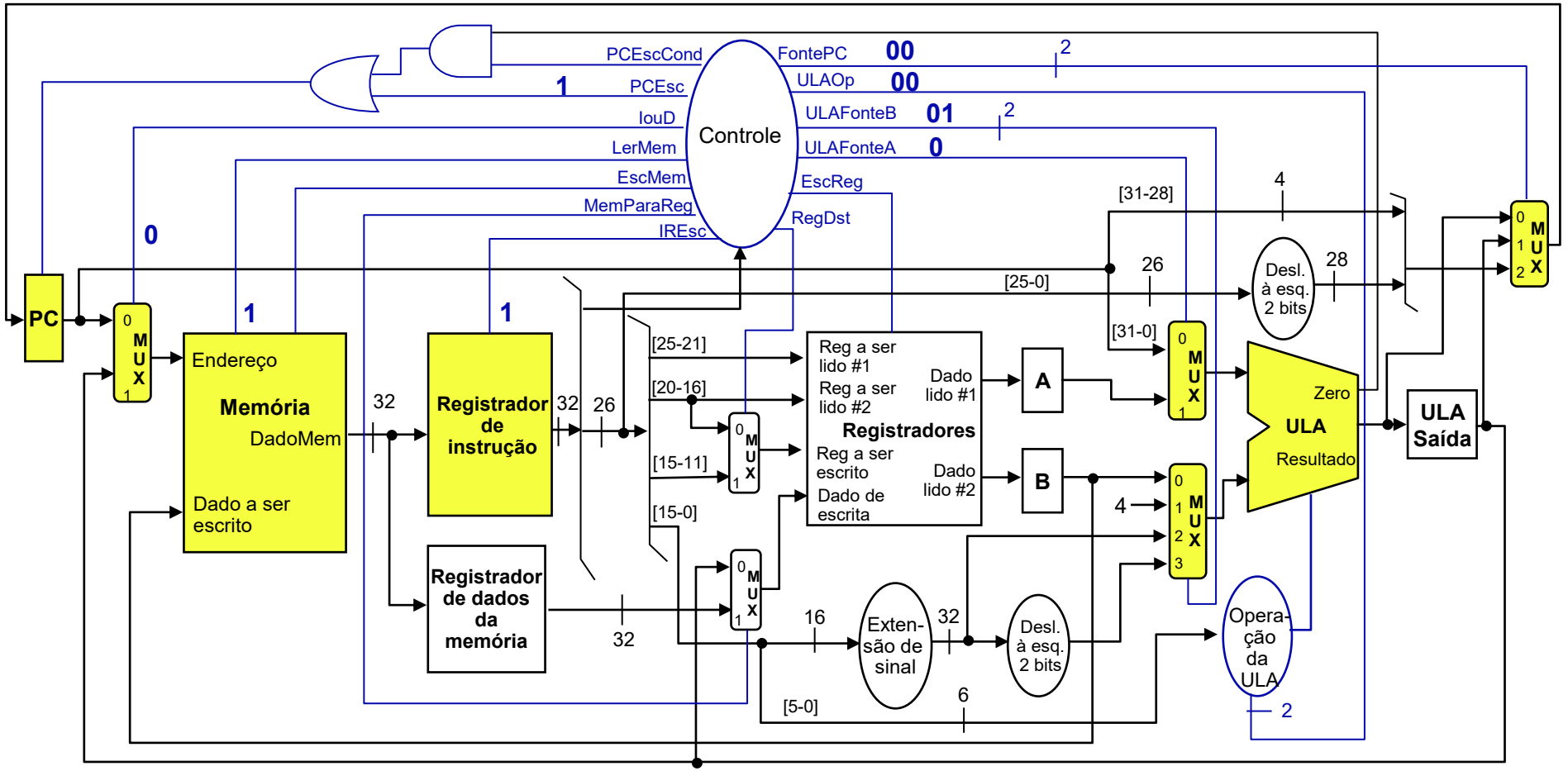
O Processador MIPS Multiciclo

Passos Necessários para Qualquer Instrução no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	**4**	**5**	**4**	**3**	**3**

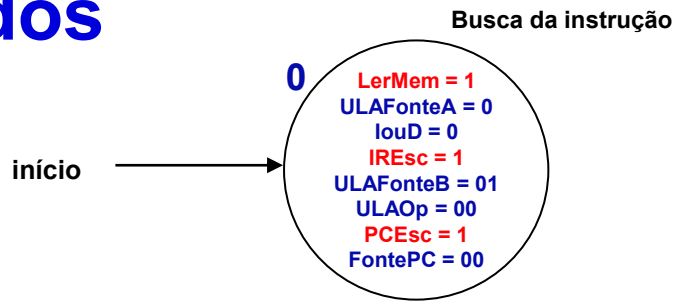
O Processador MIPS Multiciclo

Busca da Instrução (e PC ← PC+4)



O Processador MIPS Multiciclo

Máquina de Estados



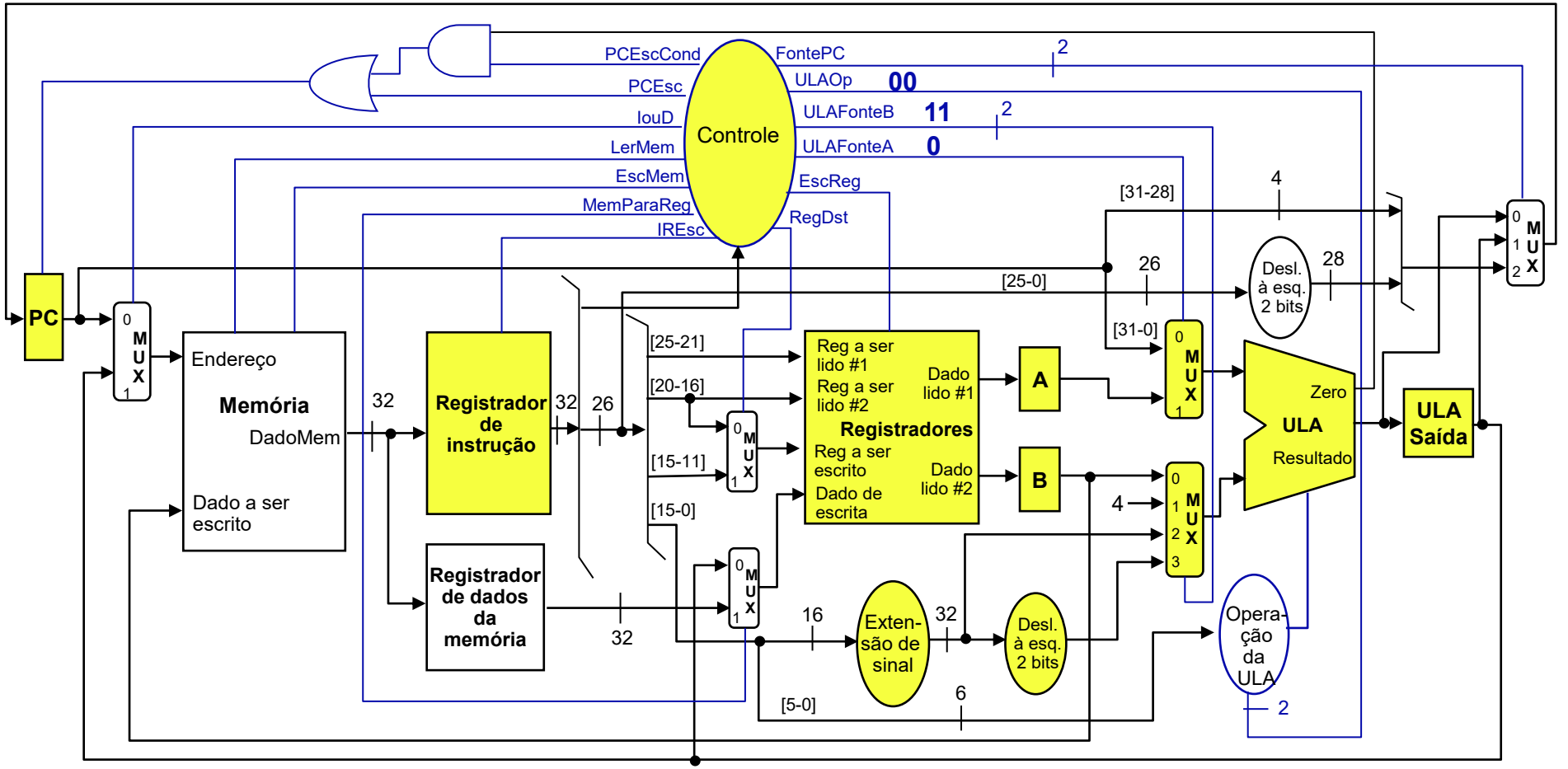
O Processador MIPS Multiciclo

Passos Necessários para Qualquer Instrução no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	4	5	4	3	3

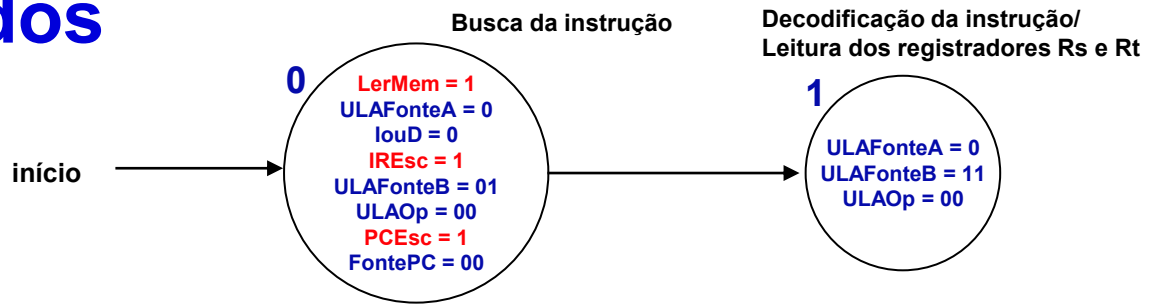
O Processador MIPS Multiciclo

Decodificação da instrução & Leit. Rs e Rt (cálculo do end. p/ beq)



O Processador MIPS Multiciclo

Máquina de Estados



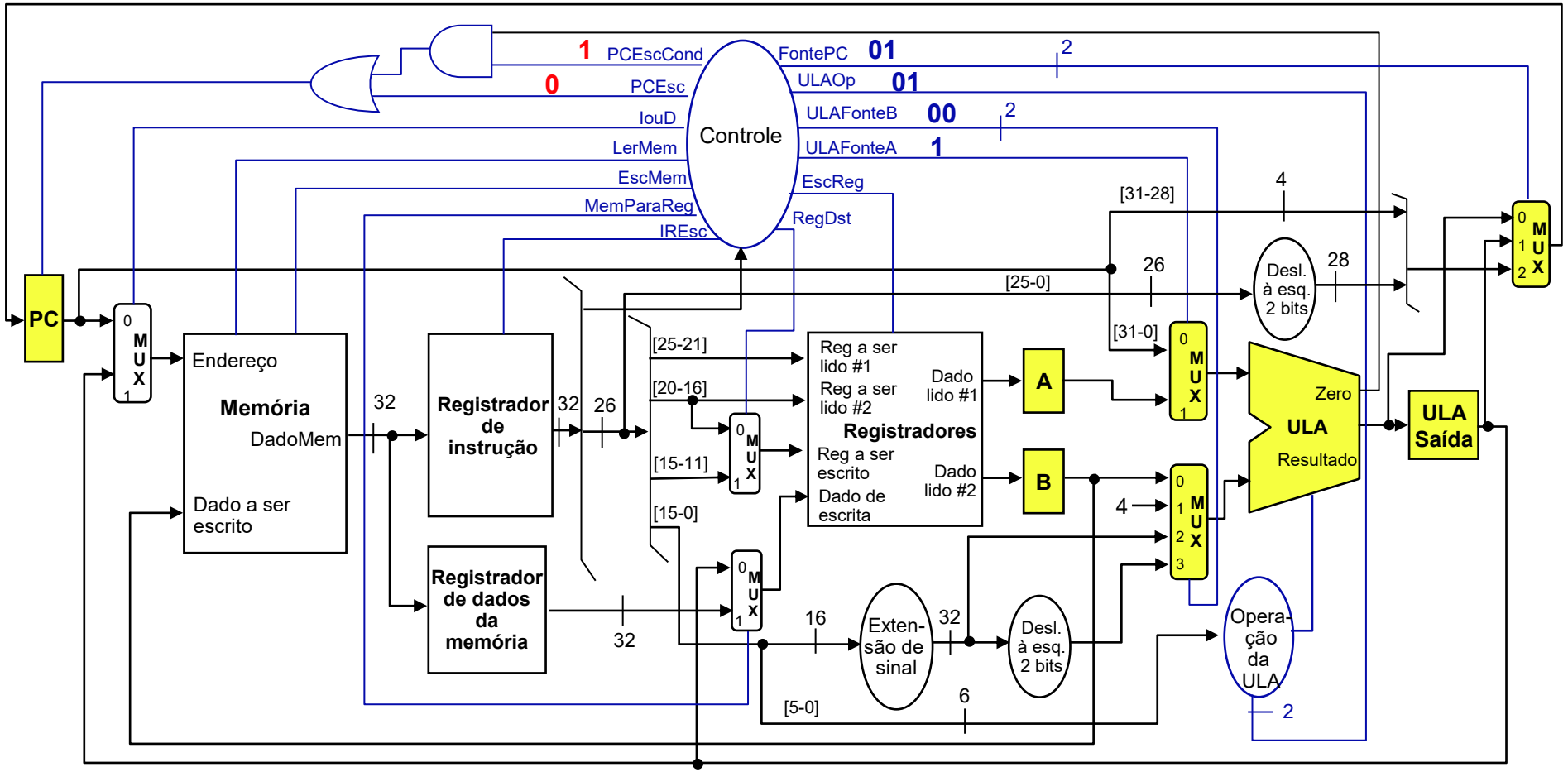
O Processador MIPS Multiciclo

Passos Necessários para Instrução beq no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	**4**	**5**	**4**	**3**	**3**

