

Lista 4

Gustavo Soares Silva

5.

a) Reg Write: não permitiria mais escrita no banco de registradores

b) AluOp 0: permitiria apenas instruções de SW, LW e instruções de branch

c) AluOp 1: permitiria apenas instruções de SW, LW e instruções do tipo aritméticas-lógicas

d) Branch: não permitiria mais instruções de branch

e) Mem Read: não deixaria mais leitura da memória (LW)

f) Mem Write: não deixaria mais escrita na memória (SW)

8) * Não é possível fazer essa pergunta pois não temos acesso às essas figuras 5.17 e 5.18.

14) swap \$rs, \$rt

↓
add \$aux, \$rs, \$0

add \$rs, \$rt, \$0

add \$rt, \$aux, \$0

Por serem 3 instruções, a recomendação seria incrementar o clock period em 30% para compensar a single-instruction implementation.

35) A instrução SW tem dois registradores de entrada e faz um incremento de add.

Para calcular o endereço de memória, é necessário passar o RS (com o imediato) no EXE.

Nessa forma, o RT (com o dado a ser escrito) pode ser atrasado e movido apenas no estágio de memória, usando apenas um register file.

6.

1) O cálculo de instruções por ciclo no monociclo é a soma do tempo que cada etapa (ID, EXE, ...) tem. Já o multiciclo é o maior tempo entre estas etapas $\times 5$.

Nessa forma, enquanto no monociclo, o CPI vai ter 2 ns, o maior, o multiciclo vai ter o CPI $4 \text{ ns} \times 5 = 20 \text{ ns}$.

LOVE

tilibra

add	IF	ID	EX	MEM	WB
add		IF	ID	EX	MEM
add			ID	EX	MEM

* adiantamento de registrador \$4

4

add (\$2), \$5, \$4

add (\$4), (\$2), \$5

sw \$5, 100(\$2)

add \$3, (\$2), (\$4)

- \$2 não está pronto no EXE

então pode adiantar para a segunda

operação EXE - EXE

E também adianta da MEM

para EXE na terceira instrução

1 \$4 adianta da MEM - EXE da segunda para a quarta instrução

1 último \$2 não precisa de adiantar

9 IF: não é possível adiantar mas há informações suficientes

ID: instrução do tipo R

EX: operação de escrita na memória

MEM: instrução de teste condicional (branch)

WB: instrução de leitura da memória

(tudo vem da memória e é escrito no RB)



↓

IF	ID	EX	MEM	WB
	IF	ID	EX	MEM
		ID	EX	...
		IF	ID	...
			IF	

registradores no quinto ciclo

- leitor: \$10
- escritor: \$1

19) A situação de leitura após escrita com registradores e diferente de memória, porque na de registradores a escrita ocorre apenas no quinto ciclo, enquanto a leitura ocorre no segundo. Já a leitura e escrita ocorrem na mesma etapa, assim uma situação de escrita na memória e depois uma de leitura não gera conflito.

20) A forma para resolver esse problema de conflito é fazer um adiamento / realocimentação no estágio de memória, pois o resultado fica pronto no quinto ciclo e o dado é usado novamente logo em seguida para ser armazenado.

if (memWrite = 1 and
WB.endercoRegDat = MEM.endercoRegDat) {

adianta o dado da saída do MEM para a entrada;

