

Introdução a Arquitetura de Computadores

1. O que há por trás do software de aplicação?
2. Que linguagem o computador entende?
3. Quais os componentes básicos do computador de Von Neumann?
4. Defina Arquitetura do Computador.
5. Considere um microprocessador hipotético, cujas instruções de 32 bits são compostas de dois campos: o primeiro byte contém o código de operação e os demais contêm um operando imediato ou um endereço de operando.
 - a) Qual a capacidade máxima de memória endereçável diretamente (em bytes)?
 - b) Discuta o impacto sobre a velocidade do sistema, caso o barramento do microprocessador tenha:
 - o um barramento local de endereços de 32 bits e um barramento local de dados de 16 bits; ou
 - o um barramento local de endereços de 16 bits um barramento local de dados de 16 bits.
 - c) Quantos bits são necessários para o contador de programa e para o registrador de instruções?

Conjunto de instruções Neander/Ramsés

1. Faça vários programas diferentes que zerem o acumulador.
3. Escreva um programa que determine qual a maior de 3 variáveis positivas de 8 bits armazenadas nas posições de memória 80H a 82H. O resultado (a maior variável) deve aparecer na posição de memória 83H.
2. Limpeza de uma área de memória: Faça um programa para zerar 32 posições de memória consecutivas na memória. O endereço inicial desta área é fornecido na posição 80h. É garantido que o endereço inicial está entre 82h e DCh.
3. Explique como a representação em complemento de 2 é usada pela ULA do processador Neander.
4. A partir do programa de multiplicação de dois inteiros positivos, feito na sala de aula, proponha uma estratégia que permita também multiplicar números negativos.
5. Faça um programa para identificar se um número é negativo ou positivo.
6. Reescreva os programas acima usando o conjunto de instruções do processador Ramsés procurando reduzir o uso de memória de código e de dados.
7. Considerando o processador Ramsés, qual seria o código binário para as instruções:
ADD B #01
STR A 80,I
JMP 10

Considere os números em hexadecimal.

Organização do processador Neander

1. Descreva o efeito de uma falha “stuck-at-0” (ou seja, independente do que deveria ser, o sinal é sempre 0) para os sinais mostrados a seguir, no caminho de dados do processador Neander. Que instruções, se houver, não funcionarão corretamente? Explique por quê.
 - a) CargaAC
 - b) CargaRDM
 - c) SEL
 - d) AND
 - e) Write
2. Considere a figura que mostra a organização interna do processador Neander. Explique as etapas de execução da instrução STA end indicando os blocos e sinais envolvidos em cada fase.

3. Desejamos acrescentar a instrução SLA (shift left Accumulator – deslocando um bit à esquerda no acumulador) ao caminho de dados do processador Neander. Faça as inclusões e modificações necessárias no caminho de dados para que esta instrução possa ser executada.
4. Desejamos acrescentar a instrução JR (Jump Relative – desvio do fluxo de execução relativo ao endereço da própria instrução) ao caminho de dados do processador Neander. Faça as inclusões e modificações necessárias no caminho de dados para que esta instrução possa ser executada.
5. Faça uma sugestão de implementação do processador Neander monociclo. Avalie os benefícios e custos desta implementação, comparando com a versão multiciclo.
6. Faça sugestões de como melhorar o desempenho das implementações monociclo e multiciclo dos processadores estudados.
7. Defina processador multiciclo e monociclo. Indique duas vantagens de cada um deles sobre o outro.

Modos de endereçamento

1. Defina e exemplifique os modos de endereçamento: Imediato, Direto, Indireto e Indexado.
2. Classifique as instruções do processador Neander quanto ao número de endereços.
3. Classifique as instruções abaixo, do processador MIPS, quanto ao número de endereços e quanto ao modo de endereçamento.

```
sw $5, 100($2)
add $3, $2, $4
```

4. Pesquise um processador comercial, selecione duas instruções e classifique-as quanto ao número de endereços e quanto ao modo de endereçamento.
5. Cite e explique 2 tipos de operações/instruções comuns em processadores.
6. Quantas vezes a CPU acessa a memória quando busca e executa uma instrução com modo de endereçamento indireto, se a instrução é:
 - a) uma computação que requer apenas um operando?
 - b) um desvio?

Avaliação de desempenho

1. Deseja-se comparar o desempenho de dois computadores diferentes M1 e M2. As seguintes medições foram feitas nestes computadores.