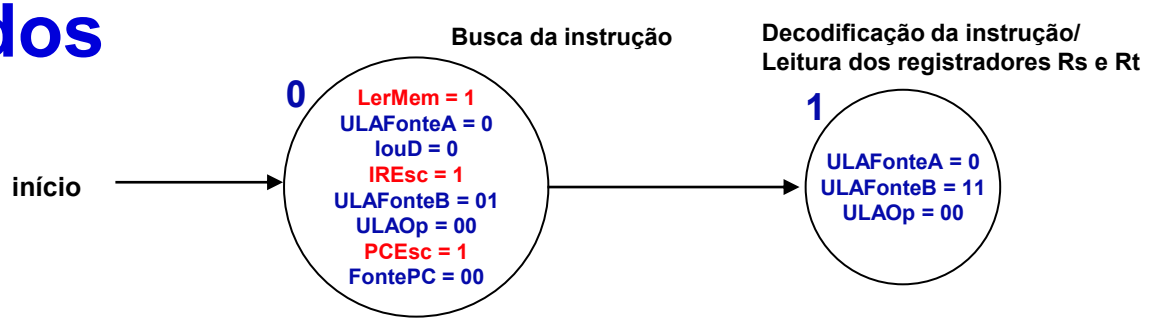
O Processador MIPS Multiciclo

Execução da Instrução beq: testa se A == B



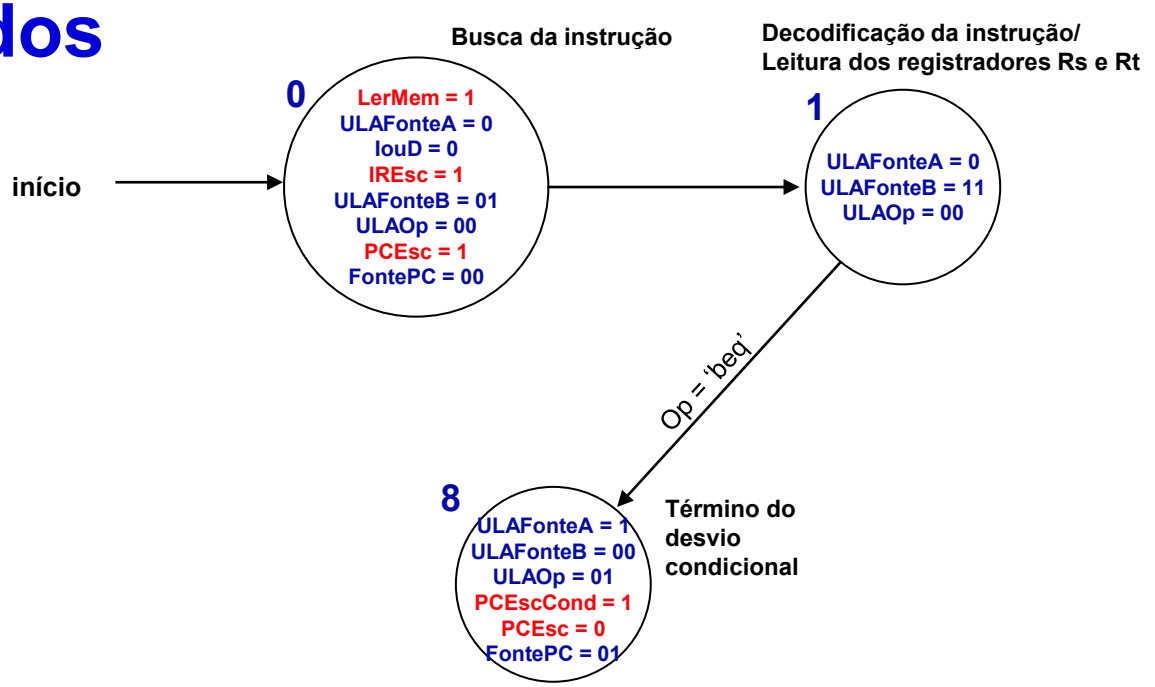
O Processador MIPS Multiciclo

Máquina de Estados



O Processador MIPS Multiciclo

Máquina de Estados



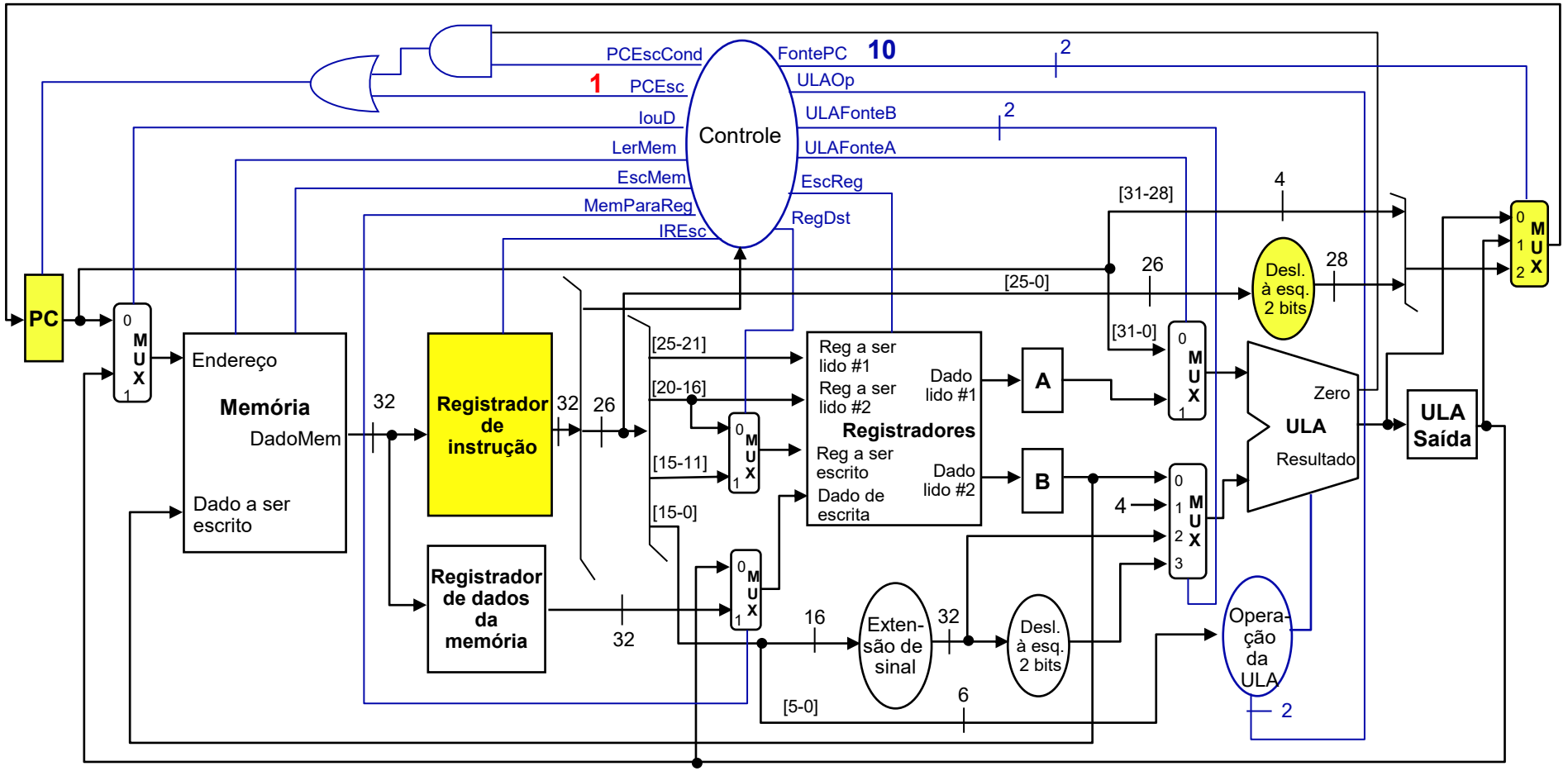
O Processador MIPS Multiciclo

Passos Necessários para Instrução jump no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	4	5	4	3	3

