

UNIVERSIDADE DE AVEIRO

DEPARTAMENTO DE ELECTRONICA, TELECOMUNICAÇÕES E INFORMÁTICA

Teste Teórico 2 - 29 de Junho de 2020, PARTE II - Datapath

NOME: GONÇALO FERNANDES MACHADO

Nº Mec.: 98359

Duração: 1h00m

Notas:

1. Justifique **todas** as respostas.
2. Assinale as alterações com cores **diferentes**.

3) A Figura II.1 representa uma implementação básica do datapath do MIPS.

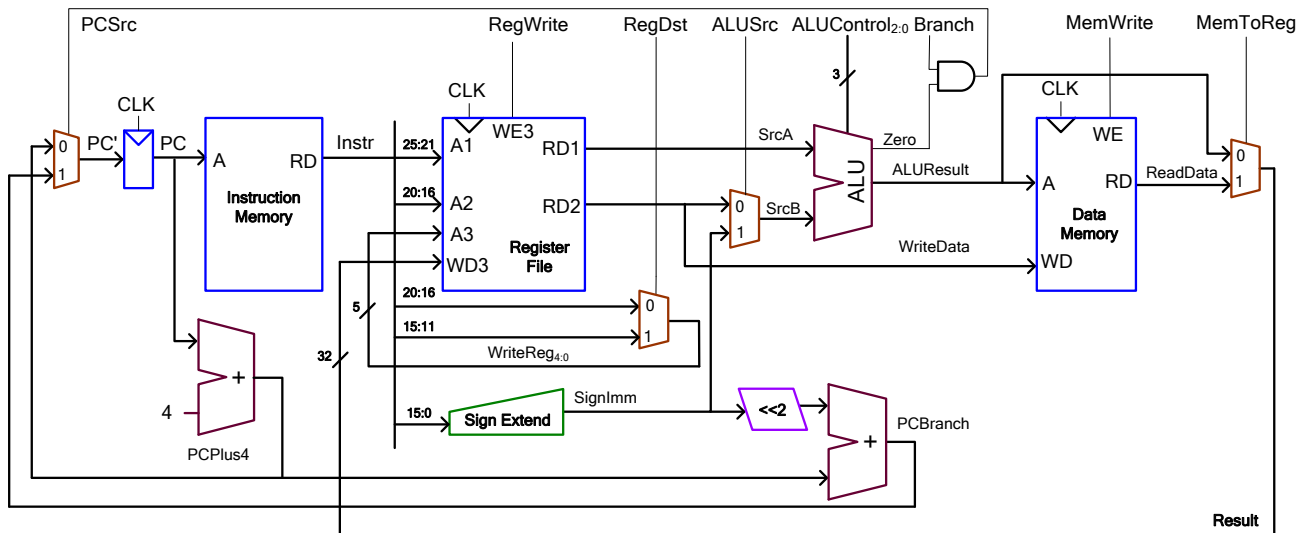


Figura II.1 - Datapath single-cycle

a) Adicione à Figura II.1 o que falta para que o datapath suporte a execução da instrução **srl** (Shift Right Logical).

b) Assinale todos os caminhos e sinais activos durante a execução da instrução.

c) Acrescente à Tabela de Verdade da **Tabela II.1** o valor das entradas e saída $ALUControl_{2:0}$ relativas à instrução **srl**. O valor do $FunCode_{5:0}$ de **srl** é igual a 2. (Sugestão: Use uma das combinações disponíveis para não ter de aumentar o número de bits de saída).

$ALUOp_{1:0}$	$Funct_{5:0}$	$ALUControl_{2:0}$
00	XXXXXX	010 (Add)
01	XXXXXX	110 (Subtract)
10	100000 (add)	010 (Add)
10	100010 (sub)	110 (Subtract)
10	100100 (and)	000 (And)
10	100101 (or)	001 (Or)
10	101010 (slt)	111 (SlT)

Tabela II.1 - Descodificador da ALU

d) Preencha todos os campos de bits da **Tabela II.2**, para a instrução "**srl \$t1, \$t0, 28**"; o número dos registos \$t0 e \$t1 é igual a 8 e 9, respetivamente.

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

Tabela II.2 - Instrução srl rd, rt, shamt

4) A **Figura II.2** representa uma implementação melhorada do *datapath* do MIPS, com suporte para a execução das instruções "**jal label**" e "**jr rs**".

