

## **AD2 - Organização de Computadores 2016.1**

### **Data de entrega: 17/05/2016**

1. (1,5) Considere um sistema onde o número de ciclos de relógio para uma operação por programa é igual a 120, o processador utiliza um relógio de 1200MHz para executar as instruções e nenhuma transferência de dados pode ser perdida. Determine o overhead, em termos de fração de tempo de CPU consumida, que ocorre quando se utiliza a interface por programa para os seguintes dispositivos:

a) Um mouse que deve ser interrogado pelo sistema 30 vezes por segundo para garantir que nenhum movimento dele seja perdido.

*O mouse é consultado 30 vezes por segundo. Cada operação de consulta consome 120 ciclos.*

*São necessários, então,  $30 \times 120 = 3600$  ciclos/segundo para as operações de consulta ao mouse.*

*O relógio é de 1,2GHz, que corresponde a  $1,2 \times 10^9$  ciclos.*

*O overhead será obtido pela seguinte razão:*

$$3600 \text{ ciclos/seg} / (1,2 \times 10^9 \text{ ciclos/seg}) = 0,000003 \text{ ou } 0,0003\%$$

b) Um CD-ROM que transfere dados para o processador em unidades de 32 bits e possui uma taxa de transferência de dados de 240KB/segundo.

*Taxa de transferência = 240KB/s = 240.000 B/s.*

*Em cada operação é transferida uma unidade de 32 bits ou 4 bytes.*

*A cada segundo ocorrem  $240.000B / 4B = 60.000$  operações, como cada operação leva 120 ciclos, então, o total de ciclos será  $60.000 \times 120 = 7,2 \times 10^6$  ciclos/s.*

*O relógio é de 1,2GHz, que corresponde a  $1200 \times 10^6$  ciclos/s.*

*O overhead será obtido pela seguinte razão:*

$$7,2 \times 10^6 \text{ ciclos/s} / 1200 \times 10^6 \text{ ciclos/s} = 0,006 \text{ ou } 0,6\%$$

*OBS: as taxas de transferência utilizam a base 10, e não a base 2.*

*1KB/s corresponde a 1.000B/s e 1MB/s a 1.000.000B/s.*

c) Um disco rígido que transfere dados para o processador em unidades de 16 bits e possui uma taxa de transferência de 3MB/segundo.

*Taxa de transferência = 3MB/s = 3.000.000 B/s.*

*Em cada operação é transferida uma unidade de 16 bits ou 2 bytes.*

*A cada segundo ocorrem  $3.000.000B / 2B = 1.500.000$  operações, como cada operação leva 120 ciclos, então, o total de ciclos será  $1.500.000 \times 120 = 180 \times 10^6$  ciclos/s.*

*O relógio é de 1,2GHz, que corresponde a  $1200 \times 10^6$  ciclos/s.*

*O overhead será obtido pela seguinte razão:*

$$180 \times 10^6 \text{ ciclos/seg} / 1200 \times 10^6 \text{ ciclos/s} = 0,15 \text{ ou } 15,0\%$$

2. (1,5) Explique como funciona um disco rígido (sugestões de fonte de consulta: livro do Stallings e sites na Internet. Na sua resposta indique as suas fontes de consulta).

*Fonte: Texto base retirado de <http://www.tecmundo.com.br/aumentar-desempenho/3469-como-funciona-um-disco-rigido-.htm>*

#### **1) Os discos:**

*Os dados são gravados em discos magnéticos chamados platters, formados por discos extremamente rígidos que garantem a qualidade de gravação e leitura. A utilização de materiais duros é necessária para evitar a ocorrência de erros que podem surgir caso ocorram deformações na superfície. Os platters são compostos de duas camadas, a primeira chamada de substrato. Geralmente feita de alumínio, embora existam modelos mais caros que utilizem vidro como base, esta camada é constituída de um disco polido em salas limpas (sem a presença de fatores ambientais como pó ou sujeira) para se tornar perfeitamente plana e espelhada. O que permite a gravação de dados é uma segunda camada feita de material magnético, aplicada nos dois lados do substrato polido. Para aplicar esta camada, nos discos rígidos antigos era*

