Universidade Católica Dom Bosco
Curso de Engenharia de Computação
Disciplina: Arquitetura de Computadores II
Professor: Ricardo Santos

Simulado nº 1

1) Assinale a alternativa correta (V ou F)

   ( ) Uma das vantagens do projeto multiciclo em relação ao monociclo é a possibilidade de manter uma instrução na via de dados enquanto é "realmente" utilizada.

   ( ) Na definição dos sinais de controle da via de dados monociclo, os sinais "MemtoReg" e "MemRead" poderiam ser unidos em um único sinal já que são utilizados apenas pela operação `lw`.

   ( ) Na implementação de um processador MIPS, há sempre necessidade de um multiplexador antes do banco de registradores pois, esse multiplexador é o responsável por selecionar se o endereço apontado por `rs` ou `rd` será utilizado.

   ( ) As operações aritméticas que manipulam imediatos, casos de `addi` e `ori`, utilizam a unidade de extensão do sinal e a unidade de deslocamento à esquerda de 2 bits, uma vez que essas unidades realizam, respectivamente, a expansão de 16 para 32 bits e o alinhamento do número.

   ( ) O sinal `PCWriteCond` é habilitado em qualquer instrução de desvio condicional e não apenas na instrução `beq`.

   ( ) A diferença entre um microcódigo vertical e um horizontal é que o primeiro é mais lento e utiliza menos memória enquanto o segundo é mais rápido mas utiliza mais memória.

   ( ) A razão para a utilização de duas "Dispatch ROM" na UC do processador MIPS é possibilitar que as instruções sejam diferenciados de acordo com o seu tipo (R, I ou J).

   ( ) O conteúdo (sem considerar sinais de controle) do registrador de pipeline IF/ID é a instrução buscada na memória e o endereço do PC. Logo, o tamanho desse registrador é 8 bytes.

   ( ) Duas instruções `a` e `b` com dependência de dados (`a` depende de `b`), não necessitam utilizar a unidade de forwarding se `b` estiver no estágio WB e `a` estiver no estágio ID.

   ( ) Tomando como base a implementação em *pipeline* do processador MIPS, não há possibilidade de colocar a detecção de hazard de controle antes do estágio ID.

2) Indique onde ocorre *stall* nas instruções a seguir e reordene-as de maneira a reduzir esses *stalls*. Se considerarmos que não há *stall* na instrução de desvio condicional, como ficaria o código?

```
4    lw $t3, $t0(0)
8    beq $t2, $t3, 1
12   add $t2, $t1, $t1
16   addi $t1, $t3,1
20   addi $t0, $t0,2
24   st $t1, $t0(0)
```

3) Considere a existência de uma pseudo-instrução `copy rt, rs(imm)` que copia o conteúdo da posição de memória `rs + imm` para a posição de memória indicada pelo conteúdo de `rt`. Essa pseudo-instrução seria traduzida em quais instruções? Quantos ciclos (na implementação multiciclo) levaria para ser executada?

4) Se acrescentarmos mais uma unidade de *forwarding* na via de dados com pipelines do processador MIPS, entre os estágios WB e MEM, essa unidade seria útil em qual(is) situação(ões). Exemplifique sua resposta.

5) Mostre o conteúdo (desconsidere sinais de controle) dos registradores de pipeline de acordo com o código a seguir, após cada ciclo de relógio. Mostre a execução até o estágio EX da última instrução. Quando não for possível determinar o conteúdo, indique 1x11. Tome como base a via de dados com pipelines apresentada na Figura 1. Quando for apresentar o conteúdo de um registrador rx, a partir do banco de registradores, utilize a notação R[rx]. O conteúdo da ALU, a partir das entradas rs e rt para uma operação de soma, deve ser indicado como ALU[[rs] + [rt]]. O conteúdo da memória de dados, cujos operandos foram, por exemplo, rs e 10, deve ser MEM[rs + 10]. O resultado de um FLUSH em um registrador é indicado como 0.

```
4  lw $t0, $t1(10)
8  add $t2, $t2, $t0
12 add $t1$,$t2, $t0
```

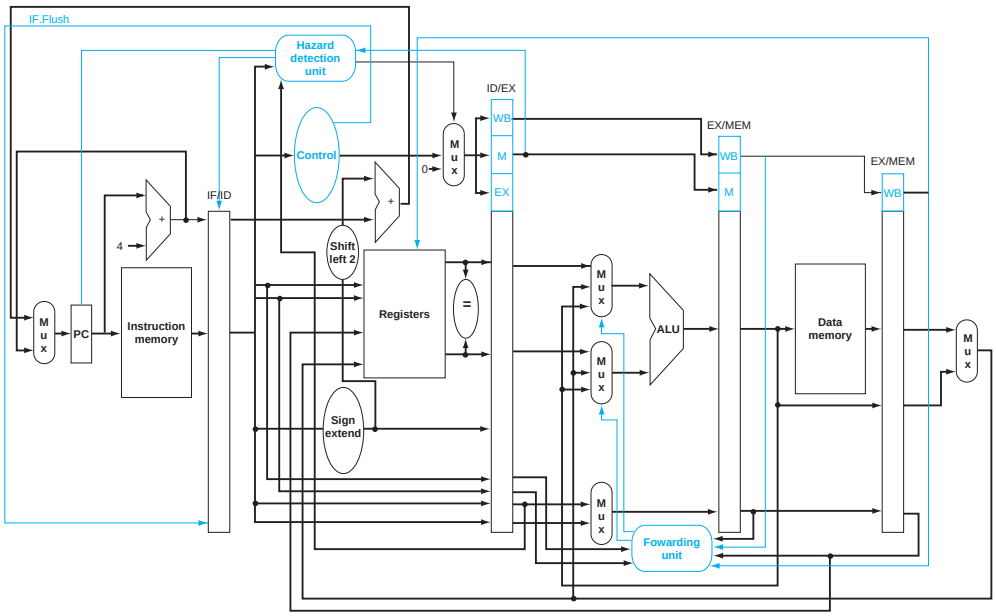

Figura 1: Via de Dados com Pipelines