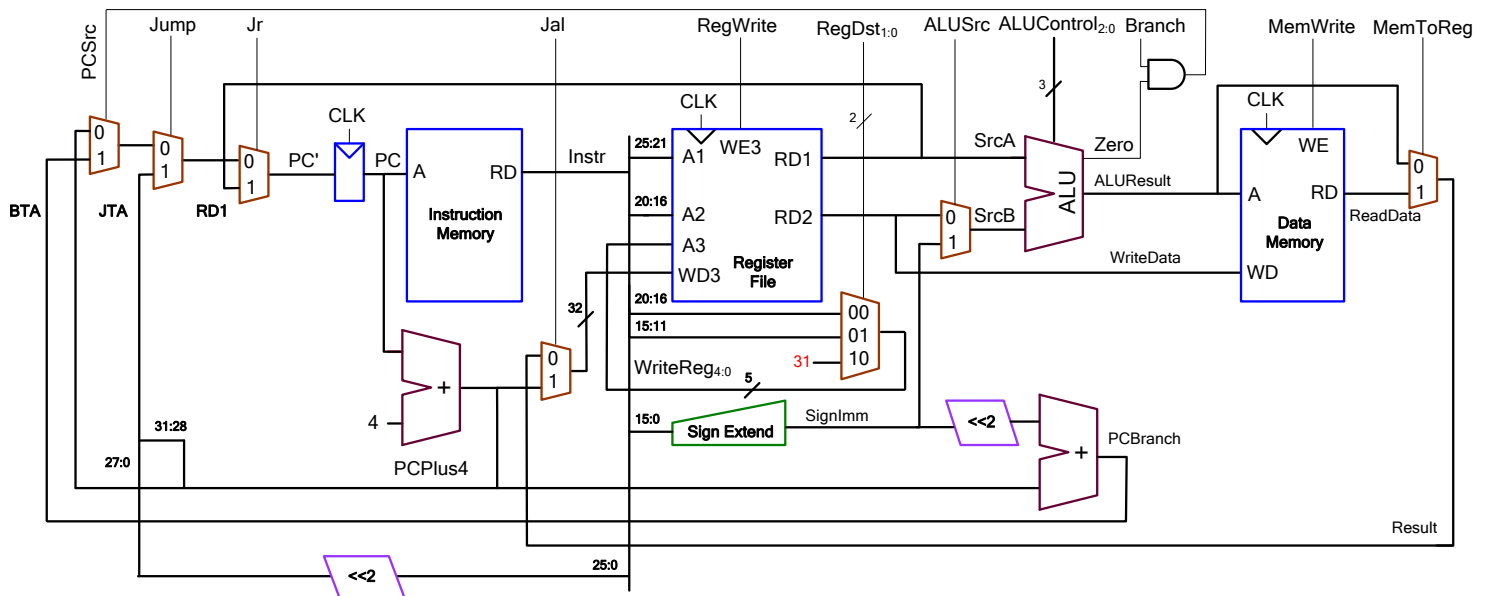


Figura II.2 - Datapath single-cycle com Jal e Jr

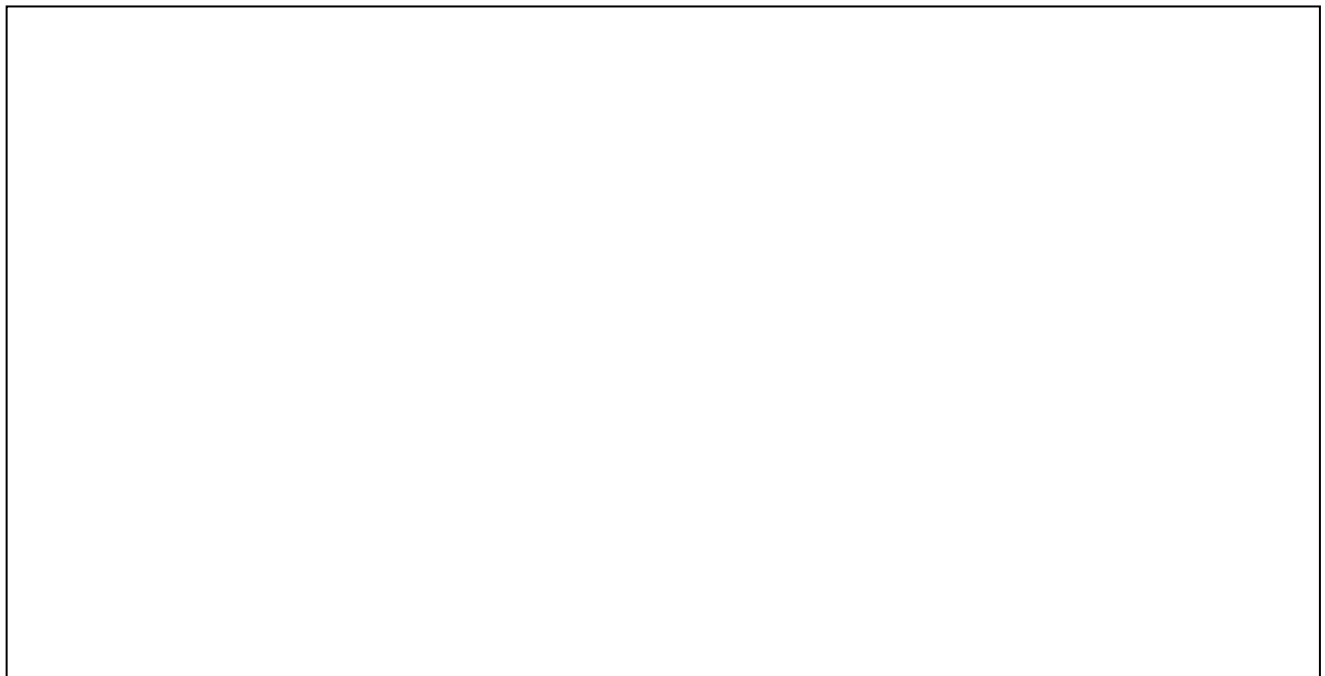
A instrução "**jalr rs**" é uma instrução do tipo-R que junta a funcionalidade das instruções **jal** e **jr**. Isto é, executa um 'salto' para o endereço especificado no registo **rs** (como a instrução **jr**) e guarda no registo **\$ra** o endereço de retorno (como a instrução **jal**). O formato da instrução está indicado na **Tabela II.3**.