utilizada uma técnica chamada eletroplating, semelhante à eletrólise usada para banhar bijuterias a ouro. Como o eletroplating não permite obter uma superfície muito uniforme, o que limitava o espaço de armazenamento disponível, foi desenvolvido o PLA, tecnologia semelhante à usada para soldar transistores em processadores. Como a camada magnética possui espessura de somente alguns microns, acima dela há uma fina camada protetora que evita que pequenos impactos danifiquem o disco rígido. A necessidade de utilizar discos totalmente planos vem do fato de a gravação e a leitura serem feitas a velocidades muito grandes, o que faz com que qualquer variação na superfície seja fatal para o funcionamento do componente. Os HDs mais comuns são capazes de alcançar entre 5.600 a 7.200 rotações por minuto, embora existam modelos que chegam até os 10.000 RPM. Depois de polidos e já com a camada magnética aplicada, os platters são montados em um eixo geralmente feito de alumínio, que deve ser sólido o bastante para evitar qualquer vibração dos discos, mesmo em altas velocidades. O eixo também passa por um processo de polimento, já que os discos devem ficar perfeitamente presos e alinhados ao componente. A maioria dos discos rígidos disponíveis no mercado utiliza múltiplos discos de gravação e leitura, o que permite aumentar a capacidade do dispositivo.

## **2) Cabeça de leitura e motor de movimentação da cabeça:**

Todo o processo de gravação e leitura dos dados de um disco rígido é feito com a utilização de cabeças de leitura eletromagnéticas, presas a um braço móvel que permite o acesso a toda a superfície do disco rígido. O braço de leitura geralmente é feito de alumínio, pois precisa ser ao mesmo tempo leve e resistente para suportar a grande velocidade com que os discos giram. Para coordenar o movimento das cabeças de leitura, os HDs contam com um dispositivo especial chamado actuador. Os discos mais recentes utilizam um mecanismo chamado voice coil, que trabalha com atração e repulsão magnética. Esse sistema é muito mais rápido e preciso do que os motores de rotação, permitindo um melhor aproveitamento da densidade do disco, além de uma maior velocidade na gravação e leitura de arquivos. A movimentação do braço do disco rígido é feita através de dois poderosos ímãs de neodímio, localizados em cada um de seus lados. A passagem de correntes elétricas de diferentes intensidades faz com que a potência de cada ímã seja alterada, o que ocasiona a movimentação dos braços e, conseqüentemente, das cabeças de leitura.

## **3) A gravação e leitura de dados :**

A gravação e a leitura dos dados são feitas através de minúsculos eletroímãs presentes nas cabeças de leitura do disco rígido. Compostos de uma bobina de fios que envolvem um núcleo de ferro, esses ímãs são extremamente precisos, o que permite a gravação de trilhas medindo centésimos de milímetros. Durante o processo de gravação, o campo magnético gerado pelos ímãs presentes nas cabeças faz com que as moléculas de óxido de ferro presentes na superfície magnética dos discos se reorganizem, alinhando os pólos negativos delas com os pólos positivos da cabeça. Da mesma forma, os pólos positivos se alinham com os pólos negativos. Os eletroímãs presentes nas cabeças de leitura e gravação podem ter sua polaridade alternada constantemente, o que permite variar livremente as moléculas da superfície magnética do disco rígido. Conforme a direção de cada pólo, obtém-se um bit interpretado como 1 ou 0 pelo computador. Na hora de ler os dados gravados, a cabeça de leitura capta o campo magnético gerado pelas moléculas alinhadas: a variação dos sinais magnéticos positivos e negativos gera uma corrente elétrica transmitida para a bobina de fios presentes na cabeça. Ao chegar à placa lógica do HD, esta corrente é interpretada como uma sequência de bits 1 e 0, que formam os diferentes arquivos gravados no disco rígido. Vale mencionar que todo esse processo ocorre sem nenhuma espécie de contato entre as cabeças de leitura e a superfície do disco. Isso porque, devido às altas velocidades com os que os discos rodam, forma-se um colchão de ar que repele as cabeças de leitura e impede qualquer espécie de contato. Caso houvesse contato entre os componentes, dificilmente um disco rígido funcionaria durante muito tempo devido aos danos físicos ocorridos. Para evitar acidentes, a maioria dos HDs conta com um ímã ao lado do atuador, responsável por atrair as cabeças a uma posição segura toda vez em que o computador é desligado ou não há gravação ou leitura de dados.

