

UNIVERSIDADE DE AVEIRO
 DEPARTAMENTO DE ELECTRÓNICA, TELECOMUNICAÇÕES E INFORMÁTICA

Teste Teórico-Prático 2 - 17 de Junho de 2024 Duração: 1h25m

Nome: _____ Nº Mec. _____

Nota: Justifique **todas** as respostas.

1) Imagine que está em discussão a evolução do processador MIPS que estudou nesta cadeira para uma nova versão com 64 registos de 32 bits, mantendo o acesso à memória e a codificação das instruções em 32 bits (eventualmente adaptando o tamanho de alguns campos).

Quais as implicações diretas desta alteração no modo de codificar as instruções? Todos os tipos seriam afetados?

2) Considere o trecho de código *Assembly* do MIPS apresentado.

Endereço	Label	Assembly	Comentário
		.text	
0x00400000		la \$t0, array	# Instrução Virtual
		li \$t5, 3	
	for:	beq \$t5, \$0, endfor	
		sw \$t5, 0(\$t0)	
		addi \$t5, \$t5, -1	
		addiu \$t0, \$t0, 4	
		j for	
	endfor:	...	

Sabendo que o opcode da instrução **sw** é 43 e que o registo \$t0 é registo \$8, determine o código máquina da instrução (**sw \$t5, 0(\$t0)**).

31:26	25:21	20:16	15:11	10:6	5:0
=0x _____					

Rascunho:

3) A Figura 1 representa uma implementação básica do *datapath single cycle* do MIPS.

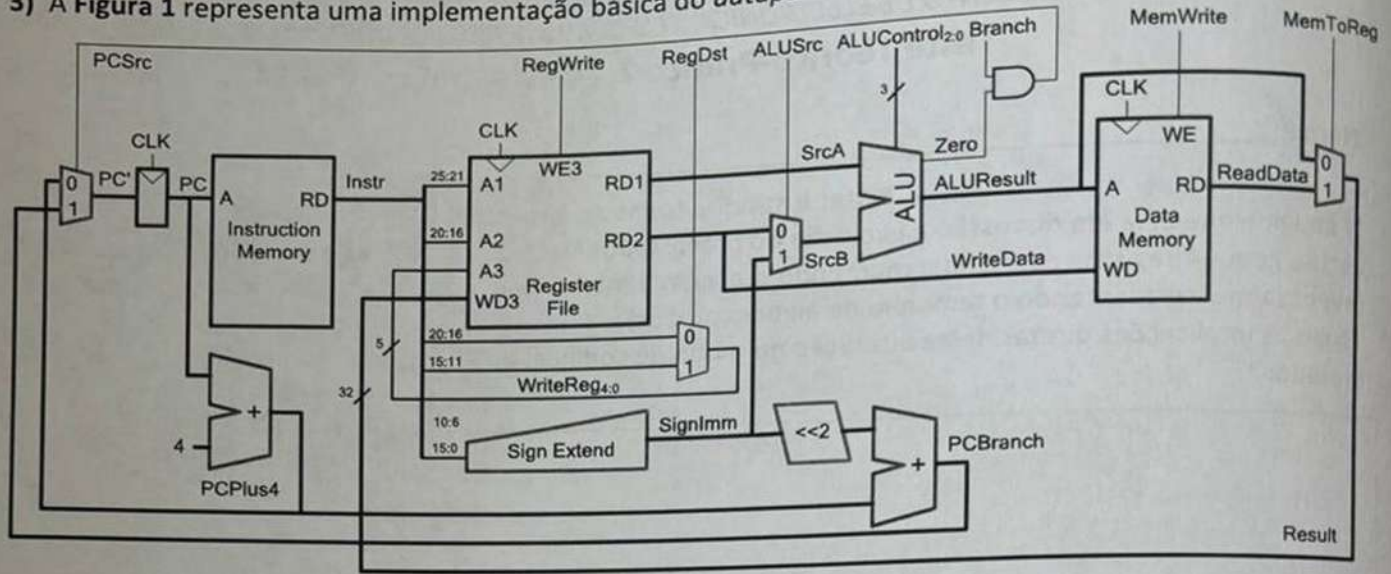


Figura 1 - Datapath single-cycle

a) Indique o valor dos sinais de controlo do *datapath* durante a execução da instrução:

```
beq $t5, $0, endfor
```

Sinal	Valores para <code>beq \$t5, \$0, endfor</code>
RegWrite	
RegDst	
ALUSrc	
ALUOp (ver tabela I, pág.4)	
Branch	
MemWrite	
MemToReg	

b) Indique o valor dos sinais de controlo durante a execução da instrução: `addi $8, $10, -1`