31:26	25:21 (rs)	20:16 (rt)	15:11 (rd)	10:6	5:0
0	rs	00000	11111	00000	001001

Tabela II.3 - Instrução jalr rs

Para que o *datapath* suporte esta instrução, a Unidade de Controlo deve gerar um sinal de controlo, **jalr**, possibilitando o armazenamento, no Banco de Registos, do valor do endereço de retorno (**PC4**) no registo **\$ra** (31).

a) Admitindo que a Unidade de Control gera o sinal **jalr**, modifique o *datapath* da **Figura II.2**, por forma a que suporte adicionalmente a execução da instrução "**jalr rs**". Assinale essa modificação na figura.



b) Indique na **Tabela II.4**, o valor dos sinais de controlo para os códigos de função das instruções **jr** e **jalr**.

ALUOp _{1:0}	Funct _{5:0}	ALUControl _{2:0}	jr	jalr
00	XXXXXX	010 (Add)	0	0
01	XXXXXX	110 (Subtract)	0	0
10	100000 (add)	010 (Add)	0	0
10	100010 (sub)	110 (Subtract)	0	0
10	0	0
10	001000 (jr)			
10	001001 (jalr)			

Tabela II.4 - Descodificador da ALU com suporte para 'jr' e 'jalr'

5) A **Figura II.3** representa uma implementação do *datapath multicyle* e a **Figura II.4** o diagrama de estados parcial do controlador respetivo.

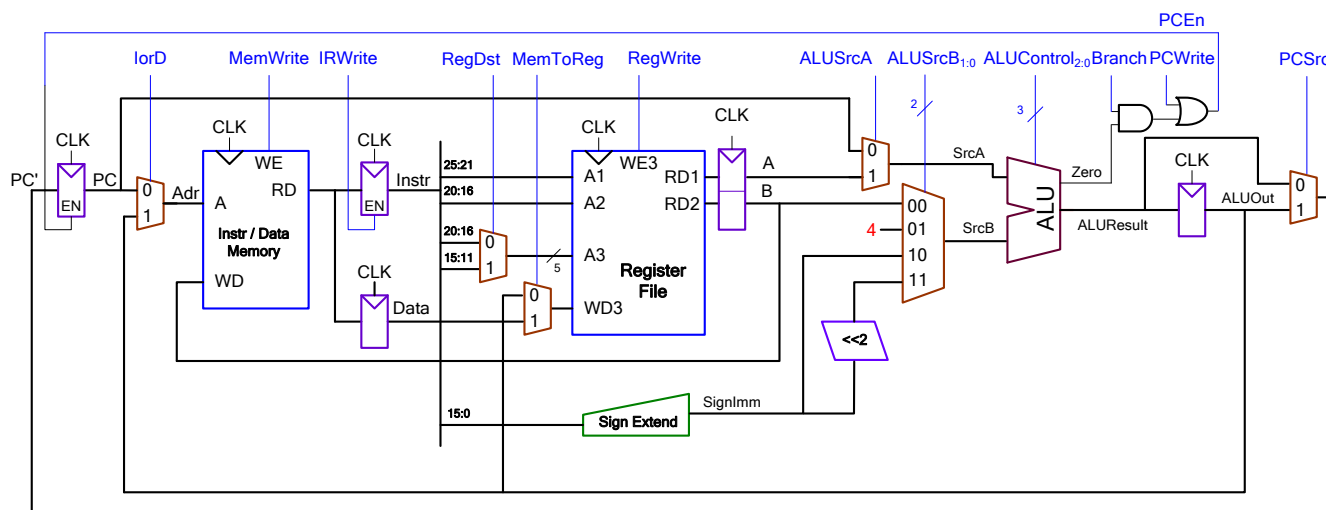


Figura II.3 - Datapath multicyle

a) Modifique o *datapath* da **Figura II.3** para suportar a execução da instrução **andi**. Indique os caminhos ativos e o valor dos sinais de controlo relevantes durante a execução. Explique por palavras as ações envolvidas na sequência de estados.

b) Indique, na **Tabela II.5**, as alterações necessárias.

