



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Organização de Computadores

Gabarito-AP2 1º semestre de 2012.

Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
 2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
 3. Você pode usar lápis para responder as questões.
 4. Ao final da prova devolva as folhas de questões e as de respostas.
 5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

1. (2,0) Explique a classificação de arquiteturas segundo Flynn.

Flynn classificou as arquiteturas em SISD, SIMD e MIMD. MISD foi incluído para a complementação da tabela.

SISD - Single instruction stream, single data stream. Um único processador executa uma única sequência de instruções sobre dados armazenados em uma única memória. Exemplo: Processadores de computadores pessoais (núcleo simples)

SIMD – Single instruction stream, multiple data stream. Vários elementos de processamento. Cada um tem uma memória de dados. Cada instrução é executada sobre um conjunto de dados diferente. Exemplo: Processadores matriciais.

MISD – Multiple instruction stream, single data stream. A sequência de dados é transmitida para um conjunto de processadores, cada um dos quais executa uma sequência de instruções diferente. Não existem processadores comerciais que utilizam este modelo.

MIMD – Multiple instruction stream, multiple data stream. Conjunto de processadores executa simultaneamente sequências diferentes de instruções sobre conjuntos de dados diferentes. Exemplo: SMPs, clusters, sistemas NUMA.

2. (2,0) Considerando os diversos tipos de endereçamentos de instruções:

- a) Projete um mecanismo de endereçamento que permita que um conjunto arbitrário de 128 endereços, não necessariamente contíguos, em um grande espaço de endereçamento, seja especificável em um campo de 7 bits.

Uma solução seria usar endereçamento por registrador base mais deslocamento. Por exemplo, teríamos 2 bits para especificar um registrador e 5 bits para especificar um deslocamento. Poderíamos, assim, usar 4 registradores, cada um com até 32

deslocamentos possíveis, fornecendo 128 endereços diferentes.

- b) Analise os modos de endereçamento base+deslocamento e indexado, estabelecendo diferenças de aplicação, vantagens e desvantagens de cada um.

*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. Já o **modo indexado** é utilizado para manipulação de estrutura de dados.*

*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.*

Em relação aos outros modos de endereçamento, os modos indexado e base mais deslocamento tem um desempenho melhor na execução das instruções de acesso aos dados (a alteração dos endereços é realizada na UCP). O modo indexado tem um desempenho melhor que o base + deslocamento devido a inclusão do registrador base no cálculo.

Vantagens do modo indexado: Foi criado principalmente para trabalhar com vetores, pois os valores destes são armazenados sequencialmente na memória e que a sua localização pode ser referenciado por ponteiro.

Vantagens do modo base + deslocamento: Criado com o propósito de reduzir o tamanho das instruções e facilitar o processo de realocação dinâmica de programas. Outra vantagem é que em vez de ser necessário que o campo operando tenha um tamanho correspondente à capacidade total de endereçamento da MP, basta que o endereço desejado seja obtido pela soma de um valor existente em um dos registradores da CPU com o valor contido na instrução.

3. (2,0) Crie um conjunto de instruções de dois operandos, definidas em Linguagem Assembly, necessárias para a realização de operações aritméticas e elabore um programa para o cálculo da seguinte equação:

$$Y = (B/A + B)*C) - A$$

OBS: Elabore o programa em Assembly de forma que os conteúdos dos endereços A, B e C sejam mantidos. Utilize endereços temporários, como T1, T2, etc.

CONJUNTO DE INSTRUÇÕES PARA 2 OPERANDOS:

```
ADD X,Y => (X) ← (X) + (Y)
SUB X,Y => (X) ← (X) - (Y)
MUL X,Y => (X) ← (X) * (Y)
DIV X,Y => (X) ← (X) / (Y)
MOV X,Y => (X) ← (Y)
```

$$Y = (B/A + B)*C) - A$$

```
MOV T1,B => (T1) ← (B)
DIV T1,A => (T1) ← (T1) / (A)
ADD T1,B => (T1) ← (T1) + (B)
MUL T1,C => (T1) ← (T1) * (C)
SUB T1,A => (T1) ← (T1) - (A)
MOV Y,T1 => (Y) ← (T1)
```

4. (2,0) Considere um computador que utiliza 24 bits para representar números inteiros e

números em ponto flutuante. Na representação para **ponto flutuante**, como na representação IEEE 754, o bit mais à esquerda representa o sinal e deve ser 0 para números positivos e 1 para números negativos, os próximos 8 bits representam o expoente e os próximos 15 bits representam a parte fracionária da mantissa. Quando todos os bits do expoente são iguais a 0 ou iguais a 1, representam-se os casos especiais referentes a números denormalizados, infinito e Not a Number. Caso contrário, o número a ser representado deve ser expresso na notação científica normalizada $\pm(1,b_1b_2b_3..b_{15})_2 \times 2^{\text{expoente}}$. O expoente deve ser representado em excesso de 128.