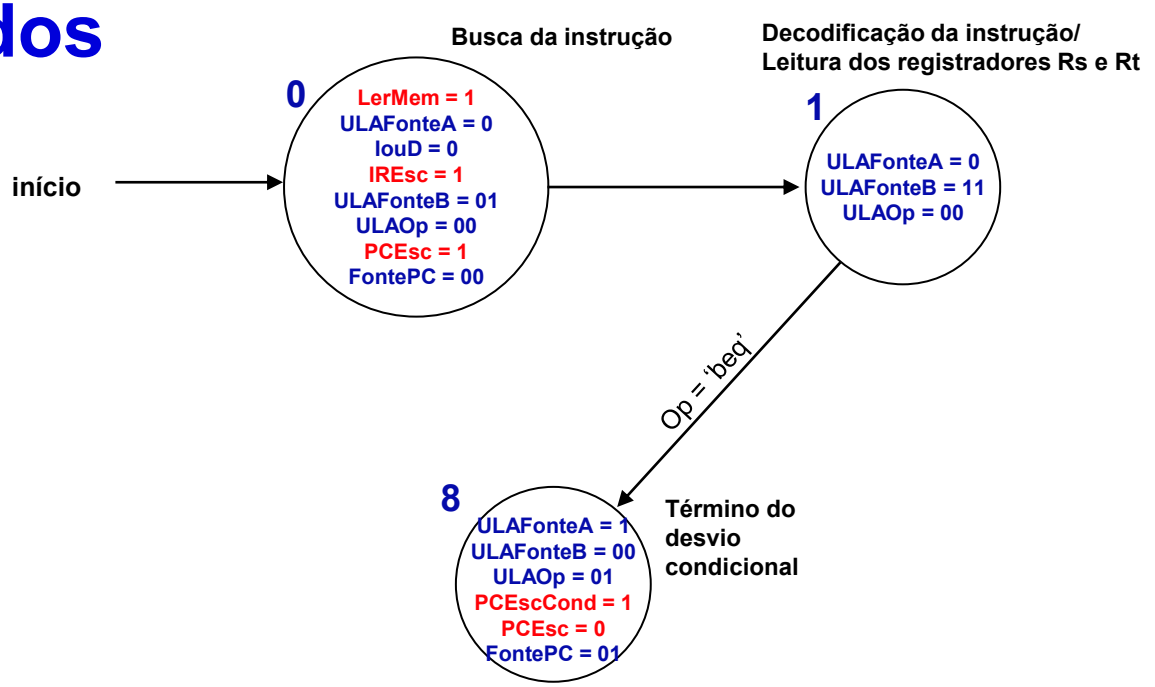
O Processador MIPS Multiciclo

Execução da Instrução jump: somente seleção de caminho



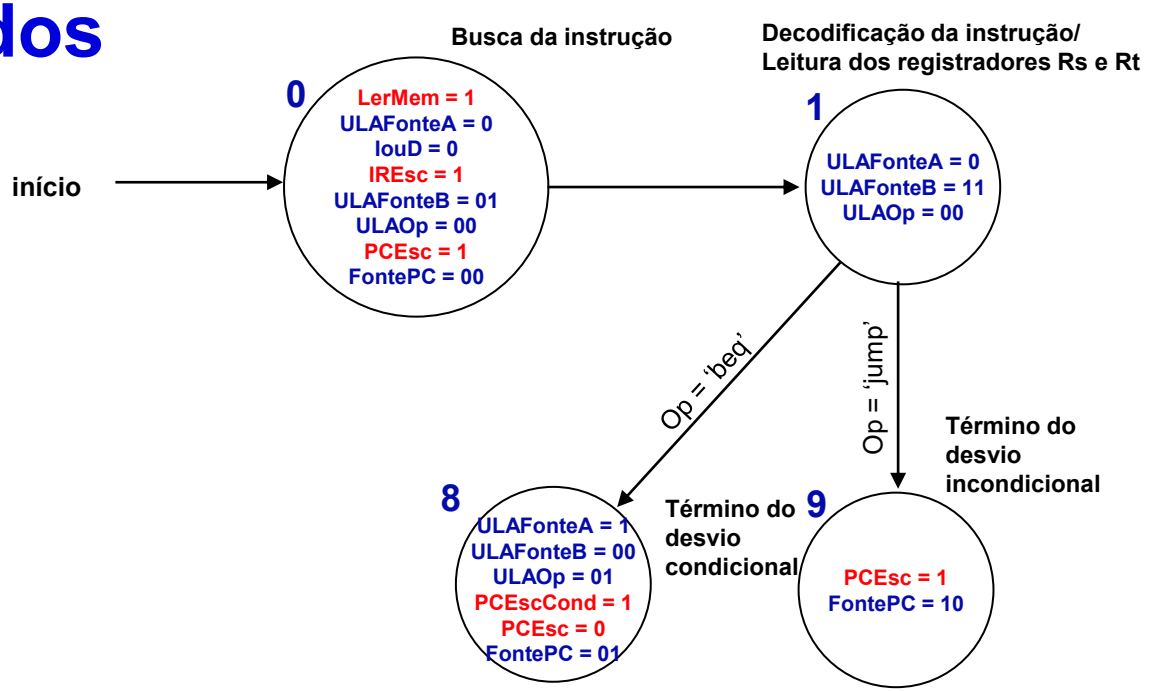
O Processador MIPS Multiciclo

Máquina de Estados



O Processador MIPS Multiciclo

Máquina de Estados



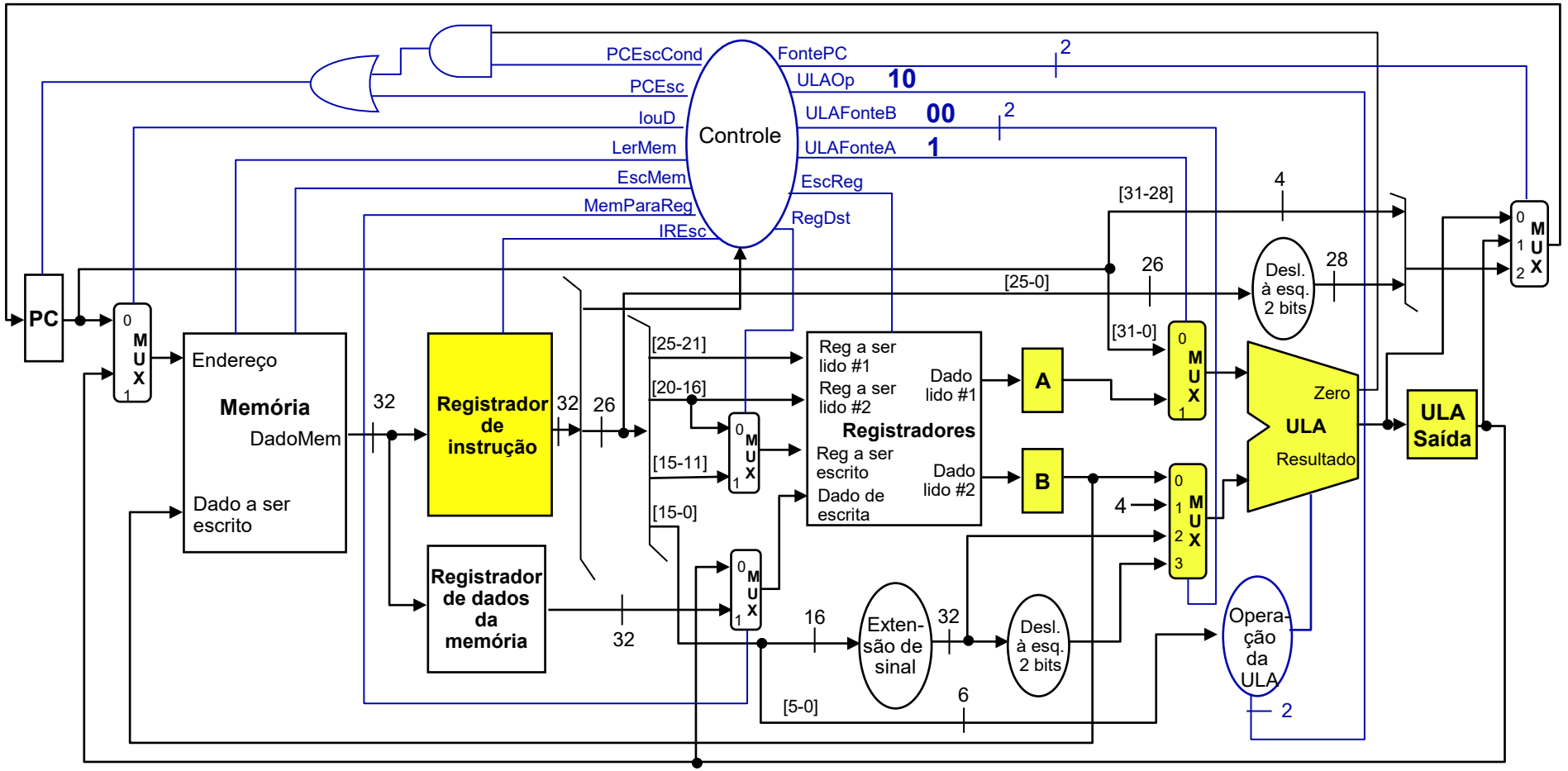
O Processador MIPS Multiciclo

Passos Necessários para Instruções Tipo R no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	4	5	4	3	3

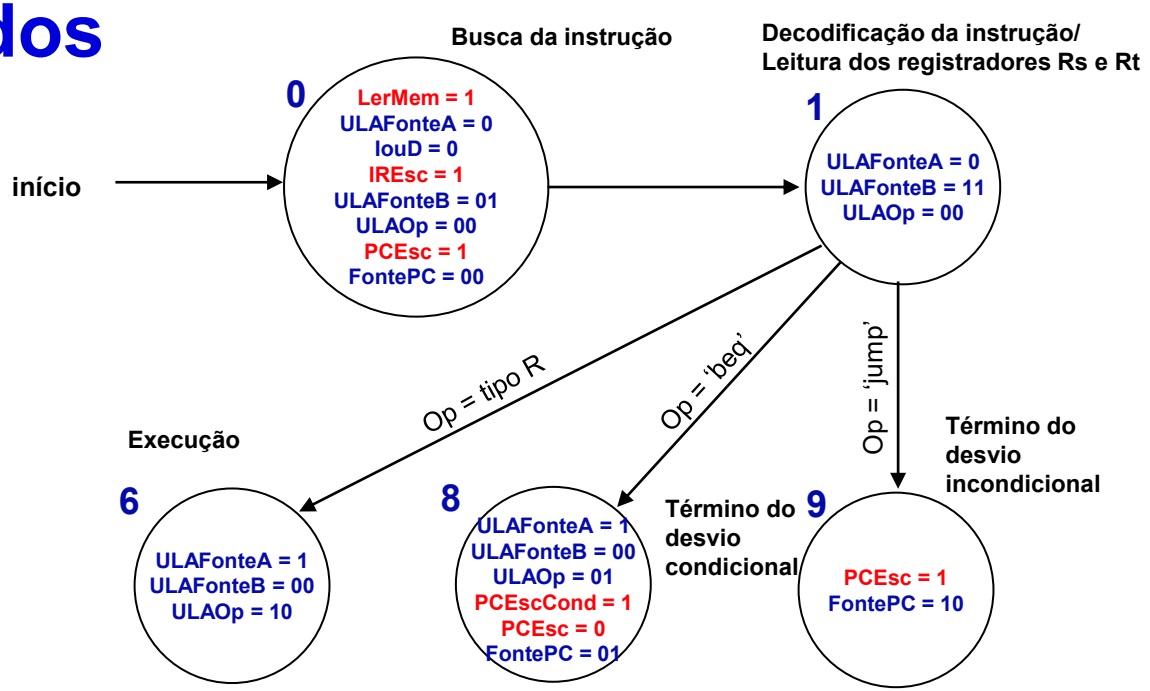
O Processador MIPS Multiciclo

Execução de Instruções Tipo R: execução na ULA



O Processador MIPS Multiciclo

Máquina de Estados



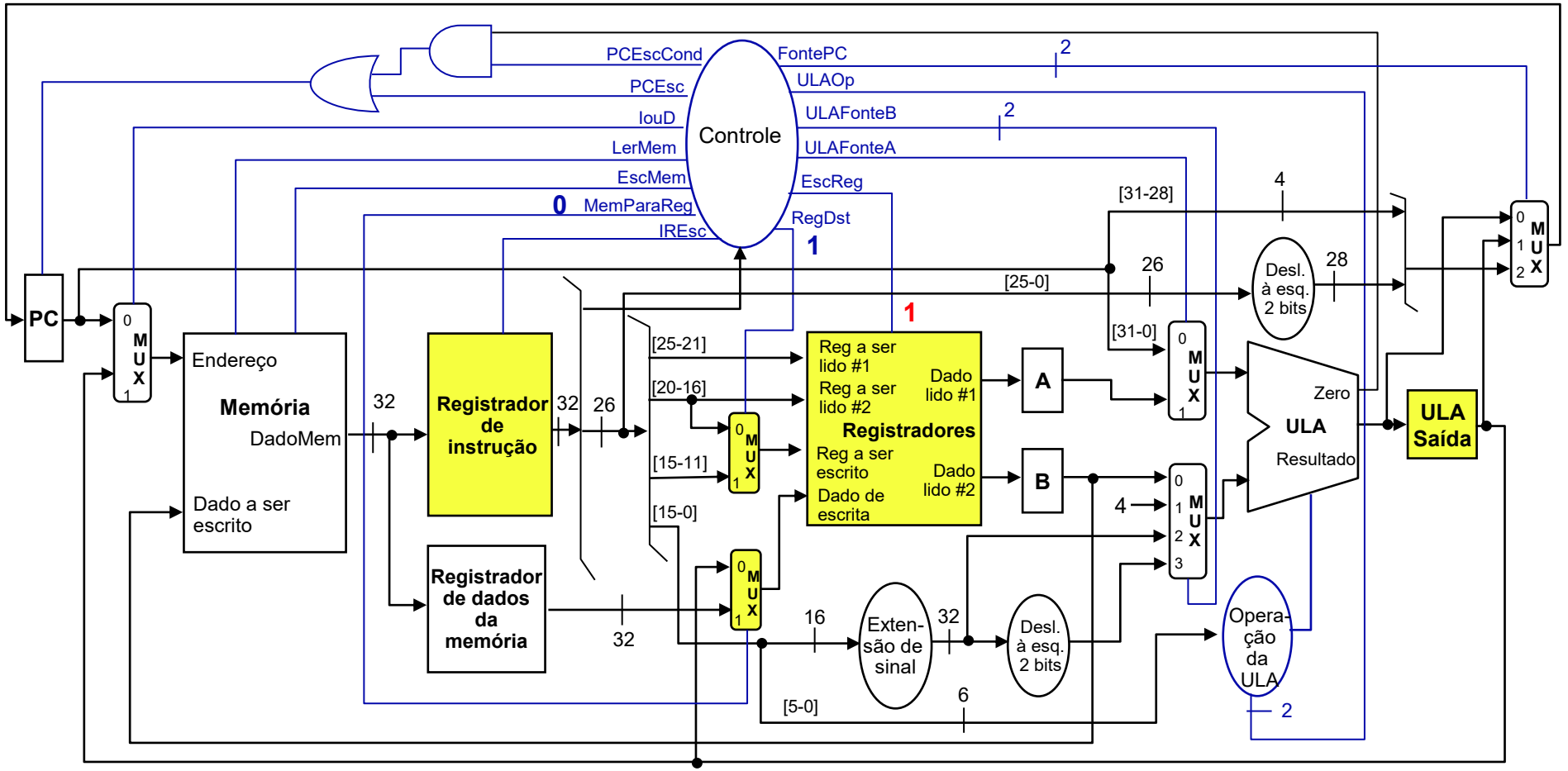
O Processador MIPS Multiciclo

Passos Necessários para Instruções Tipo R no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	4	5	4	3	3

