

Universidade Federal de Santa Catarina, INE/CTC
INE 5366 – Arquitetura de Computadores I
Segunda avaliação 2006.2

Aluno(a):

Matrícula:

Parte I [valor 3,0 pontos]

1. Responda sucintamente as perguntas abaixo: [0,5 ponto cada item]

- a) Devido a uma falha de fabricação, um circuito integrado que implementa a CPU do Anexo III teve a saída “zero” da ALU fixada para sempre no nível lógico “1”. Afirmção: “Todos os desvios condicionais de um programa arbitrário que rode nesse circuito executarão incorretamente”. A afirmação é verdadeira ou falsa ? **Justifique**.
- b) No datapath do Anexo IV, uma instrução buscada é armazenada no registrador de instrução (IR). Afirmção: “O conteúdo de IR precisa ser preservado apenas até a etapa de decodificação da instrução e leitura dos operandos”. Esta afirmação é verdadeira ou falsa ? **Justifique**.
- c) Afirmção: “Se o datapath do Anexo IV fosse estendido para suportar tipos adicionais de exceções, sem usar interrupção vetorizada, o multiplexador que define o novo valor a ser carregado em PC precisaria ser modificado”. A afirmação é verdadeira ou falsa ? **Justifique**.
- d) Modifique o datapath do Anexo III entre a saída “zero” da ALU e a entrada inferior da porta AND (conforme esboçado abaixo) de forma que ele possa executar dois tipos de desvio condicional, beq e bne, sem que nenhuma outra modificação seja necessária no datapath, mas com a seguinte alteração na unidade de controle: há um novo sinal de controle denominado “**eq/~ne**”, que é colocado em nível “1” ou nível “0”, conforme a instrução seja decodificada como sendo beq ou bne, respectivamente. Suponha também que o sinal “Branch” é ativado em nível “1” somente quando a instrução for um desvio condicional, independentemente de qual tipo.



Nota: nos itens 1e e 1f, você deverá preencher diagramas de ocupação de um pipeline de 5 estágios. Indique com o respectivo acrônimo (IF, ID, EX, ME, WB) o estágio ocupado por uma instrução em um dado ciclo. Indique com um "X" o(s) ciclo(s) em que uma instrução deveria ocupar um estágio, mas não pode ocupá-lo num determinado ciclo por causa de um hazard.

- e) Complete o diagrama (abaixo) de ocupação dos 5 estágios de um pipeline, supondo a ocorrência de "hazard" estrutural devido a uma memória única compartilhada por dados e instruções.

lw \$s2, 0(\$s1)									
lw \$s4, 0(\$s3)									
lw \$s6, 0(\$s5)									
add \$s9, \$t0, \$t1									

- f) Assuma que o datapath do Anexo 5 seja modificado para incluir um caminho de "forwarding" da porta de leitura da memória de dados para a entrada da ALU e outro da saída para a entrada da ALU. Complete o diagrama mostrando a ocupação dos estágios do pipeline durante a execução do código abaixo. **Quando ocorrer fluxo de dados através de caminhos de forwarding, indique tal fluxo com setas** (origem no estágio produtor, destino no estágio consumidor).

lw \$s0, 20(\$t2)	IF	ID	EX	MEM	WB						
sub \$t2, \$s0, \$t3											
add \$t4, \$t2, \$t4											

Parte II [valor 4,0 pontos]

Lembrete: O valor de um sinal de controle pode ser "1", "0" ou "X" (don't care). Atribua a um sinal de controle o símbolo "X", sempre que se seu valor puder ser ou "1" ou "0", sendo portanto irrelevante. Em sinais de controle de mais de um bit, o valor de cada bit deve ser representado individualmente.

2. [1,0; 0,1 cada sinal correto] Consulte o Anexo III e indique os valores dos sinais de controle para realizar cada uma das instruções abaixo. (Penalidade: 5 ou mais sinais errados no total anulam a questão).

	RegDst	ALUSrc	MemtoReg	RegWrite	MemWrite
sw					
addi					

3. [1,0; 0,05 cada sinal correto] Consulte os Anexos e indique os valores dos sinais de controle que comandam cada uma das fases de execução de uma instrução **lw** na CPU do Anexo IV. **Lembrete:** Naquela CPU, qualquer que seja a instrução, o cálculo especulativo do endereço-alvo é sempre realizado. (Penalidade: 10 ou mais sinais errados no total anulam a questão).

a) 1ª fase

MemRead	MemWrite	ALUSrcA	IoD	IRWrite	ALUSrcB	PCWrite	PCSource

b) 2ª fase

ALUSrcA	ALUSrcB

c) 3ª fase

ALUSrcA	ALUSrcB	PCWrite	PCWriteCond

d) 4ª fase

MemRead	IoD	IRWrite

e) 5ª fase

RegWrite	RegDst	ALUSrcA

4. [0,5] Suponha que o datapath do Anexo V foi estendido para implementar previsão estática para desvios condicionais: o controlador comanda a busca de uma nova instrução a cada ciclo, assumindo que os desvios condicionais não sejam tomados. O teste (comparação) associado ao desvio é realizado na ALU. Se o resultado indicar que a hipótese de previsão estava incorreta, a instrução correta é buscada no ciclo seguinte ao que o resultado do teste é definido. Além disso, a partir desse ciclo a(s) instrução(ões) buscadas incorretamente são anuladas. O diagrama abaixo ilustra a ocupação de estágios durante a execução de um programa. Supondo que a previsão resultou incorreta, indique no diagrama os ciclos de relógio onde a ocupação do estágio por uma instrução foi anulada por causa do hazard de controle. Para isso, risque os ciclos em que uma instrução ocupa indevidamente um dado estágio (~~EX~~, por exemplo).

