

Aluno: Allan Cordeiro Rocha de Araújo
Professor: Herbert Oliveira Rocha
Organização e Arquitetura de Computadores

Questão 1

Como o Unicidade trabalha com as instruções em apenas um ciclo de clock, o ciclo sempre é ajustado para a instrução mais demorada já nos Multiciclos a execução das instruções é quebrada em passos, assim elas podem usar quantidades diferentes de períodos do clock. É usado menos Hardware para a implementação de tais processos.

Questão 2

É necessário a implementação de 4 registradores (IF/ID, ID/EX, EX/MEM, MEM/WB) para armazenar os dados entre os ciclos de clock.

IF – Busca de Instruções e incremento do PC.

ID – Decodificação da instrução e busca dos registradores

EX – Execução e cálculo do Endereço

MEM – Acesso à memória

WB – Gravação do resultado no banco de registradores

Considerando a execução de instruções do tipo-R, no multiciclo a escrita no banco de registradores é feita no quarto passo já no pipeline é feita no quinto estágio.

Questão 3

```
Loop: subi $t2, $t2, 4      1
      lw  $t1, 0($t2)      2
      add $t3, $t1, $t4    3
      add $t4, $t3, $t3    4
      sw  $t4, 0($t2)      5
      beq $t2, $0, loop    6
```

O primeiro conflito é na primeira linha onde depende do resultado da comparação (desvio condicional) do 'beq' da linha 6, o 'add' da linha 3 precisa esperar o resultado do 'sw' da linha 5.

Questão 4

```
Loop: subi $t2, $t2, 4      linha 1
      lw  $t1, 0($t2)      linha 2
      add $t3, $t1, $t4    linha 3
      add $t4, $t3, $t3    linha 4
      sw  $t4, 0($t2)      linha 5
      beq $t2, $0, loop    linha 6
```

Aplicando a técnica de bypassing resolveria, em tese, os conflitos (Hazards) que causam

Aluno: Allan Cordeiro Rocha de Araújo
Professor: Herbert Oliveira Rocha
Organização e Arquitetura de Computadores

os stalls deste código, por exemplo o segundo 'add' não precisaria esperar o resultado do primeiro 'add' ser armazenado na memória de instruções, quando as ULA terminar a operação já pode ser fornecida como entrada para esta outra soma que depende do valor de '\$t3'.

Questão 5

```
div.d F1, F2, F3  
sub.d F4, F5, F1  
s.d    F4, 4(F10)  
add.d F5, F6, F7  
div.d F4, F5, F6
```

Dependências reais:

3 dependências reais que existem, são elas:

DIV.D(primeiro DIV.D) e SUB.D

SUB.D e S.D

ADD.D e DIV.D (segundo DIV.D)

Dependência de saída:

Entre o SUB.D e DIV.D, onde SUB.D pode terminar depois de DIV.D.

WAW: Uso do F4.

Antidependência

Entre SUB.D e ADD.D.

ADD.D pode escrever em reg. que SUB.D lê.

WAR: Uso de F5 por SUB.D.