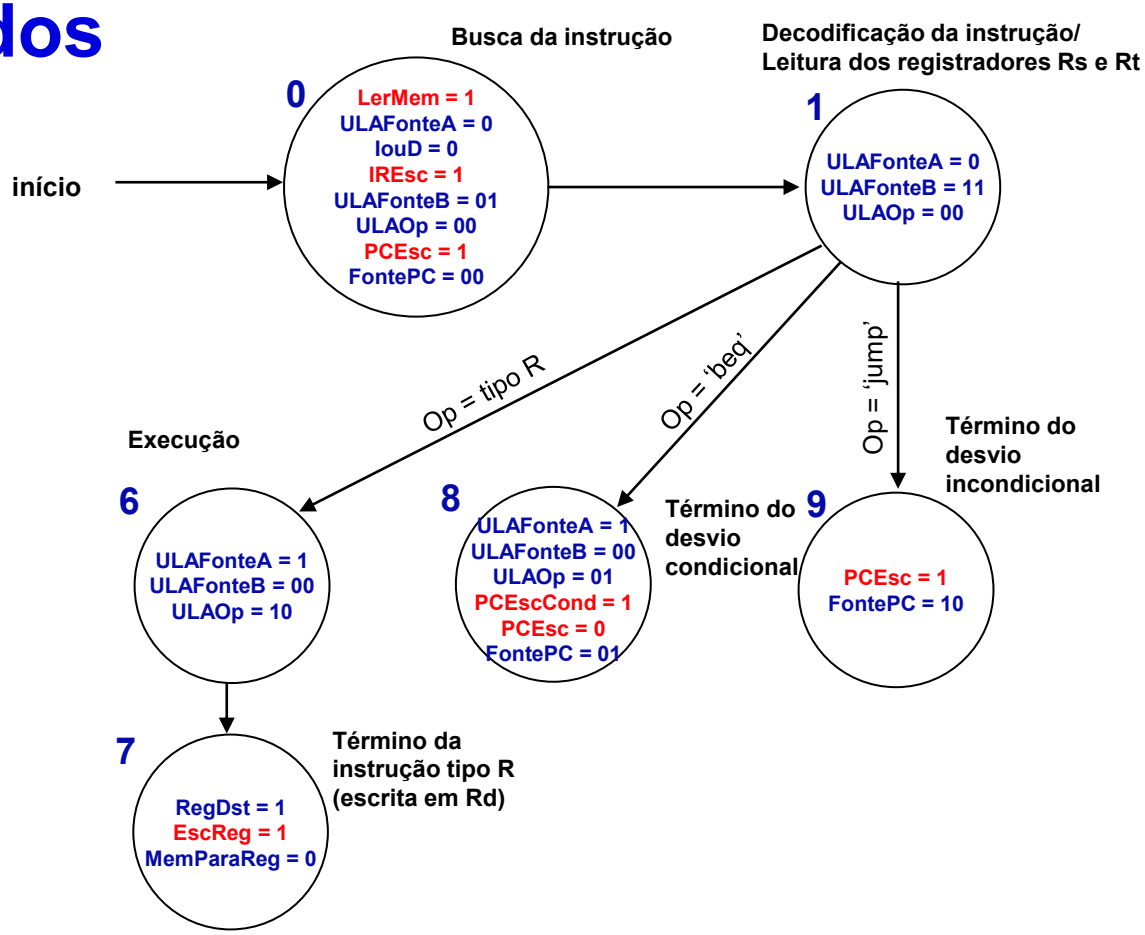
O Processador MIPS Multiciclo

Execução de Instruções Tipo R: escreve em registrador



O Processador MIPS Multiciclo

Máquina de Estados



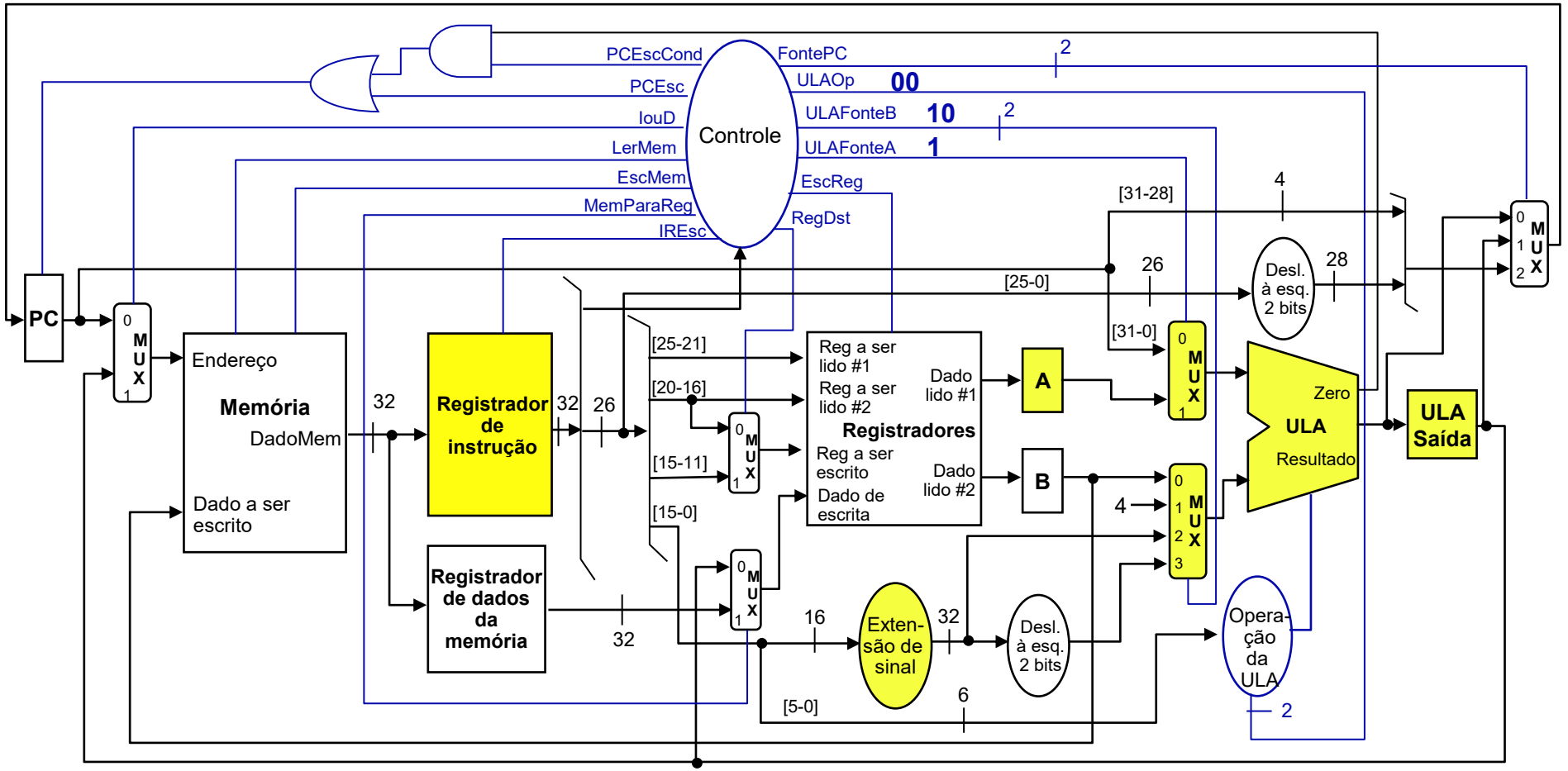
O Processador MIPS Multiciclo

Passos Necessários para Instruções lw e sw no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	4	5	4	3	3

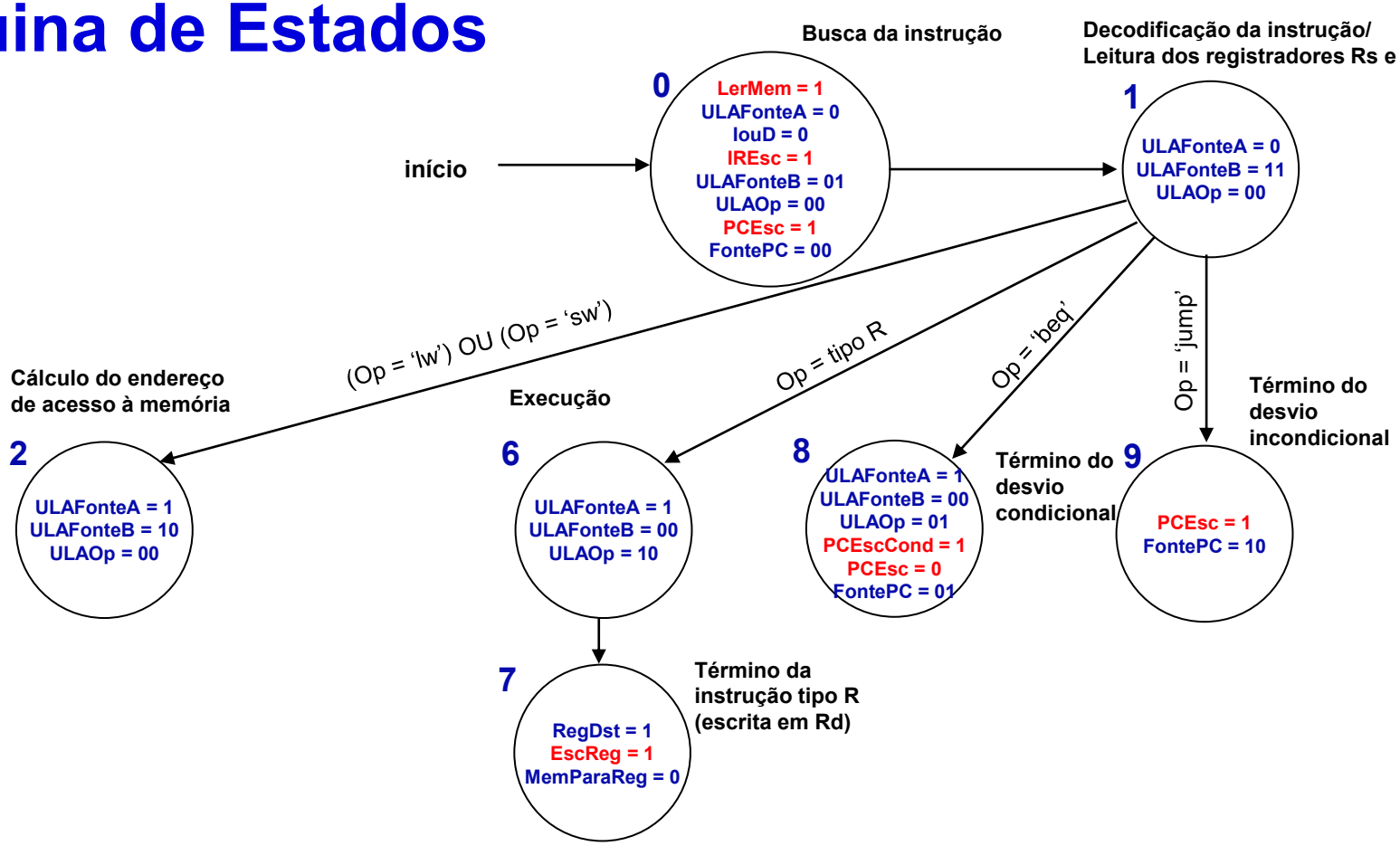
O Processador MIPS Multiciclo

Execução de Instruções: execução lw/sw (calc. endereço desvio)