Sinal	Valores para <code>addi \$8, \$10, -1</code>
RegWrite	
RegDst	
ALUSrc	
ALUOp (ver tabela I, pág.4)	
Branch	
MemWrite	
MemToReg	

c) Assinale, na Figura 1, todos os caminhos e sinais ativos durante a execução da instrução "`addi $8, $10, -1`".

d) Sabendo que o opcode da instrução **addi** é 8₁₀, indique o código máquina da instrução:

addi \$8, \$10, -1

31:26	25:21	20:16	15:11	10:6	5:0
=0x _____					

Zona de Rascunho:

e) Adicione na **Figura 1** o que falta para suportar a execução da instrução **jal target** (*jump and link*). **Justifique/Descreva** as alterações efetuadas.

f) Sabendo que a função **target** está armazenada a partir do endereço 0x00400040, e que o opcode de **jal** é 3 preencha todos os campos de bits do quadro seguinte, para a instrução "**jal target**".

31:26	25:21	20:16	15:11	10:6	5:0
=0x _____					

Zona de Rascunho:

4) A Figura 2 representa uma implementação do *datapath multicyle* e a Figura 3 (na pag. 6) o diagrama de estados parcial do controlador respetivo.

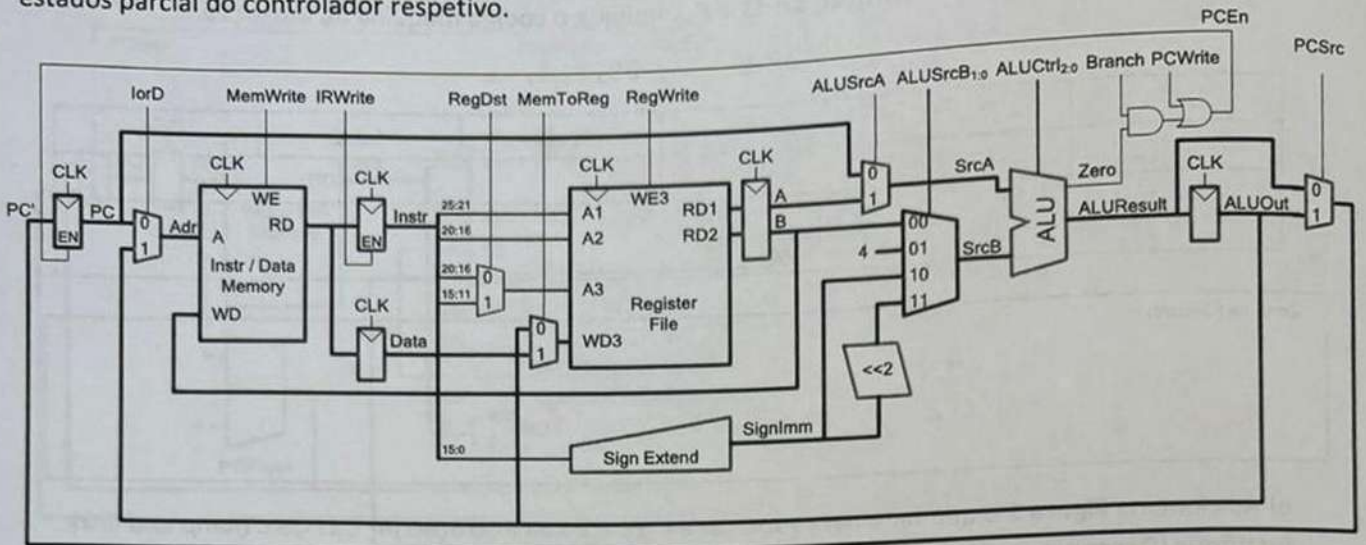


Figura 2 - Datapath multicyle

a) Modifique, se necessário, o *datapath* da Figura 2 para suportar a execução da instrução *xori*. Justifique a(s) alteração(ões) efetuada(s).

b) Indique, na Tabela III, as alterações necessárias no decodificador da ALU para suportar a execução da instrução *xori*.

Sugestão: Use uma das combinações disponíveis e não altere o número de bits de *ALUControl*.

