

## PRIMEIRO EXERCÍCIO INDIVIDUAL EM CLASSE

### EA078A – Primeiro de Semestre de 2007

(1,5) 1) Considere o número  $(0.625)_{\text{dec}}$ . Mostre qual será a sua representação em ponto flutuante (precisão simples – padrão IEEE 754 com bias). Usando o algoritmo discutido em aula, faça a multiplicação desse número por ele mesmo. Mostre o resultado final na mesma representação em ponto flutuante.

(3,0) 2) Considere o caminho de dados para um único ciclo visto em aula (e mostrado na Fig. 1) e responda às seguintes questões (estruture bem a resposta):

a) Por que, na execução de uma instrução tipo R, o sinal RegDst precisa ser igual a 1, o sinal OrigALU precisa ser igual a zero e o sinal MemparaReg precisa ser igual a zero? Nessa instrução, de que forma se determina a ação a ser tomada pela ALU?

b) Por que, na execução de um load, o sinal RegDst precisa ser igual a 0, o sinal OrigALU precisa ser igual a 1 e o sinal MemparaReg precisa ser igual a 1? Que operação realiza a ALU numa instrução desse tipo?

c) Por que, na execução de um branch, é preciso fazer uma composição entre um sinal 'Branch' e o sinal 'Zero' da ALU para que se determine qual endereço deve ser passado ao PC?

d) Por que, na execução de um store, o bit MemparaReg é um "don't care"? Por que o bit "EscreveMem" vale 1? Que operação fará a ALU nessa instrução?

(1,5) 3) Discorra sobre a veracidade / falsidade da seguinte afirmação (estruture bem a resposta):

O único limitante para a "frequência de operação" de um caminho de dados é a tecnologia disponível para a construção de um sinal de clock suficientemente rápido.

(2,0) 4) Suponha que você esteja analisando duas máquinas, A e B, com a mesma arquitetura do conjunto de instruções. Essas máquinas executam uma mesma sequência, e verifica-se que  $CPI_A = 2.5$  e  $CPI_B = 1.6$ . Sabe-se que a frequência de clock da máquina A é de 500MHz. Sabendo que as máquinas tiveram o mesmo desempenho, determine o período de clock da máquina B.

(2,0) 5) Proponha uma sequência de instruções MIPS que corresponda ao seguinte trecho de um programa de alto nível:

```
a = B[2];
```

```
if a ≥ 5
```

```
  a = a - 2;
```

```
else
```

```
  a = a + 5;
```



.5-2.0

4 - 2,0

R.A. 036291  
26/04/07

OK

5)  $a = \$S_3$      $\$S_4 = \text{end. base de } B$      $\$S_0 = 0$

```
lw      $S3, 8($S4) -
addi    $t0, $S0, 0 -
slt     $t1, $S3, $t0 -
bne     $t1, $S0, ELSE -
add     $t2, $S3, $S0 -
addi    $S3, $t2, -2 -
        END
ELSE:   addi $t2, $S3, $S0 -
        addi $S3, $t2, 5 -
END:    instrução nula.
```

o código tem  
redundância;  
mas ok.

0,75 2a) O sinal RegDst necessita ser 1, uma vez que o registrador destino encontra-se no campo RD da instrução. Já o sinal OrigAlu tem que ser 0 porque o segundo operando deve vir da saída do banco de registradores. —

MemParaReg deve ficar em 0 já que o dado a ser escrito é o resultado da operação que se encontra na saída da ALU. A opção B se tem de depender do campo "FUNCT". —

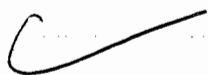
0,75 b) RegDst é 0 uma vez que o registrador de destino vem de RT. OrigAlu é 1 porque o segundo operando é determinado pelo campo de endereço. MemParaReg é 1 porque os dados a serem escritos devem vir da memória. A ALU calculará o endereço de memória com o valor de RS (registrador base) e o valor de deslocamento em relação ao valor base. —

0,25 c) Porque em Branch 2 operandos são comparados, o sinal indica o resultado dessa comparação. A comparação com que índice se haverá algum salto, determinando o endereço a ser passado para PC. —

qual sinal?  
E  
branch 11?

0,75 d) MemParaReg é don't care state porque não são escritos dados nos registradores. EscrereMem é 1 porque serão escritos dados em uma posição de memória. Assim sendo a ALU calculará o endereço da memória com o registrador base e o offset de endereço. —

3) A afirmação não pode ser tomada como verdadeira, uma vez que o período do clock depende ainda dos atrasos inerentes ao circuito. Como exemplo podemos citar os atrasos que ocorrem nas unidades de memória, ALU, Registradores, entre outros.



$$4) \text{Disp.}_A = \text{Disp.}_B \Rightarrow \frac{1}{T_{\text{exec}_A}} = \frac{1}{T_{\text{exec}_B}}$$

$$T_{\text{exec}} = \frac{N^{\circ} \text{inst} \times \text{CPI}}{f}$$

$$T_{\text{exec}_A} = \frac{N \cdot 2,5}{500}$$

$$T_{\text{exec}_B} = \frac{N \cdot 1,6}{f_B}$$

$$\frac{500}{2,5 \cdot N} = \frac{f_B}{1,6 \cdot N}$$

$$2,5 f_B = 500 \cdot 1,6$$

$$f_B = 320 \text{ MHz}$$

$$T_B = \frac{1}{f_B} = 3,125 \text{ ns}$$