O Processador MIPS Multiciclo

Máquina de Estados



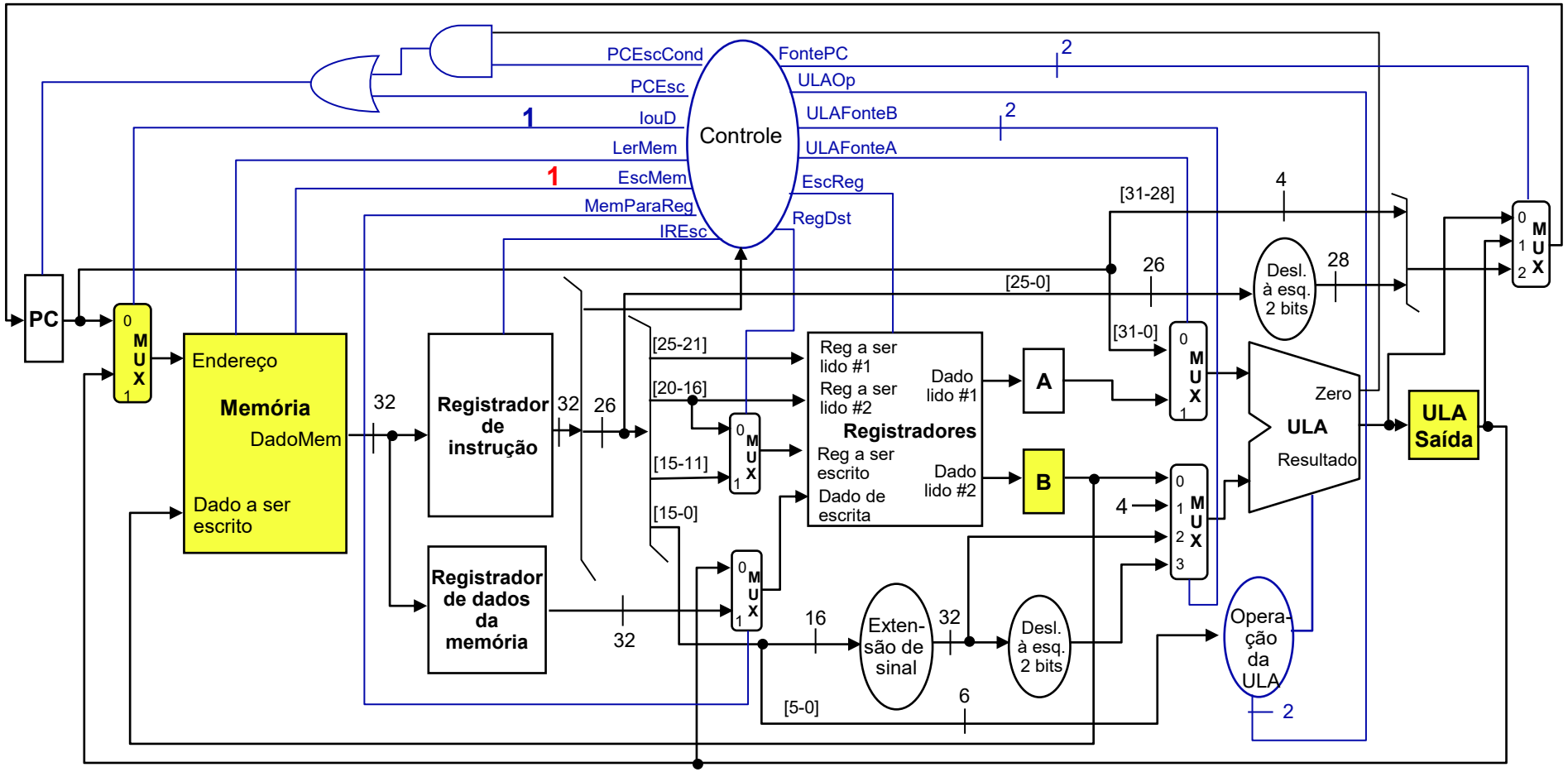
O Processador MIPS Multiciclo

Passos Necessários para Instrução sw no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	**4**	**5**	**4**	**3**	**3**

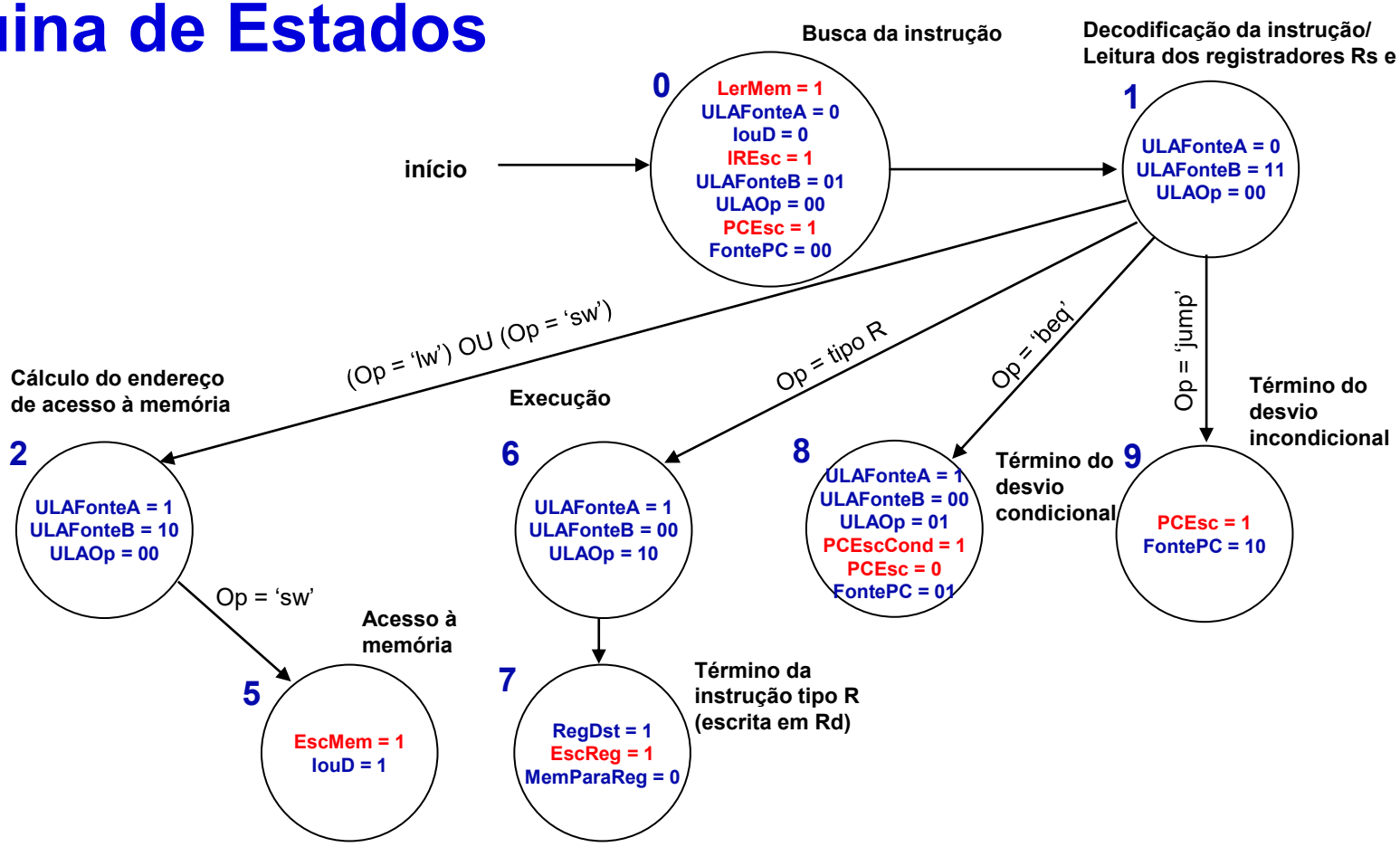
O Processador MIPS Multiciclo

Término da Instrução sw: escreve na memória



O Processador MIPS Multiciclo

Máquina de Estados



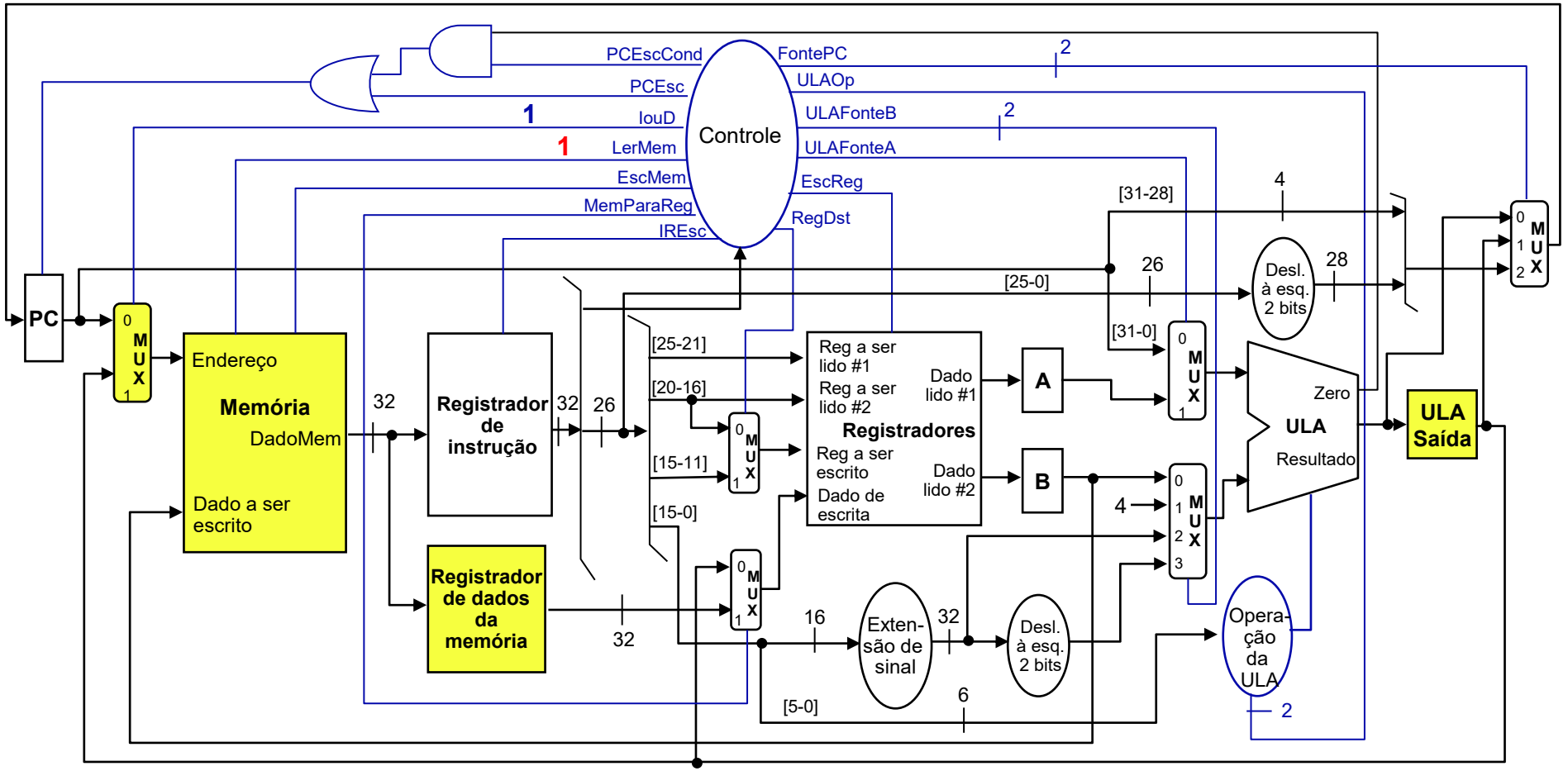
O Processador MIPS Multiciclo

Passos Necessários para Instrução lw no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	4	5	4	3	3