ALUOp _{1:0}	Funct _{5:0}	ALUControl _{2:0}
00	XXXXXX	010 (Add)
01	XXXXXX	110 (Subtract)
10	100000 (add)	010 (Add)
10	100010 (sub)	110 (Subtract)
10	100101 (or)	001 (Or)
10	100110 (xor)	100 (Xor)
10	101010 (slt)	111 (SlT)

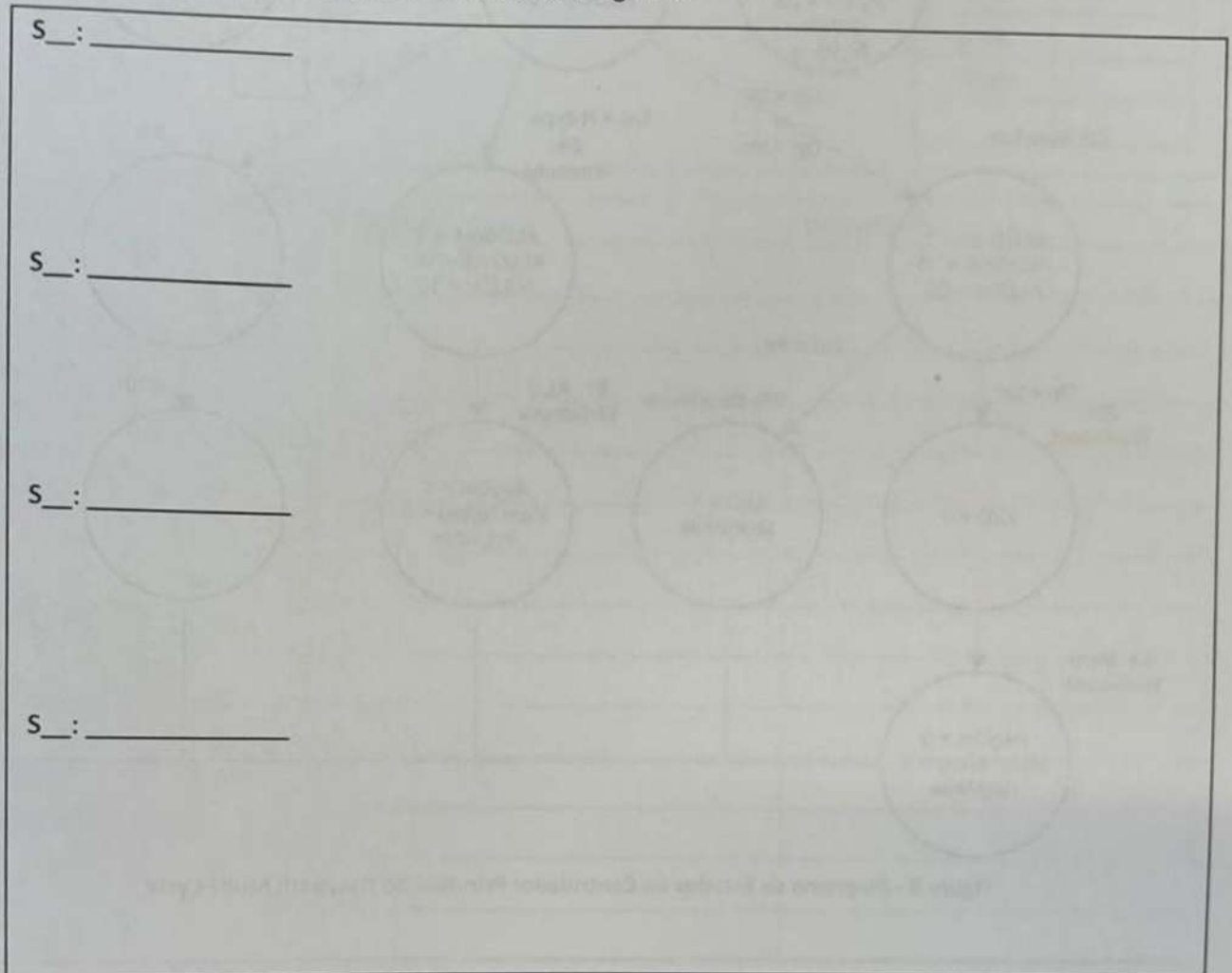
Tabela I - Decodificador da ALU, multi cycle.

Nome: _____ Nº Mec. _____

- c) Para executar a instrução **xori** é necessária alguma alteração no diagrama de estados da Figura 3? Caso considere necessário atualize o diagrama da Figura 3.

Identifique cada um dos estados percorridos pela unidade de controlo durante a execução da instrução **xori e explique por palavras as ações envolvidas em cada um deles.**

Sugestão: Consulte o diagrama de estados da Figura 3.



- d) Considere nesta alínea, os caminhos ativos e o valor dos sinais de controlo relevantes durante a fase DECODE, da instrução **xori**.

Preencha na tabela seguinte o valor que os sinais de controlo assumem durante esse estado e assinale na **Figura 2** os caminhos ativos nessa fase.

Instr.	Op _{5:0}	IorD	MemWrite	RegDst	MemToReg	RegWrite	AluSrcA	AluSrcB	Branch	PCWrite	PCSrc	ALUCtrl ¹
sll	000000											

¹ (ver tabela I, pág.4)