3. (2,0) Considere um computador, cuja representação para ponto fixo e para ponto flutuante utilize 32 bits. Na representação para ponto flutuante, utiliza-se o padrão IEEE 754 para precisão simples.

a) Caso o computador tenha armazenado o conteúdo  $(C5980000)_{16}$ , qual o valor deste número em decimal, se considerarmos que este padrão de bits representa um inteiro utilizando-se representação sinal magnitude, um inteiro em complemento a 2 e um real em ponto flutuante ?

$$(C5980000)_{16} = (11000101110011000000000000000000)_2$$

*Sinal magnitude:*

$$-(2^{30} + 2^{26} + 2^{24} + 2^{23} + 2^{20} + 2^{19}) = -1.167.589.376$$

*Complemento a 2:*

$$-2^{31} + (2^{30} + 2^{26} + 2^{24} + 2^{23} + 2^{20} + 2^{19}) = -979.894.272$$

*Ponto flutuante:*

$$1 \ 10001011 \ 001100000000000000000000$$

*Sendo:*

$$\text{Sinal} = 1 \Rightarrow \text{negativo}$$

$$\text{Expoente} = 139 - 127 \Rightarrow \text{expoente} = 12$$

$$\text{Mantissa} = 001100000000000000000000$$

$$\begin{aligned} \text{Temos então} \Rightarrow & -1,001100000000000000000000 \times 2^{+12} \\ & -1001100000000,000 = -4864 \end{aligned}$$

b) Qual será a representação em ponto flutuante dos seguintes valores decimais neste computador:

i. +9.325,125

*Convertendo para binário = +10010001101101,001 =>*

*colocando na notação científica  $1,0010001101101,001 \times 2^{+13}$*

*Temos então:*

$$\text{Sinal} = 0 \text{ (positivo)}$$

$$\text{Expoente} = 13_{10} \Rightarrow 13 + 127 = 140 \Rightarrow 10001100_2 \text{ (por excesso)}$$

$$\text{Mantissa} = ,0010001101101001$$

$$\begin{aligned} \text{Resultado: } & 0 \ 10001100 \ 001000110110100100000000 \quad \text{ou} \\ & 01000110 \ 00010001 \ 10110100 \ 10000000 \end{aligned}$$

ii. -11,375

*Convertendo para binário = -1011,011 => colocando na notação científica  $-1,011011 \times 2^3$*

*Temos então:*

$$\text{Sinal} = 1 \text{ (negativo)}$$

$$\text{Expoente} = 3_{10} \Rightarrow 3 + 127 = 130 \Rightarrow 10000010 \text{ (por excesso)}$$

$$\text{Mantissa} = ,011011$$

$$\begin{aligned} \text{Resultado: } & 1 \ 10000010 \ 011011000000000000000000 \quad \text{ou} \\ & 11000001 \ 00110110 \ 00000000 \ 00000000 \end{aligned}$$

c) Mostre a representação dos números dos itens da questão acima, caso se utilizasse a representação complemento a 2 para representar o expoente.

i. +9.325,125

*Convertendo para binário = +10010001101101,001 =>*

*colocando na notação científica  $1,0010001101101,001 \times 2^{+13}$*

*Temos então:*

$$\text{Sinal} = 0 \text{ (positivo)}$$

$$\text{Expoente} = 13_{10} \Rightarrow 00001101_2 \text{ (por C2)}$$

$$\text{Mantissa} = ,0010001101101001$$

$$\begin{aligned} \text{Resultado: } & 0 \ 00001101 \ 001000110110100100000000 \quad \text{ou} \\ & 00000110 \ 10010001 \ 10110100 \ 10000000 \end{aligned}$$

ii. -11,375

*Convertendo para binário = -1011,011 => colocando na notação científica  $-1,011011 \times 2^3$*

