

Valor das questões: 1) 3,5 pontos 2) 3 pontos 3) 3,5 pontos

1. Determine o número **real** de ciclos de clock para executar uma vez o trecho abaixo (do **blez \$t1,end** até uma instrução que salte para trás ou até a última instrução do trecho ser executada, o que acontecer primeiro). Suponha máxima capacidade de resolução de conflitos de dados (isto inclui uma unidade de adiantamento capaz de adiantar dados da saída do terceiro estágio para a entrada do terceiro estágio, da saída do quarto estágio para a entrada do terceiro, e da saída do quarto estágio para a entrada do quarto estágio). Assuma também que está disponível uma estrutura mestre-escravo para acesso ao banco de registradores e um preditor de saltos de 2 bits que inicialmente prevê salto não-realizado. O PC é escrito no quarto ciclo de relógio de cada instrução. Detalhe a execução no diagrama pipeline abaixo, indique todos os adiantamentos de dados que ocorrerem (se ocorrerem) e mostre as bolhas nas posições adequadas, caso estas existam.

Os valores iniciais dos registradores pertinentes são: \$t0=0x10010000, \$t1=0x44 (=68 base 10), \$t2=0, \$t4=0, \$t3=0x100100AA

INSTRUÇÃO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
loop: blez \$t1,end																						
sltiu \$t2,\$t1,65																						
bne \$t2,\$zero,nxtch																						
sltiu \$t2,\$t1,91																						
beq \$t2,\$zero,nxtch																						
addiu \$t4,\$t4,1																						
nxtch: addiu \$t0,\$t0,4																						
lw \$t1,0(\$t0)																						
j loop																						
end: sw \$t4,0(\$t3)																						

2. O conjunto de instruções do MIPS foi estendido com duas novas instruções chamadas **movn** (move se não zero) e **movz** (move se zero). Por exemplo, a instrução:

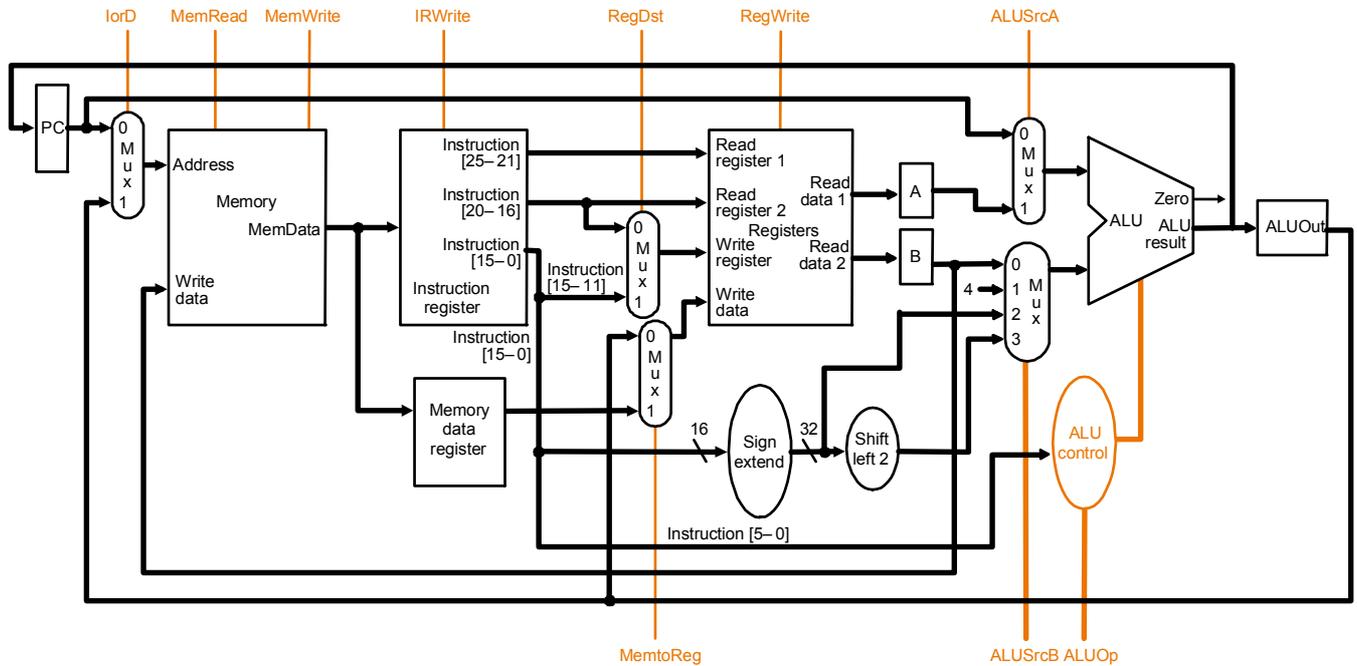
movn \$t3, \$t4, \$t5

copia o conteúdo do registrador \$t4 para o registrador \$t3, se o valor do registrador \$t5 for diferente de zero (caso contrário não faz nada). A instrução **movz** funciona de maneira similar, copiando o conteúdo se o valor do último registrador for igual a zero. Mostre como utilizar estas novas instruções fazendo um trecho de programa para copiar o conteúdo do registrador \$t6 ou \$t7, aquele que contiver o maior valor, para o registrador \$t8. Se os valores dos registradores \$t6 e \$t7 forem iguais, copiar qualquer um deles para \$t8. Você **deve** utilizar o registrador \$t1 como um registrador extra temporário. **Não é permitido utilizar qualquer instrução ou pseudo-instrução de salto condicional.**

3. O conjunto de instruções da arquitetura MIPS foi modificado para acrescentar a instrução **add3** no bloco de dados multi-ciclo descrito na figura abaixo. Esta instrução permite a utilização de 4 operandos, adicionando o conteúdo de 3 registradores e armazenando o resultado da adição em um quarto registrador:

add3 \$t5, \$t6, \$t7, \$t8 # \$t5 ← \$t6 + \$t7 + \$t8

Quais modificações devem ser realizadas no bloco de dados e bloco de controle para atender a modificação proposta?



`add3 $t5, $t6, $t7, $t8 # $t5 ← $t6 + $t7 + $t8`

Quais modificações devem ser realizadas no bloco de dados e bloco de controle para atender a modificação proposta?

Resposta: Uma maneira é fazer com que a instrução use a ULA 2 vezes. Um multiplexador deveria ser adicionado para permitir que a ULA utilize o resultado calculado previamente (armazenado em ALUOut) como entrada para a segunda adição. É importante observar que seriam necessárias mudanças no processo de leitura dos registradores. A instrução pode ser implementada em 5 ciclos, fazendo com que o terceiro registrador fonte seja lido durante o ciclo em que a primeira adição está sendo realizada. Obviamente, um multiplexador e um novo sinal de controle serão necessários para uma das portas de leitura do banco de registradores.

Outras possibilidades, muito menos viáveis na prática incluem:

- 1) Aumentar o número de portas do Banco de Registradores de 3 para 4 (3 de leitura e 1 de escrita) e sequencializar as operações de soma com a ULA de duas entradas.
- 2) Fazer o mesmo que 1) mas fazer a ULA ter três entradas de dados, sendo uma destas ignorada em todas as situações exceto nesta instrução de 4 operandos.