

2a) Função - número de minúsculas

```
int nr_minus (char *s) {
int mins=0;
    while ( *s != '\0' ) {
        if( ( *s >= 'a' ) && ( *s <= 'z' ) ) mins++;
        s++;
    }
    return mins;
}
```

Variável	Registro(s)
s	
*s	
mins	

```
int nr_minus(char *s);

void main( void ) {
    static char* msg = "\n-->Teste2 de IAC, ex2b<--" ;
    static char * minus = "\n0 nr de minusculas e: " ;

    print_str( msg );
    print_str( minus );
    print_int10( nr_minus( msg ) );
    exit();
}
```

[illegible]

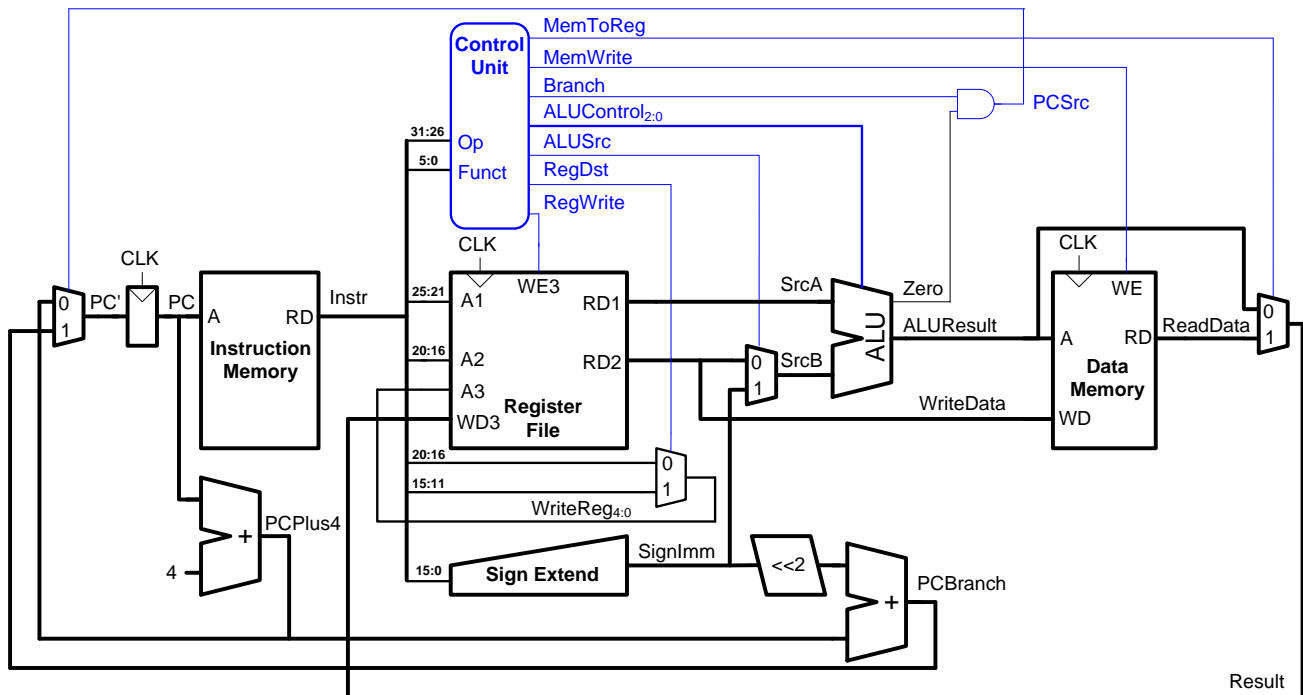
PI: 1) 7; 2a) 4,5; 2b) 2,5; PII: 3a) 0,5; 3b) 0,5; 3c) 1; 3d) 1; 4a) 0,5; 4b) 0,5; 4c) 0,5; 5a) 0,5; 5b) 0,5; 5c) 0,5.