*Temos então:*

$$\text{Sinal} = 1 \text{ (negativo)}$$

Expoente =  $1_{10} \Rightarrow 00000011$  (por C2)  
Mantissa = , 011011  
Resultado: 1 00000011 011011000000000000000000 ou  
10000001 10110110 00000000 00000000

d) Supondo que se utilize a representação complemento a 2 para o expoente ao invés da representação em excesso, indique quais o menor e o maior valor positivos normalizados na representação em ponto flutuante para este computador (Considere, neste caso, que todas as representações são utilizadas para números normalizados, não existem os casos especiais).

OBS: Não considerando casos especiais, então

Menor valor positivo:  
0 10000000 000000000000000000000000 =  
 $= 1,0 \times 2^{-128} = +2,94 \times 10^{-39}$   
Maior valor positivo:  
0 01111111 111111111111111111111111 =  
 $= 1, 111111111111111111111111 \times 2^{+127} = +3,4 \times 10^{+38}$

4. (2,5) Crie 1 conjunto de instruções de um operando, definidas em Linguagem Assembly, necessárias para a realização de operações aritméticas e elabore um programa para o cálculo da equações abaixo:

#### I\_ CONJUNTO DE INSTRUÇÕES PARA 1 OPERANDO:

ADD X  $\Rightarrow (ACC) \leftarrow (ACC) + (X)$   
SUB X  $\Rightarrow (ACC) \leftarrow (ACC) - (X)$   
MUL X  $\Rightarrow (ACC) \leftarrow (ACC) * (X)$   
DIV X  $\Rightarrow (ACC) \leftarrow (ACC) / (X)$   
LOAD X  $\Rightarrow (ACC) \leftarrow (X)$   
STORE X  $\Rightarrow (X) \leftarrow (ACC)$

$X = (A/D - C) + (B * (D - E/B))$

LOAD A  $\Rightarrow (ACC) \leftarrow (A)$   
DIV D  $\Rightarrow (ACC) \leftarrow (ACC) / (D)$   
SUB C  $\Rightarrow (ACC) \leftarrow (ACC) - (C)$   
STORE T1  $\Rightarrow (T1) \leftarrow (ACC)$   
LOAD E  $\Rightarrow (ACC) \leftarrow (E)$   
DIV B  $\Rightarrow (ACC) \leftarrow (ACC) / (B)$   
STORE T2  $\Rightarrow (T2) \leftarrow (ACC)$   
LOAD D  $\Rightarrow (ACC) \leftarrow (D)$   
SUB T2  $\Rightarrow (ACC) \leftarrow (ACC) - (T2)$   
MUL B  $\Rightarrow (ACC) \leftarrow (ACC) * (B)$   
ADD T1  $\Rightarrow (ACC) \leftarrow (ACC) + (T1)$   
STORE X  $\Rightarrow (X) \leftarrow (ACC)$

5. (2,5) Responda as questões abaixo:

a) Analise os modos de endereçamento direto, indireto e imediato, estabelecendo diferenças de desempenho, vantagens e desvantagens de cada um.

MODO DE ENDEREÇAMENTO	DEFINIÇÃO	VANTAGENS	DESVANTAGENS	DESEMPENHO
Imediato	O campo operando contém o dado	Rapidez na execução da instrução	Limitação do tamanho do dado. Inadequado para o uso com dados de valor variável	Não requer acesso a memória principal. Mais rápido que o modo direto
Direto	O campo operando contém o endereço do dado	Flexibilidade no acesso a variáveis de valor diferente em	Perda de tempo, se o dado é uma constante	Requer apenas um acesso a memória principal. Mais

		<i>cada execução do programa</i>		<i>rápido que o modo indireto</i>
<i>Indireto</i>	<i>O campo operando corresponde ao endereço que contém a posição onde está o conteúdo desejado,</i>	<i>Manuseio de vetores (quando o modo indexado não está disponível). Usar como "ponteiro"</i>	<i>Muitos acessos à MP para execução</i>	<i>Requer 2 acessos a memória principal</i>

b) Qual é o objetivo do emprego do modo de endereçamento base mais deslocamento? Qual é a diferença de implementação e utilização entre esse modo e o modo indexado?