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	100100 (and)	000 (And)
10	100101 (or)	001 (Or)
10	101010 (slt)	111 (SlT)

Tabela II.5 - Descodificador da ALU

c) No diagrama de estados da **Figura II.4** preencha os estados **S9** e **S10** com os valores dos sinais de seleção dos *multiplexers* e de *enable*. Use o decodificador da ALU da **Tabela II.5**.

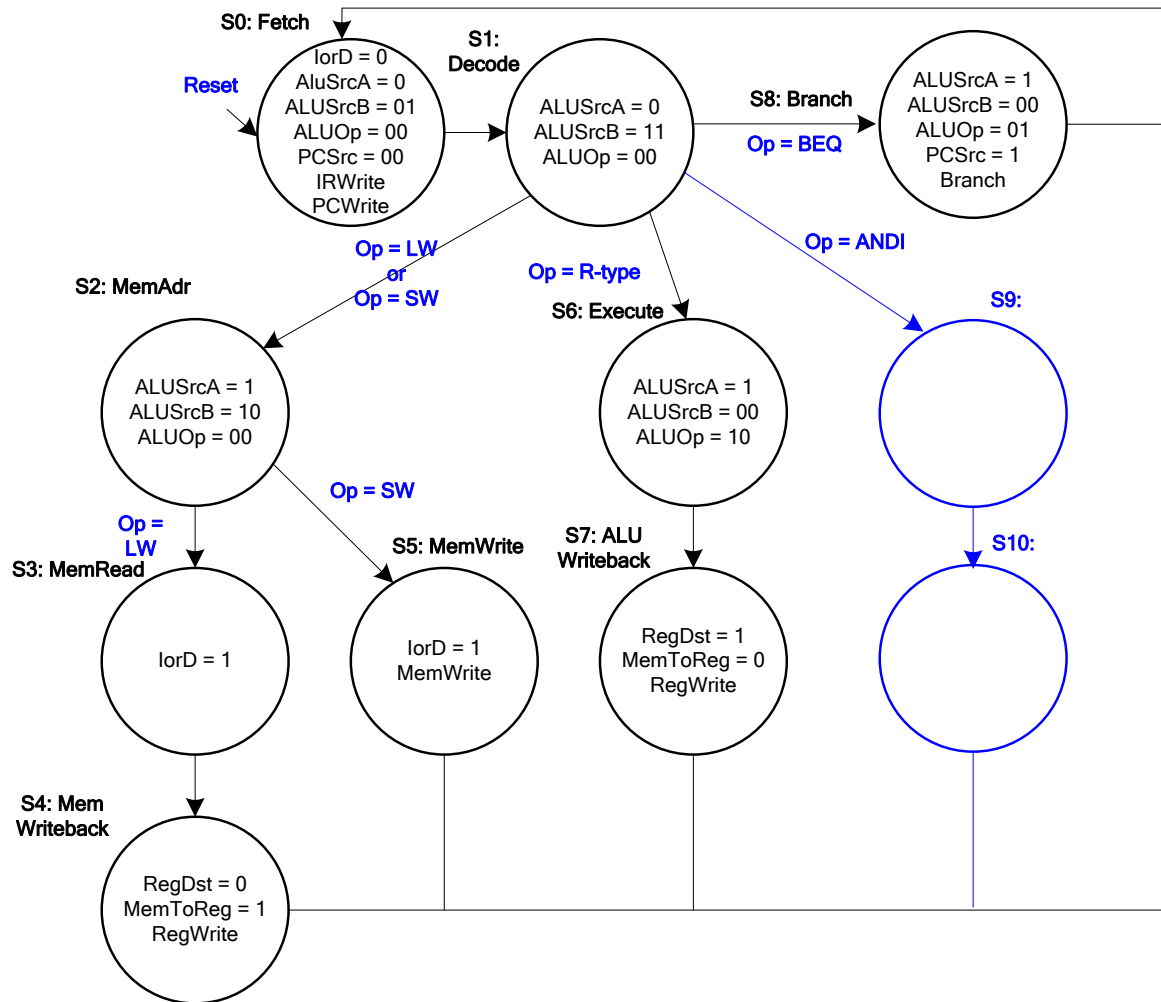


Figura II.4 - Diagrama de Estados do Controlador Principal