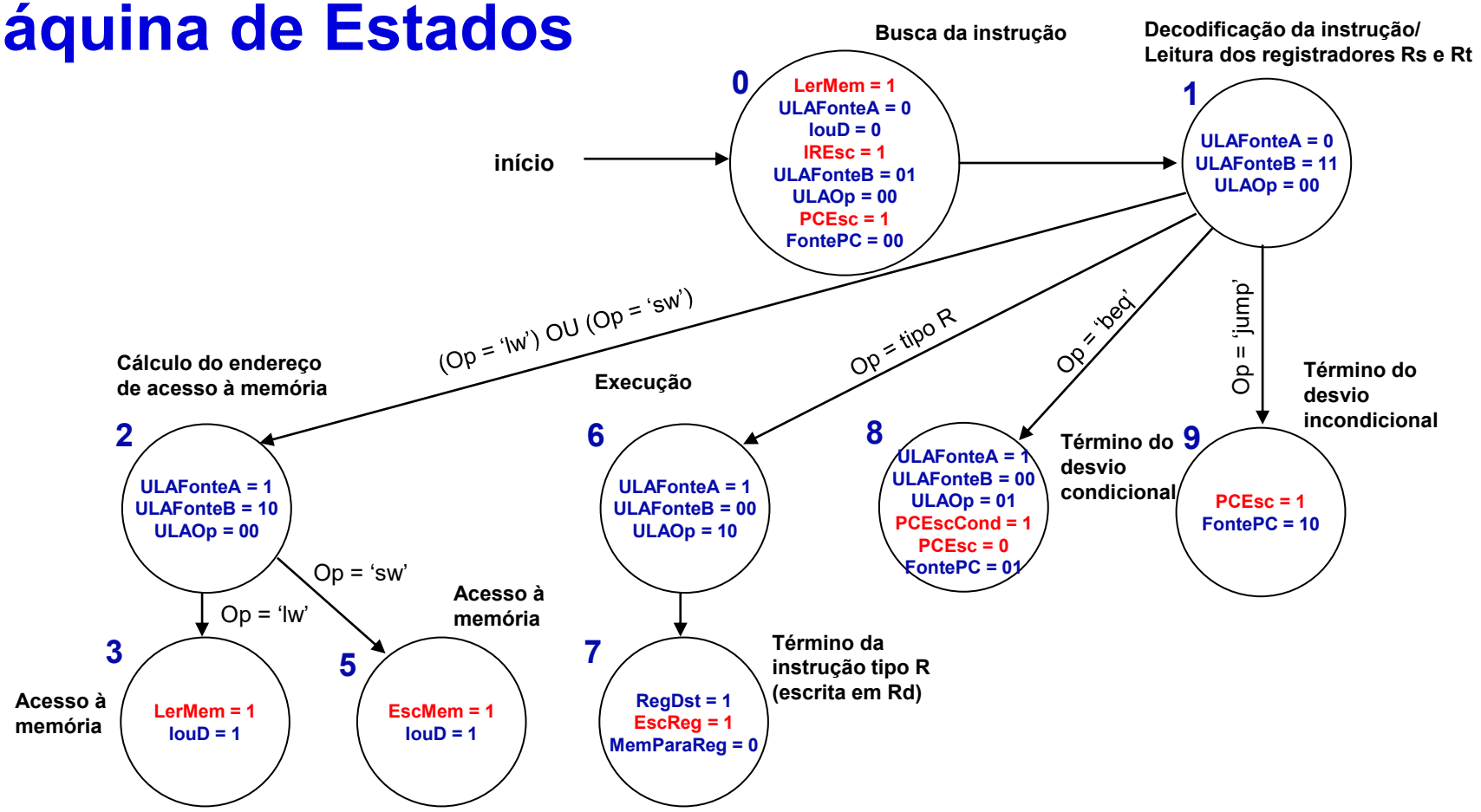
O Processador MIPS Multiciclo

Acesso à Memória na Instrução lw: lê dado da memória



O Processador MIPS Multiciclo

Máquina de Estados



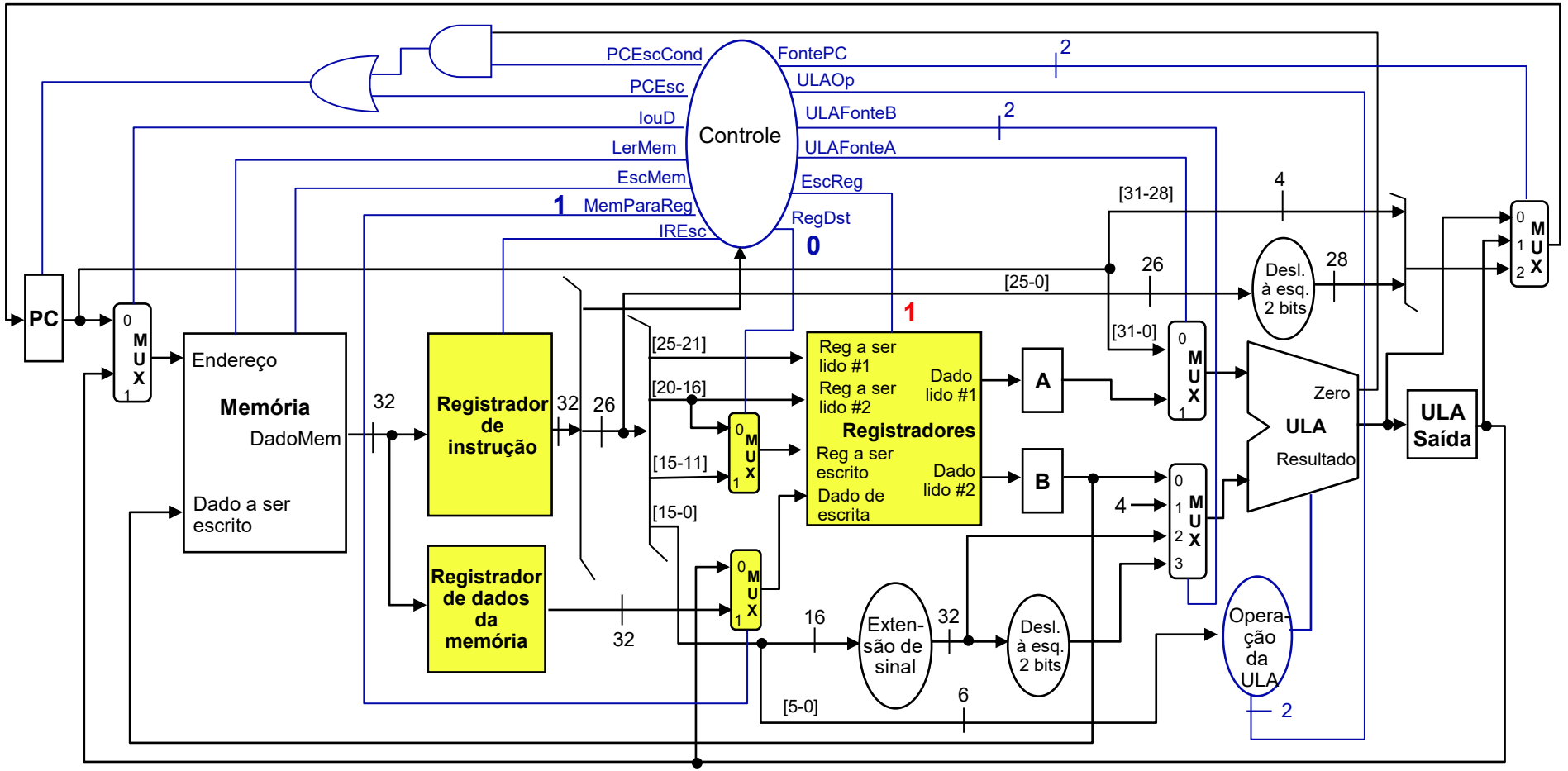
O Processador MIPS Multiciclo

Passos Necessários para Instrução lw no MIPS Multiciclo

Nome do passo	Instrução tipo R	Instrução lw	Instrução sw	Instrução beq	Instrução j
Busca da instrução	RI = Mem[PC] PC = PC + 4				
Decodificação da instrução & leitura dos registradores Rs e Rt & cálculo do endereço de desvio (cond.)	A = Reg [RI[25-21]] B = Reg [RI[20-16]] ULASaída = PC + (extensão de sinal(RI[15-0]) <<2)				
Execução, cálculo do endereço de acesso à memória, término de uma instrução branch/jump	ULASaída = A op B	ULASaída = A + extensão de sinal (RI[15-0])		Se (A == B) então PC = ULASaída	PC = PC[31-28] || (RI[25-0] <<2)
Término de uma instrução store word ou de tipo R	Reg [RI[15-11]] = ULASaída	RDM = Mem [ULASaída]	Mem [ULASaída] = B		
Término de uma instrução load word		Reg[RI[20-16]] = RDM			
Número de passos	4	5	4	3	3