*O base mais deslocamento tem como seu principal objetivo permitir a modificação de endereço de programas ou módulos destes (que é a relocação de programa), bastando para isso uma única alteração no registrador base.*

*O base mais deslocamento tem como característica o endereço ser obtido da soma do deslocamento com o registrador base, diferindo do modo indexado onde o do registrador base é fixo e variar no deslocamento, ao contrário deste onde o deslocamento é fixo e com a alteração do registrador base permite-se a mudança do endereço. O modo indexado é utilizado para acessar arrays, o campo de "deslocamento" guarda o endereço inicial e registrador mantem o índice do array.*

*Exemplos de instruções modo indexado:*

*LDX Ri, Op ==> ACC <--- ((Op) + (Ri))*

*ADX Ri, Op ==> ACC <--- ACC + ((Op) + (Ri))*

*Exemplo: instrução base mais deslocamento:*

*LDB Rb, Op ==> (ACC) <--- ((Op) + (Rb))*

*ADB Rb, Op ==> ACC <--- ACC + ((Op) + (Rb))*

*Sendo,*

*Op = Operando*

*Ri = Registrador de índice*

*Rb = Registrador base*

c) Compilação e Interpretação (Dê exemplos de linguagens que se utilizem de compiladores e de linguagens que se utilizem de interpretadores).

*A compilação consiste na análise de um programa escrito em linguagem de alto nível (programa fonte) e sua tradução em um programa em linguagem de máquina (programa objeto).*

*Na interpretação cada comando do código fonte é lido pelo interpretador, convertido em código executável e imediatamente executado antes do próximo comando.*

*A interpretação tem como vantagem sobre a compilação a capacidade de identificação e indicação de um erro no programa-fonte (incluindo erro da lógica do algoritmo) durante o processo de conversão do fonte para o executável.*

*A interpretação tem como desvantagem o consumo de memória devido ao fato de o interpretador permanecer na memória durante todo o processo de execução do programa. Na compilação o compilador somente é mantido na memória no processo de compilação e não utilizado durante a execução. Outra desvantagem da interpretação está na necessidade de tradução de partes que sejam executadas diversas vezes, como os loops que são traduzidos em cada passagem. No processo de compilação isto só ocorre uma única vez. Da mesma forma pode ocorrer para o programa inteiro, em caso de diversas execuções, ou seja, a cada execução uma nova interpretação.*

*Exemplos de linguagem interpretadas: ASP, BASIC, Java, PHP, Python, Lisp entre outras.*

*Exemplos de linguagem compilada: C, Pascal, Delphi, Visual Basic, entre outras.*

d) Sistemas SMP e Sistemas NUMA (Forneça exemplos atuais de sistemas SMP e Sistemas NUMA).

*Sistemas SMP (ou UMA) têm como característica o acesso a todas as partes da memória principal com tempo de acesso uniforme. Em sistemas NUMA, todos os processadores possuem também acesso a todas as partes da memória principal podendo diferir o tempo de acesso em relação às posições da memória e processador.*

*Nos sistemas SMP o aumento no número de processadores tem como consequência problemas de tráfego no barramento comum degradando o desempenho. Uma solução para isto é a utilização de clusters, que tem, usualmente, como consequência alterações significativas na aplicação (software). Nos sistemas NUMA podem-se ter vários nós multiprocessadores, cada qual com seu próprio barramento, resultando em pequenas alterações na aplicação (software).*

*Exemplo de NUMA: Cray T3D*

*Exemplo de SMP: Maioria dos servidores da HP, Compac, IBM.*