Programa	Tempo em M1	Tempo em M2
1	2,0 segundos	1,5 segundo
2	5,0 segundos	10,0 segundos

- a) Suponha que M1 custe R\$ 500,00 e M2 custe R\$800,00. Se precisasse usar o programa 1 um grande número de vezes, qual computador você compraria e por quê?
 - b) Suponha que outro usuário possui as seguintes necessidades: o P1 precisa ser executado 1600 vezes a cada hora. Qualquer tempo restante é usado para executar o P2. Se o computador possui desempenho suficiente para executar o programa 1 a quantidade necessária de vezes por hora, então, o desempenho é medido pela vazão para o programa 2. Que computador é mais rápido para este *workload*? Que computador é mais econômico?
3. Considere duas implementações diferentes do mesmo conjunto de instruções (m1 e m2). Há três classes de instruções (A, B e C) no conjunto de instruções. A implementação m1 possui um clock de 6,0GHz e a implementação m2 um clock de 3,0GHz. A tabela fornece um número médio de ciclos para cada classe de instrução em cada implementação. Além disso, é fornecida a proporção média de classes de instrução gerada pelo compilador.

Classe	CPI em m1	CPI em m2	Uso pelo compilador
A	2	1	50%
B	3	2	25%
C	5	2	25%

Qual das duas implementações oferece melhor desempenho?

4. Defina software de benchmark. Cite 2 exemplos.

Pipeline

1. Mostre, através de esboço gráfico, que quanto mais estágios contiver um pipeline ideal, maior será o número de instruções executadas no final de um período. Considere para tal uma sequência de 4 slots de tempo, um caso inicial sem a adoção de pipeline e pipelines de 2, 3 e 4 estágios.
2. O pipeline da questão anterior é um pipeline idealizado. Cite dois motivos pelos quais, na prática, não se obtém o resultado ilustrado na questão anterior.
3. Identifique todas as dependências de dados existentes no código a seguir. Quais dependências são conflitos que podem ser resolvidos com *forwarding* (adiantamento de dados)?

```
add $2, $5, $4
add $4, $2, $5
sw $5, 100($2)
add $3, $2, $4
```

4. Execute o código abaixo no caminho de dados do MIPS sem *forwarding* (adiantamento de dados). Faça o diagrama de tempo da pipeline. Se existem *stalls*, explique o motivo de cada um. Calcule o CPI para o programa.

```
lw $1, 0($4)
add $3, $1, 3
add $2, $3, 3
sub $3, $1, $3
sw $2, 0($4)
```

Agora refaça o diagrama de tempo considerando o caminho de dados com *forwarding*.

5. Dado um processador MIPS com um pipeline de 5 estágios, escreva um trecho de código em linguagem de montagem em que ocorram as seguintes situações:

- a. Adiantamento de dados da saída para a entrada da ALU;
- b. Adiantamento de dados da saída do estágio de memória para a entrada da ALU;
- c. Uma bolha no pipeline devido a uma dependência de controle.

6. Considere o seguinte trecho de código em linguagem de montagem:

```
add R5, R0, R0
add R20, R0, #40
Soma: ld R10, A(R20)
add R5, R5, R10
sub R20, R20, #4
bne R20, R0, Soma
```

Assuma que o pipeline do processador não possui mecanismos de “stalls” ou adiantamento de dados. Reescreva o código inserindo o menor número possível de nops para eliminar as dependências de dados. Se for possível, reordene as instruções para minimizar o número de nops. R0 sempre contém o valor 0.

7. Descreva os tipos de dependências de dados, como e quando elas ocorrem e quais as técnicas para resolvê-las por hardware e por software?

8. A tabela mostra o tempo de atraso introduzido por cada uma das unidades do caminho de dados estudado nas aulas.

Unidade	Mem. instr.	Banco de reg.	ALU	Mem. dados
Tempo de atraso (ps)	200	100	200	200

- a) Suponha que o tempo de execução da ALU é encurtado de 25%. Esta alteração afeta o desempenho do pipeline? Em caso afirmativo, determine a alteração de desempenho.

- b) Responda à questão anterior, supondo que o tempo de execução da ALU aumenta 25%.

Processadores RISC

1. O que caracteriza um microprocessador com arquitetura RISC? Dê exemplos de dispositivos comerciais (Procure exemplos na Internet).
 2. O que caracteriza um microprocessador com arquitetura CISC? Dê exemplos de dispositivos comerciais (Procure exemplos na Internet).
 3. Vantagens e desvantagens da arquitetura RISC.
 4. É fácil identificar claramente se um processador segue uma arquitetura RISC ou CISC? Por que?
- Leitura: RISC vs. CISC: the Post-RISC Era (<http://arstechnica.com/cpu/4q99/risc-cisc/rvc-1.html>)