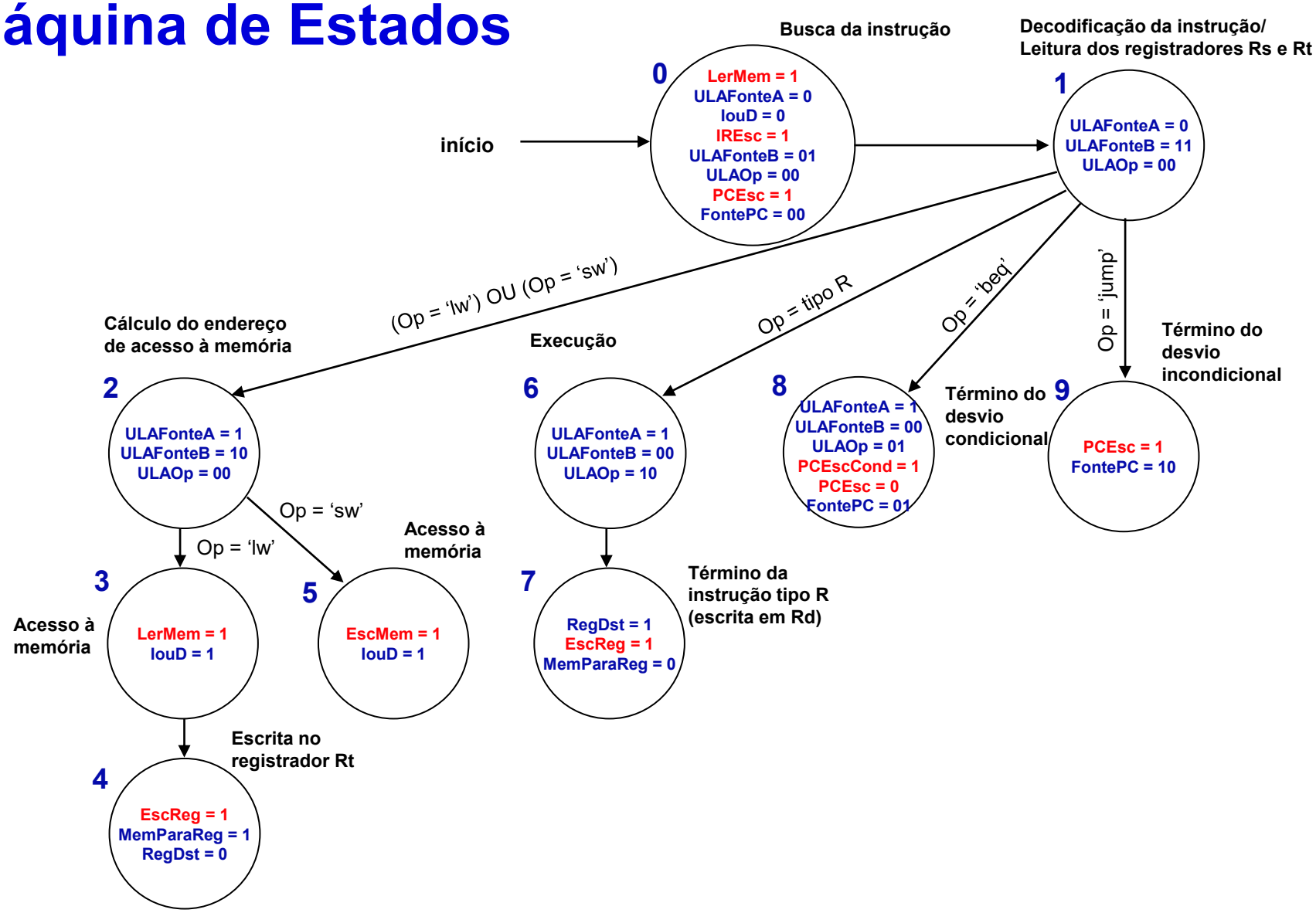
O Processador MIPS Multiciclo

Término da Instrução lw: escreve em registrador



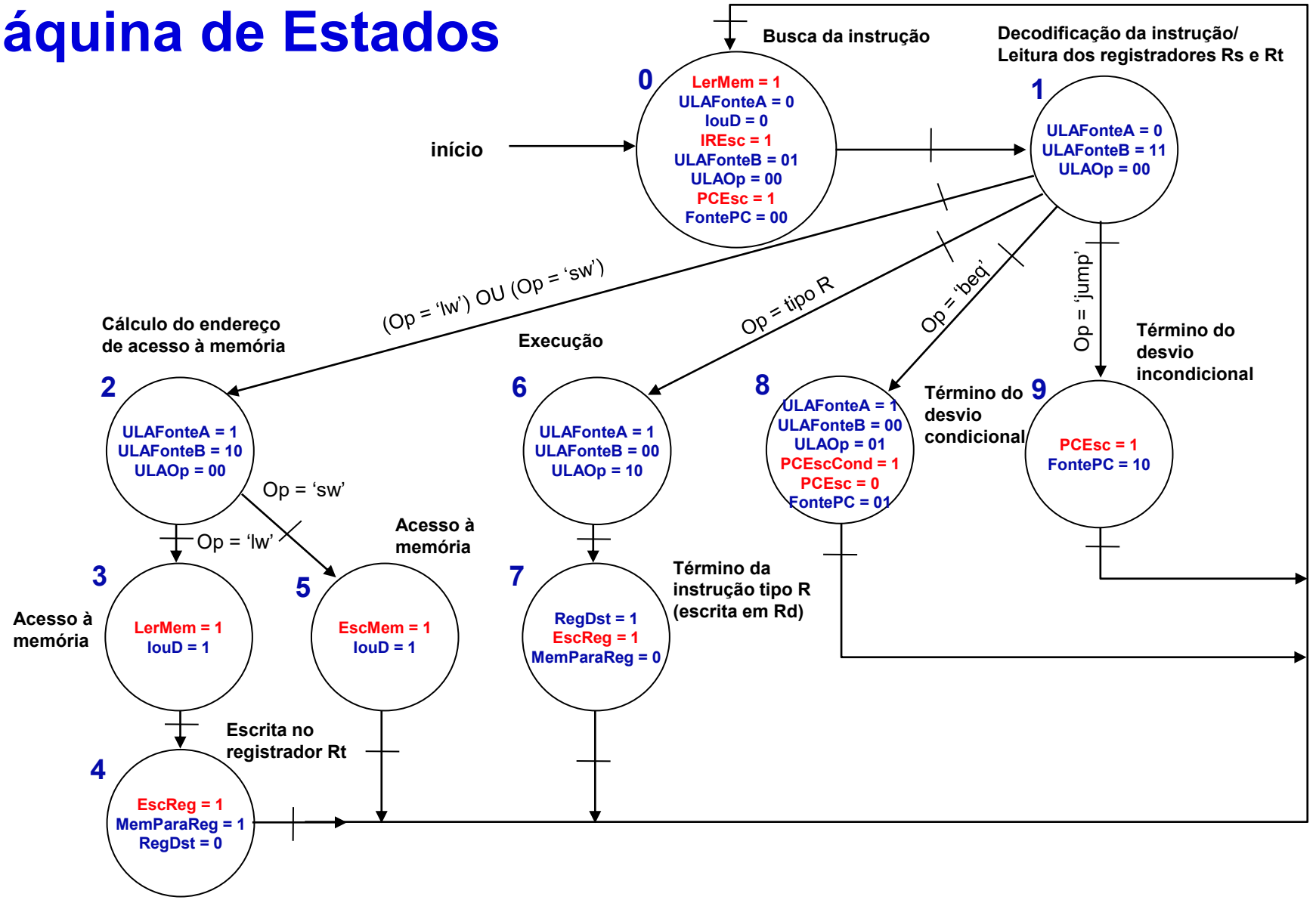
O Processador MIPS Multiciclo

Máquina de Estados



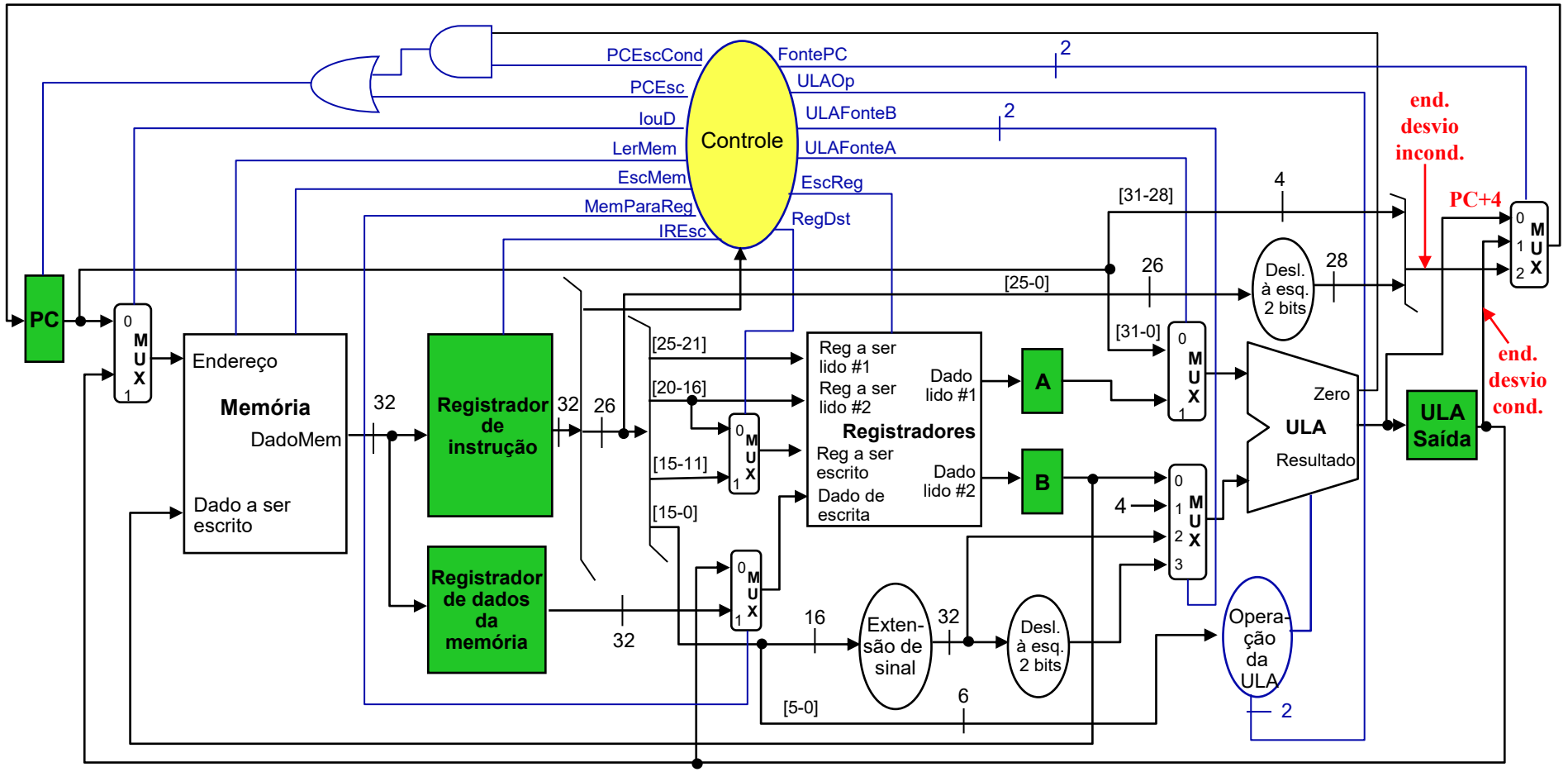
O Processador MIPS Multiciclo

Máquina de Estados



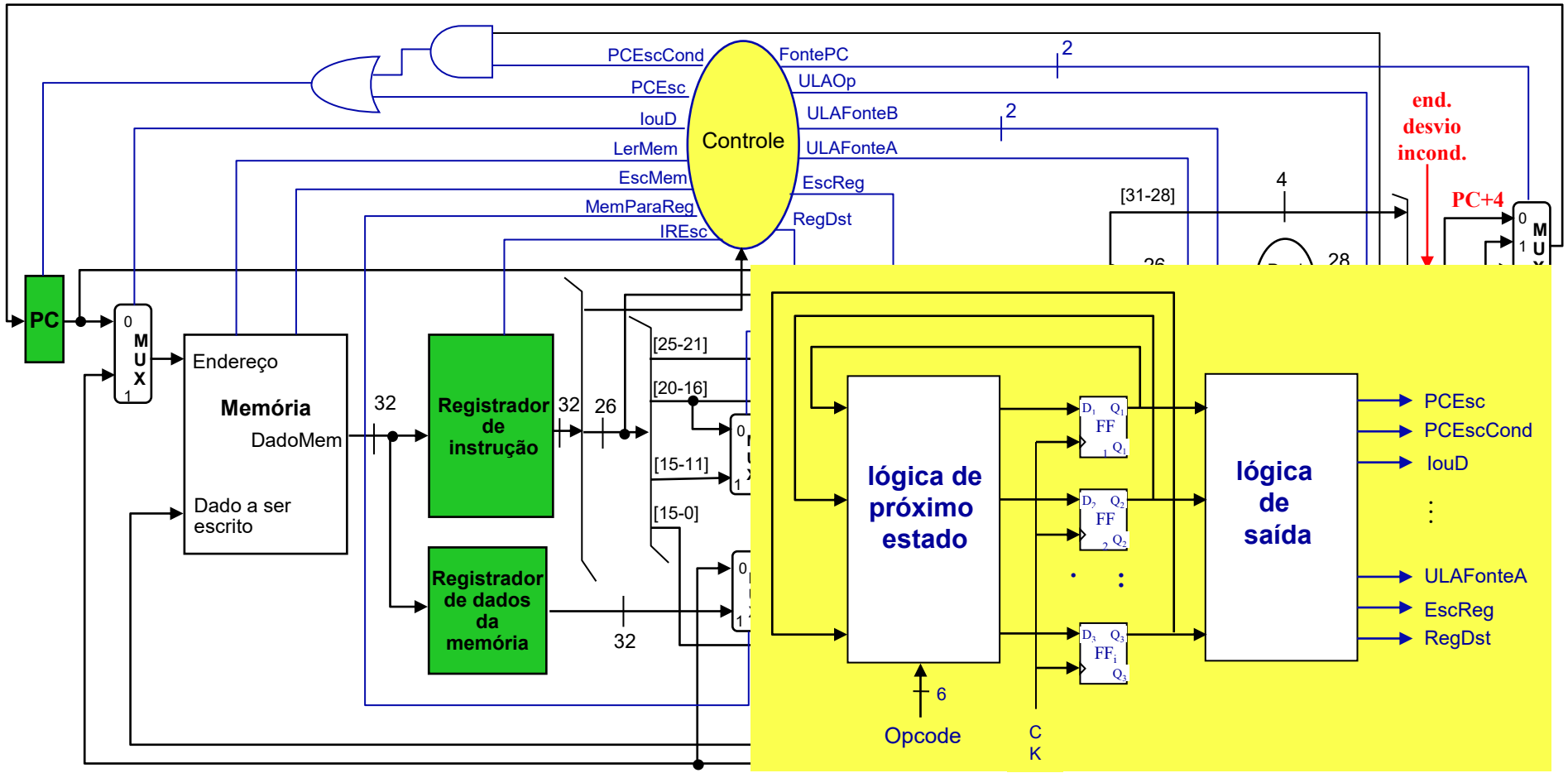
O Processador MIPS Multiciclo

Caminhos Críticos (para Estimar o Relógio)



O Processador MIPS Multiciclo

Caminhos Críticos (para Estimar o Relógio)

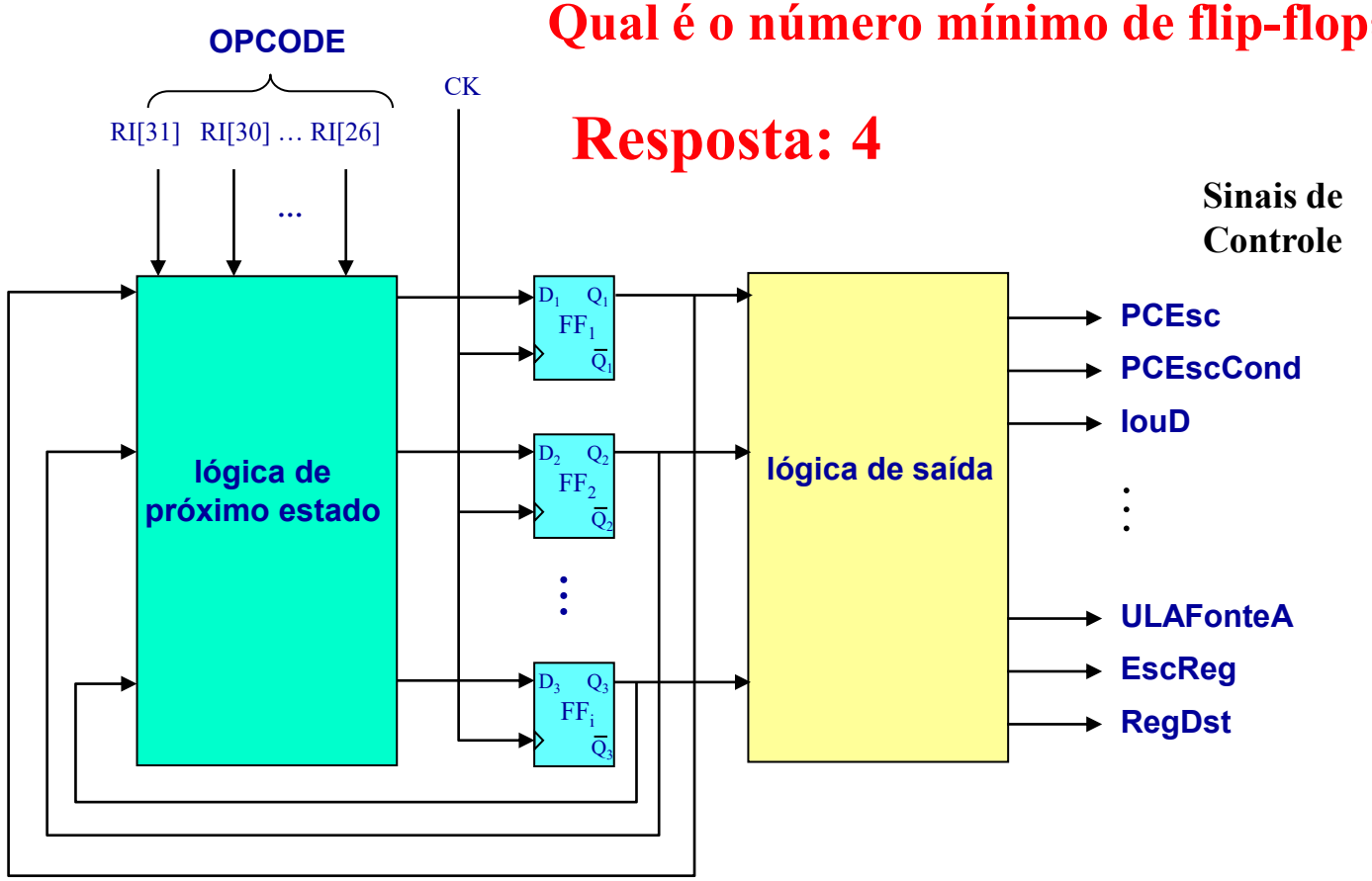


O Processador MIPS Multiciclo

Modelo de Máquina de Estados de Moore

Qual é o número mínimo de flip-flops?