[illegible]

PI: 1) 7; 2a) 4,5; 2b) 2,5; PII: 3a) 0,5; 3b) 0,5; 3c) 1; 3d) 1; 4a) 0,5; 4b) 0,5; 4c) 0,5; 5a) 0,5; 5b) 0,5; 5c) 0,5.

Teste Teórico 2 - 20 de Junho de 2018, PARTE II - Datapath

Nome: _____ Nº Mec. _____

3) A Figura II.1 representa uma implementação do *datapath* do MIPS.**Figura II.1 - Datapath single-cycle**

Considere o trecho de código Assembly do Programa II.1.

Em todas as alíneas da questão 3, considere sempre a instrução "**beq \$t0, \$t1, ewh**".

O número dos registos \$t0 e \$t1 é igual a 8 e 9, respetivamente.

```

.text
main: li    $t0, 0
      li    $t1, 5
wh:   beq   $t0, $t1, ewh
      addi  $t0, $t0, 1
      j     wh
ewh:  beq   $0, $0, ewh

```

Programa II.1a) Sendo o $OpCode_{5:0}$ de **beq** igual a 4, preencha todos os campos de bits da Tabela II.1.

31:26	25:21	20:16	15:8	7:0

Tabela II.1 - Instrução beq rs, rt, im16b) Sendo o endereço de *main* igual a 0x00400000 determine o valor do endereço-alvo (**Branch Target Address - BTA**) em função do valor de PC+4 e do valor *im16* da instrução.

c) Marque na Figura II.1 os caminhos de dados e os sinais de controlo ativos durante a execução da instrução.

d) Registe na Tabela II.2 o valor dos sinais de controlo durante a execução da instrução. Considere o decodificador da ALU apresentado na Tabela II.3.

Instruction	Op _{5:0}	RegWrite	RegDst	AluSrc	Branch	MemWrite	MemToReg	ALUOp _{1:0}
R-Type	000000	1	1	0	0	0	0	10
beq	000100							

Tabela II.2 - Decodificador Principal

Cotação:

PI: 1) 7; 2a) 4,5; 2b) 2,5; PII: 3a) 0,5; 3b) 0,5; 3c) 1; 3d) 1; 4a) 0,5; 4b) 0,5; 4c) 0,5; 5a) 0,5; 5b) 0,5; 5c) 0,5.

4) Considere de novo o *datapath* da **Figura II.1**.

a) Modifique o *datapath* por forma a suportar a execução da instrução "**ori \$t1, \$t0, im16**", assinalando as alterações na **Figura II.2**. Explique.