	1	2	3	4	5	6
beq \$s1, \$s2, L	IF	ID	EX	ME	WB	
add \$s3, \$s4, \$s5		IF	ID	EX	ME	WB
sw \$s1, 0(\$t0)			IF	ID	EX	ME
L: lw \$s2, 0(\$t1)				IF	ID	EX
sw \$s6, 0(\$t3)					IF	ID
sub \$s7, \$s0, \$s1						IF

5. [1,5: 6 x 0,25] Indique os valores dos seguintes sinais de controle no ciclo 6 do diagrama acima.

RegDst	ALUSrc	Branch	MemWrite	RegWrite	MemtoReg

Parte III [valor 3,0 pontos]

6. [1,0] Suponha que o datapath do Anexo V tenha sido estendido para incorporar dois caminhos de “forwarding”: um da porta de leitura da memória de dados para a entrada da ALU no ciclo seguinte; outro conectando a porta de leitura da memória de dados com a porta de escrita da mesma memória de dados no ciclo seguinte. Mostre a ocupação dos estágios de pipeline. **Quando ocorrer fluxo de dados através de caminhos de forwarding, indique tal fluxo com setas** (origem no estágio produtor, destino no estágio consumidor).

	1	2	3	4	5	6	7	8	9	10	11	12
lw \$s0, 0(\$t1)												
sw \$s1, 0(\$s0)												
add \$s0, \$s4, \$s5												
add \$s6, \$s0, \$t2												
lw \$s3, 0(\$s6)												
sw \$s3, 0(\$t4)												

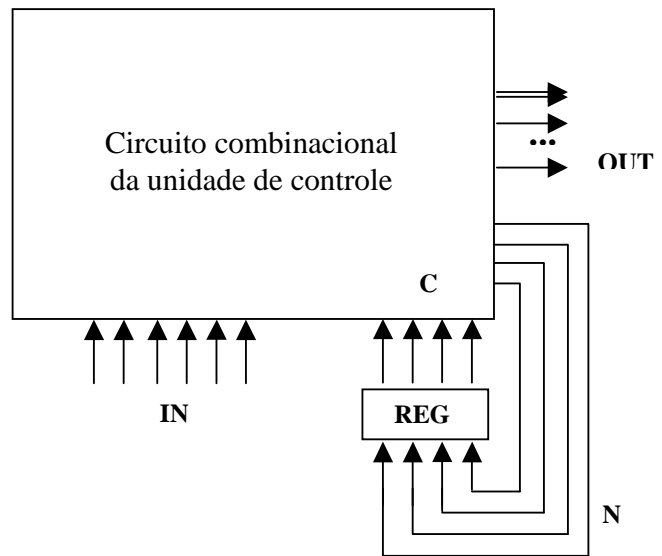
7. [0,5] Afirmção: “Para o cenário da Questão 6, o código pode ser escalonado de forma a eliminar todas as paradas devidas a hazards de dados”. A afirmação é verdadeira ou falsa? Se verdadeira, justifique mostrando o código escalonado; se falsa, justifique mostrando porque as paradas não podem ser eliminadas.

8. [1,5: 0,5 + 0,25 + 0,25 + 0,5] São mostradas a seguir a tabela de transições da máquina de estados e a estrutura do controlador para a CPU do Anexo IV. Suponha que a máquina esteja no estado 1 e que o código operacional da instrução corrente seja 000000. Lembre que no estado 1, além da decodificação e busca de operandos, é efetuado o cálculo do endereço-alvo, assumindo provisoriamente que a instrução possa vir a ser um desvio condicional.

Consulte os Anexos, interprete as informações dadas (acima, na tabela e na figura) e deduza os valores dos sinais na saída **OUT** do controlador, bem como da entrada **N** e da saída **C** do registrador de estado.

- a) **OUT** = (ALUSrcA, ALUSrcB, PCWrite, IRWrite, ...) = (, , , , ...).
b) **C** = (A3, A2, A1, A0) = (, , ,).
c) **N** = (N3, N2, N1, N0) = (, , ,).

Estado atual	IN	Próximo estado
0	XXXXXX	1
1	10X011	2
1	000000	6
1	000100	8
1	000010	9
2	100011	3
2	101011	5
3	XXXXXX	4
4	XXXXXX	0
5	XXXXXX	0
6	XXXXXX	7
7	XXXXXX	0
8	XXXXXX	0
9	XXXXXX	0



- d) Baseando-se **somente** na tabela de transições acima, desenhe a máquina de estados finitos, rotulando cada estado com seu número e cada transição com o valor da entrada (IN) que a provoca. **Transições que independem do valor das entradas NÃO devem ser rotuladas.**