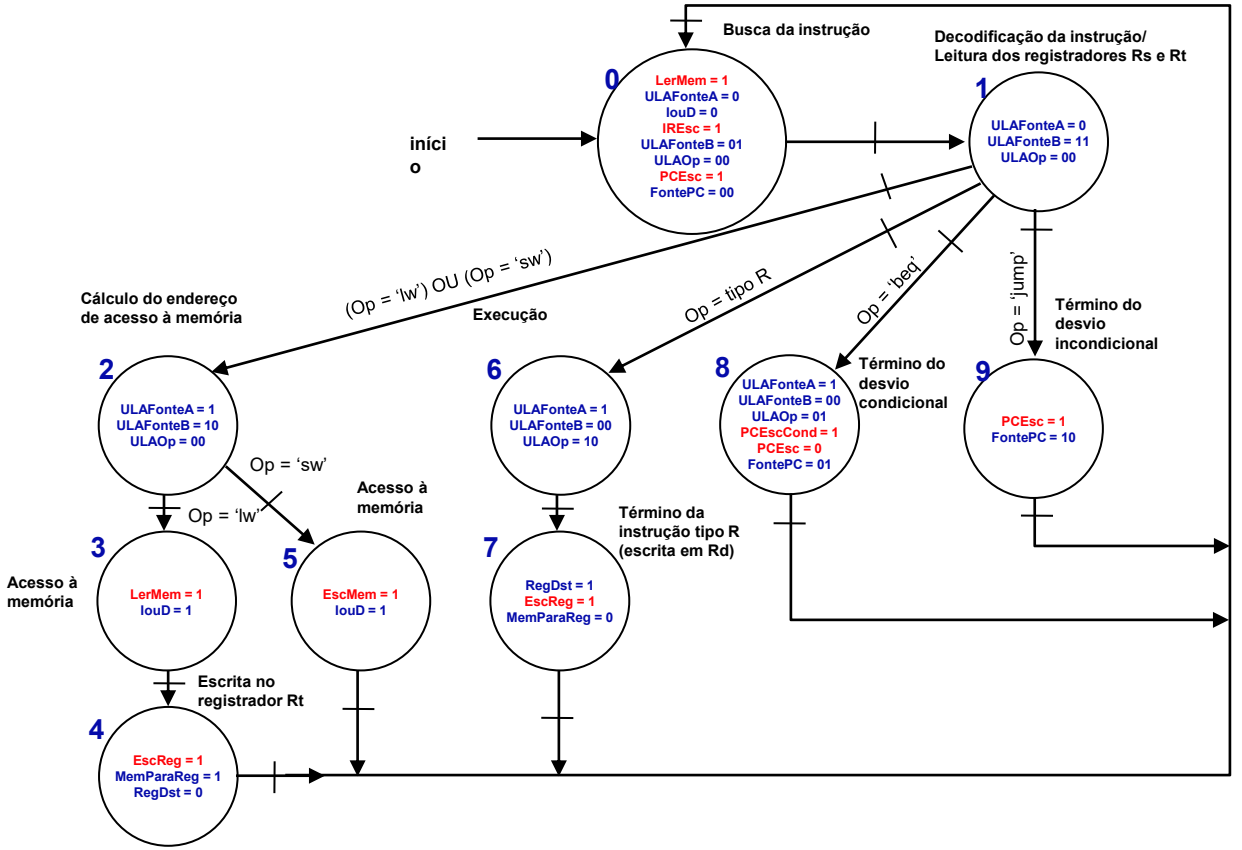
Resposta: 4



O Processador MIPS Multiciclo

Tabela de Transição de Estados

Estado Atual	Opcode	Próximo Estado
0	-	1
1	‘lw’ ou ‘sw’	2
	tipo R	6
	‘beq’	8
	‘j’	9
2	‘lw’	3
	‘sw’	5
3	-	4
4	-	0
5	-	0
6	-	7
7	-	0
8	-	0
9	-	0



O Processador MIPS Multiciclo

Tabela de Transição de Estados

Estado Atual	Opcode	Próximo Estado
0	-	1
1	'lw' ou 'sw'	2
	tipo R	6
	'beq'	8
	'j'	9
2	'lw'	3
	'sw'	5
3	-	4
4	-	0
5	-	0
6	-	7
7	-	0
8	-	0
9	-	0

Opcode = IR[31-26]:

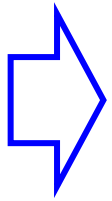
tipo R = 000000 (0)

lw= 100011 (35)

sw= 101011 (43)

beq= 000100 (4)

j= 000010 (2)



Estado Atual	Opcode	Próximo Estado
0	-	1
1	100011	2
	101011	2
	000000	6
	000100	8
	000010	9
2	100011	3
	101011	5
3	-	4
4	-	0
5	-	0
6	-	7
7	-	0
8	-	0
9	-	0

O Processador MIPS Multiciclo

Tabela de Transição de Estados

Estado Atual	Opcode	Próximo Estado
0	-	1
1	100011	2
	101011	2
	000000	6
	000100	8
	000010	9
2	100011	3
	101011	5
3	-	4
4	-	0
5	-	0
6	-	7
7	-	0
8	-	0
9	-	0

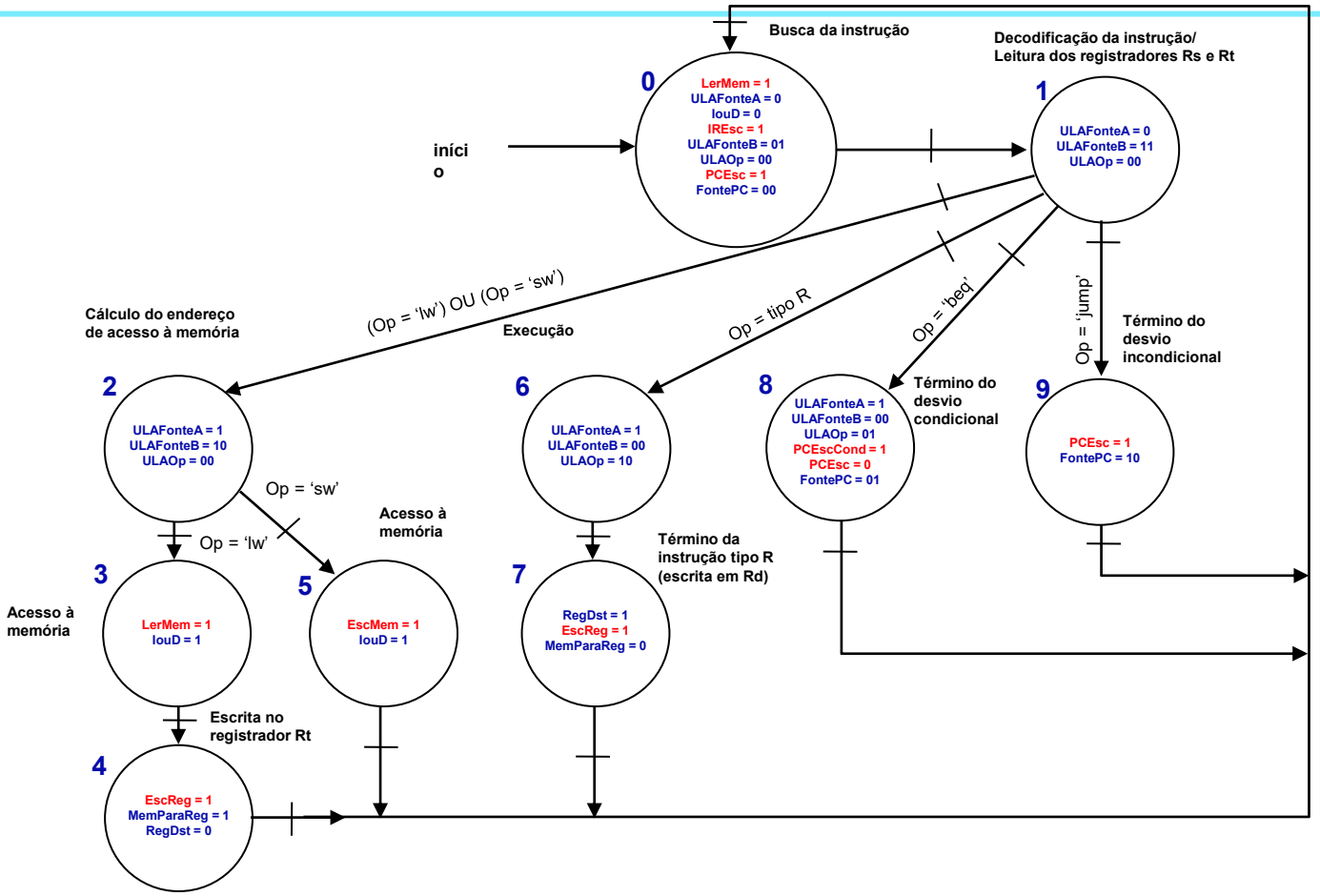
- Vars. do estado atual:
  Q3, Q2, Q1, Q0

- Vars. Do próximo estado:
  $Q3^+$, $Q2^+$, $Q1^+$, $Q0^+$

1. Codificar os estados
2. Substituir a codificação na tabela ao lado
3. Encontrar as equações minimizadas para as 4 saídas (variáveis de próximo estado)

O Processador MIPS Multiciclo

Tabela de Saída



Quando omitidos, sinais de controle devem valer '0', exceto:
Controles de muxes e ULAOp

O Processador MIPS Multiciclo

Usar a codificação do estado atual

Estado	PCEsc	PCEscCond	IouD	LerMem	EscMem	IREsc	MemParaReg	FontePC	ULAOp	ULAFonteB	ULAFonteA	EscReg	RegDst
0													
1													
2													
3													
4													
5													
6													
7													
8													
9													

O Processador MIPS Multiciclo

Exercício:

1. Determinar as equações de estado para o bloco de controle do MIPS multiciclo
2. Determinar as equações de saída para o bloco de controle do MIPS multiciclo

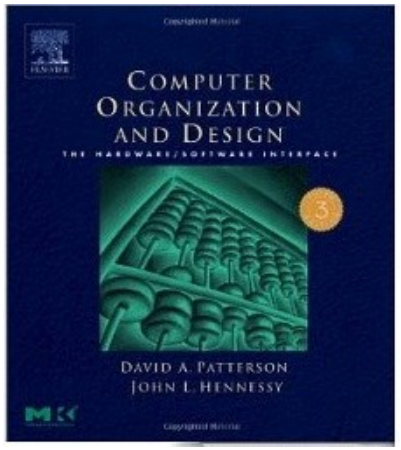
Responder os itens acima de duas formas:
- Sem necessariamente encontrar as equações mínimas
- Encontrando as equações mínimas com a ajuda de algum programa de minimização lógica (exemplo: Espresso de Berkeley)

O Processador MIPS Multiciclo

Leitura da Semana

Livro:

PATTERSON, David A.; HENNESSY, John L. "Computer Organization and Design: the hardware/software Interface", 3rd edition, Morgan Kaufmann Publishers, San Francisco, California, USA, 2007.



Se usar a 2ª Edição: Seção 5.4 e Apêndice B.
Se usar a 3ª Edição: Seção 5.5.