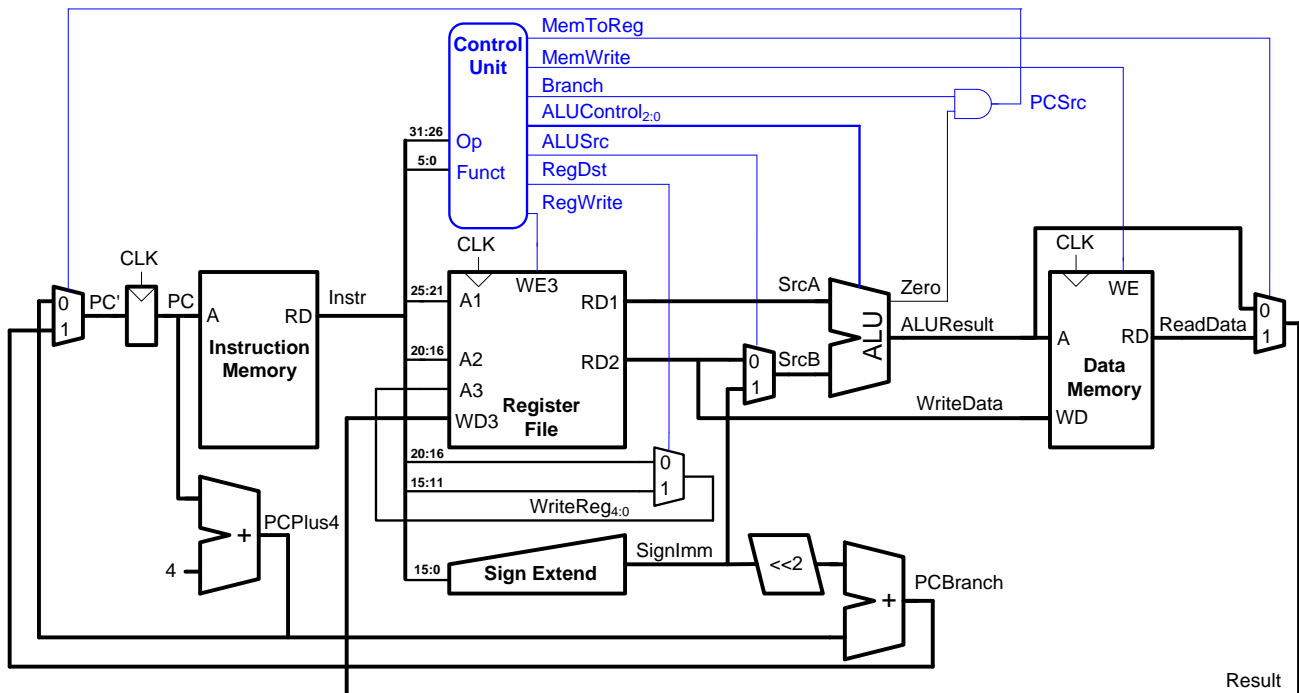


Figura II.2

b) Adapte a **Tabela II.4**, indicando as alterações para incluir suporte para a instrução **ori**.

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.3 - Decodificador da ALU

c) Sabendo que o *OpCode*_{5:0} da instrução **ori** é 13, assinale na **Tabela II.4** as alterações e o valor dos sinais de controlo durante a execução da instrução. Considere o decodificador da ALU apresentado na **Tabela II.3**.

Instruction	Op _{5:0}	RegWrite	RegDst	AluSrc	Branch	MemWrite	MemToReg	ALUOp _{1:0}
R-Type	000000	1	1	0	0	0	0	10
ori	001101							

Tabela II.4 - Decodificador Principal

Cotação:

PI: 1) 7; 2a) 4,5; 2b) 2,5; PII: 3a) 0,5; 3b) 0,5; 3c) 1; 3d) 1; 4a) 0,5; 4b) 0,5; 4c) 0,5; 5a) 0,5; 5b) 0,5; 5c) 0,5.

