

[illegible]

6.

Porque as instruções formato R usam o terceiro campo de registrador (bit11 ... bit15) para indicar o registrador de escrita, enquanto que a instrução lw usa o segundo campo de registrador (bit16 ... bit20) para indicar o registrador de escrita.

7.

Porque as instruções formato R, bem como beq, usam o segundo registrador (dado2) como o segundo operando da ULA, enquanto que as instruções lw e sw usam os 16 bits estendidos da instrução como segundo operando da ULA.

8.

Porque as instruções formato R, usam o resultado de operação da ULA como dado a ser escrito num registrador, enquanto que a instrução lw, usa o dado lido na memória.

9.

Porque numa instrução de desvio condicional, caso a condição é satisfeita, a linha de controle de desvio é um, e é enviado um endereço de desvio para o PC, calculado deslocando de 2 à esquerda os 16 bits estendidos da instrução, e somando com o valor contido em PC somado de 4. No caso em que a condição não é satisfeita a linha de controle de desvio é zero e o novo endereço de PC é PC somado de 4.

10.

No MIPS multiciclo, necessitamos acrescentar:

RI (registrador de instrução), para que a no final do ciclo de busca de instrução, uma nova instrução contida em RI represente o novo estado do computador;

A e B (registradores de dados lidos do banco de registradores), para que no final do segundo ciclo, os dados a serem processados estejam contidos nesses registradores;

ULA (registrador do resultado da ULA), para que no final do ciclo de operação da ULA, o resultado esteja guardado nesse registrador;

RDM (registrador de dado da memória), para que após a leitura de um dado da memória, o mesmo seja guardado nesse registrador.

11.

a) No estado 2, ULAFonteA = 1, ULAFonteB = 10, ULAOp = 00. O operando 1 da ULA é sempre o dado1 do banco de registradores. O operando 2, como ULAFonteB = 10, são os 16 bits da instrução estendidos para 32 bits.

A operação aritmética é dada por ULAOp = 00, que resulta em adição.

b) No estado 4, EscReg = 1, MemParaReg = 1 e Regdst = 0. O dado a ser escrito no registrador depende do multiplexador MemParaReg, que como é igual a 1, deve vir do registrador RDM.

12.

1) Acrescentar o estado 10 com a seguinte operação – calcular a adição de um operando do registrador com um segundo operando proveniente dos 16 bits da instrução estendidos para 32 bits, colocando o resultado no registrador ULA. Os sinais de controle são ULAFonteA = 1, ULAFonteB = 10 e ULALOp = 00, para a operação de adição.

2) Acrescentar o estado 11 em sequência ao estado 10, com a operação de escrita do resultado contido no registrador ULA no registrador destino. Os sinais de controle são MemParaReg = 0 para selecionar o caminho do registrador ULA para o registrador destino; EscReg = 1, para escrever no registrador; e RegDst = 0, para selecionar o registrador destino.