

UNIVERSIDADE FEDERAL DE OURO PRETO – UFOP

Ciência da Computação



ARQUITETURA DE COMPUTADORES

LISTA I

Marcus Vinícius Souza Fernandes

19.1.4046

Ouro Preto

2021

Questão 4.1

4.1.1

Podemos ressaltar que matematicamente, o fio de controle MemRead é um "don't care": a instrução será executada corretamente, independentemente do valor escolhido. Praticamente, no entanto, MemRead deve ser definido como falso para evitar causar uma falha de segmento ou perda de cache.

4.1.2

Os registros são: ALUSrc mux, ALU e o mux MemToReg.

4.1.3

Todos os blocos produzem alguma saída. As saídas de DataMemory e Imm Gen não são utilizadas.

Questão 4.2

4.2.1

Reg2Loc para ld: Ao executar ld, não importa qual valor é passado para "Registro de leitura 2", porque o ALUSrc mux ignora a saída de "Dados de leitura 2" resultante e seleciona o valor imediato estendido de sinal em seu lugar.

MemToReg para sd e beq: Nem sd nem beq gravam um valor no arquivo de registro. Não importa qual valor o memToReg mux passa para o arquivo de registro, porque o arquivo de registro ignora esse valor.

Questão 4.3

4.3.1

Cálculo: $25 + 10 = 35\%$.

Apenas Load e Store usam memória de dados.

4.3.2

Total de 100%, pois, cada instrução deve ser buscada na memória de instrução antes de ser executada.

4.3.3

Cálculo: $28 + 25 + 10 + 11 + 2 = 76\%$.

Apenas as instruções do tipo R não usam o Extensor de sinal.

4.3.4

A extensão do sinal produz uma saída durante cada ciclo. Se sua saída não for necessária, ela será simplesmente ignorada.

Questão 4.4

4.4.1

Apenas cargas são quebradas. MemToReg é 1 ou “don’t care” com todas as outras instruções.

4.4.2

Tipo I, cargas, lojas estão todas quebradas.

Questão 4.5

4.5.1

ALUop - 00.

ALU Control Lines - 0010.

4.5.2

O novo PC é o antigo PC + 4. Este sinal vai do PC, através do adicionador “PC + 4”, através do mux “branch” e de volta para o PC.

4.5.3

ALUsrc:

Inputs: Reg[x12] and 0x0000000000000014;

Output: 0x0000000000000014;

MemToReg:

Inputs: Reg[x13] + 0x14;

Branch:

Inputs: PC+4 and 0x000000000000000A;

4.5.4**ALU inputs:**

Reg[x13] and 0x0000000000000014 PC + 4 adder;

inputs: PC and 4;

Branch adder:

inputs: PC and 0x0000000000000028;

4.5.5

Tipo I, cargas, lojas estão todas quebradas.

Questão 4.7**4.7.1**

R-type : $30 + 250 + 150 + 25 + 200 + 25 + 20 = 700\text{ps}$.

4.7.2

1d : $30 + 250 + 150 + 25 + 200 + 250 + 25 + 20 = 950\text{ os}$.

4.7.3

sd : $30 + 250 + 150 + 200 + 25 + 250 = 905$.

4.7.4

beq : $30 + 250 + 150 + 25 + 200 + 5 + 25 + 20 = 705$.

4.7.5

l-type : $30 + 250 + 150 + 25 + 200 + 25 + 20 = 700\text{ps}$.

4.7.6

950ps.