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

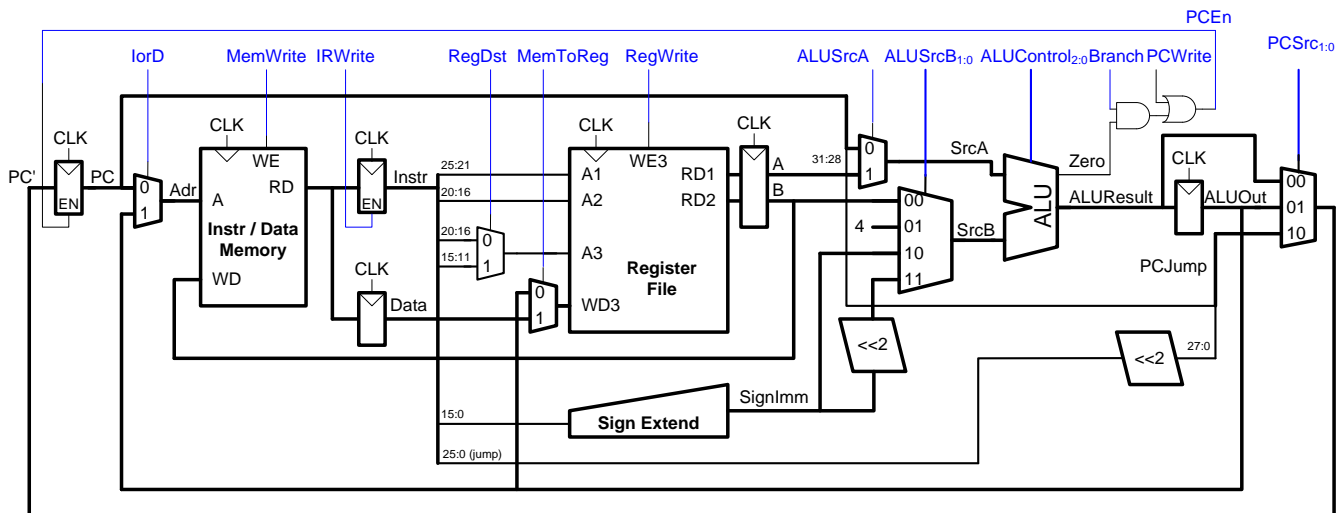


Figura II.3 - Datapath multicyle

- a) Indique na **Figura II.3** os caminhos ativos e os sinais de controlo durante a execução da instrução **j(ump)**.
b) Quais as modificações necessárias para que o *datapath* possa executar a instrução **jal**? Explique.

c) No diagrama de estados ao lado, preencha o estado **S12 (Jal)** com os valores dos sinais de seleção dos *multiplexers* e de *enable* dos registos. Justifique.

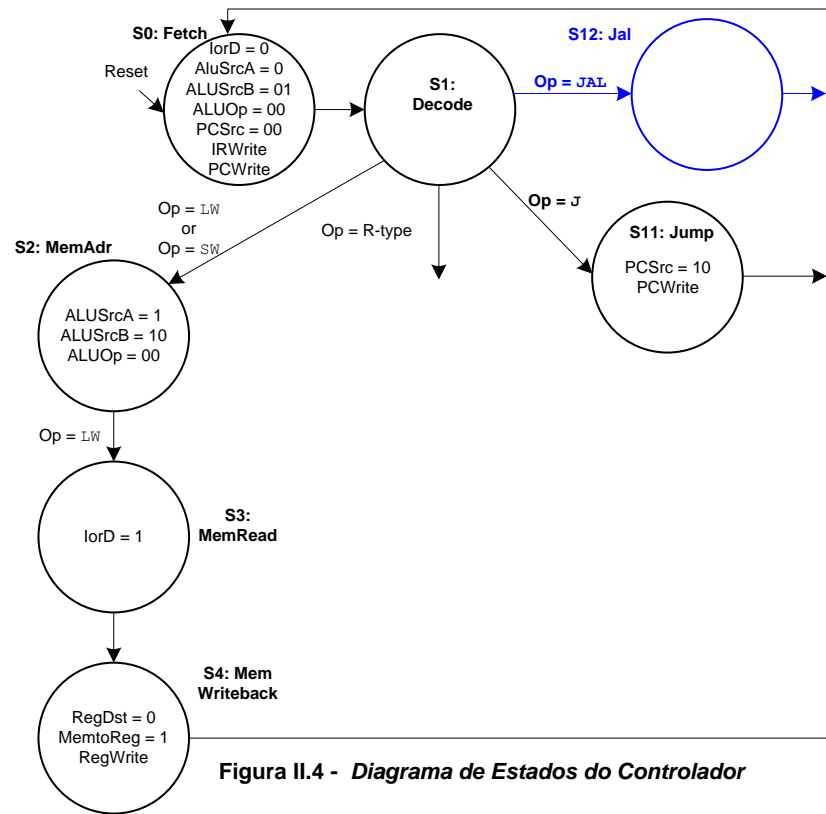


Figura II.4 - Diagrama de Estados do Controlador

Cotação:

PI: 1) 7; 2a) 4,5; 2b) 2,5; PII: 3a) 0,5; 3b) 0,5; 3c) 1; 3d) 1; 4a) 0,5; 4b) 0,5; 4c) 0,5; 5a) 0,5; 5b) 0,5; 5c) 0,5.

Parte II - Área de rascunho

Cotação:

PI: 1) 7; **2a)** 4,5; **2b)** 2,5; **PII: 3a)** 0,5; **3b)** 0,5; **3c)** 1; **3d)** 1; **4a)** 0,5; **4b)** 0,5; **4c)** 0,5; **5a)** 0,5; **5b)** 0,5; **5c)** 0,5.