A representação é mostrada na figura abaixo.

S	Expoente	Mantissa
1	8 bits	15 bits

a) (1,0) Considere o seguinte conjunto de 24 bits:

110000101110000000000000

Indique o valor **em decimal (deixe as contas indicadas)** para este conjunto de bits quando considerarmos que ele está representando:

i. um inteiro sem sinal

$$2^{23} + 2^{22} + 2^{17} + 2^{15} + 2^{14} + 2^{13} = 12.771.328$$

ii. um inteiro representado em complemento a 2

$$-2^{23} + (2^{22} + 2^{17} + 2^{15} + 2^{14} + 2^{13}) = -4.005.888$$

iii. um número em ponto flutuante utilizando a representação do enunciado

1 10000101 1100000000000000

Sinal: 1 (negativo)

Expoente: $10000101_2 = 133 - 128 = +5_{10}$ (excesso de 128)

Mantissa: 1100000000000000

$$\text{Resposta: } -1,11 \times 2^{+5} = -111000_2 = -56$$

b) (0,6) Indique o maior e o menor valor **em decimal (deixe as contas indicadas)** que pode ser representado pelos 24 bits para cada uma das representações abaixo:

i. inteiro sem sinal

$$\begin{aligned} \text{Maior valor: } 111111111111111111111111 &= \\ 2^{23} + 2^{22} + 2^{21} + 2^{20} + 2^{19} + 2^{18} + 2^{17} + 2^{16} + 2^{15} + 2^{14} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 \\ + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 &= +16.777.215 \end{aligned}$$

$$\text{Menor valor: } 000000000000000000000000 = 0$$

ii. inteiro em complemento a 2

$$\begin{aligned} \text{Maior valor: } 011111111111111111111111 &= \\ 2^{22} + 2^{21} + 2^{20} + 2^{19} + 2^{18} + 2^{17} + 2^{16} + 2^{15} + 2^{14} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^5 \\ + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 &= +8.388.607 \end{aligned}$$

$$\text{Menor valor: } 100000000000000000000000 = -2^{15} = -8.388.608$$

iii. número expresso na notação **científica normalizada** na representação ponto flutuante definida no enunciado

$$\begin{aligned} \text{Maior valor:} \\ 0 \ 11111110 \ 1111111111111111 &= 1,1111111111111111 \times 2^{+126} = +1,7 \times 10^{+38} \end{aligned}$$

$$\begin{aligned} \text{Menor valor:} \\ 1 \ 11111110 \ 1111111111111111 &= -1,1111111111111111 \times 2^{+126} = -1,7 \times 10^{+38} \end{aligned}$$

c) (0,4) Indique a representação em ponto flutuante utilizando a representação do enunciado para os seguintes valores decimais:

i. +12,25

$$1100,01 \times 2^0 = 1,10001 \times 2^{+3}$$

Sinal: 0 (positivo)

Expoente $+3 + 128 = +131_{10} = 10000011_2$ (excesso de 128)

Mantissa: 10001

Resposta: 0 10000011 1000100000000000

ii. -0,8

$$0,1100110011001100110011... \times 2^0 = 1,10011001100110011... \times 2^{-1}$$

Sinal: 1 (negativo)

Expoente $-1 + 128 = 127 = 01111111_2$ (excesso de 128)

Mantissa: 100110011001100110011...

Resposta: 1 01111111 100110011001100

5. (2,0) Considere uma máquina cujo controlador de disco rígido possui três registradores para se comunicar com o resto do sistema: um para receber comandos do sistema (pedido para enviar um byte para o disco ou receber um byte do disco, por exemplo), outro para indicar se o disco está pronto para receber bytes do sistema ou enviar bytes para o sistema, e outro para armazenar um byte enviado pelo sistema ou para armazenar um byte do disco. O primeiro registrador possui o endereço 70, o segundo 71 e o terceiro 72. Para se enviar um byte para o disco, deve-se enviar o comando 20 para o controlador de disco, utilizando-se o endereço 70. O controlador de disco armazena o valor 01 no registrador de estado (registrador 71) quando o byte puder ser enviado para o disco e o byte deve ser disponibilizado pelo sistema no registrador 72. Descreva **detalhadamente** como será realizado o envio de **5000** bytes do sistema para o controlador de disco para cada um dos três seguintes métodos de comunicação: (1) por E/S programada, (2) por interrupção e (3) por acesso direto à memória. Você deve descrever estes métodos dentro do contexto do ambiente descrito neste enunciado. **Não serão consideradas descrições gerais dos três métodos.**

E/S programada:

A UCP envia o comando 20 para o registrador do controlador do disco rígido com endereço 70. Depois, fica lendo o conteúdo do registrador com endereço 71 e verificando se o seu conteúdo é igual a 01. Quando o conteúdo for igual a 01, a UCP envia, pelo barramento de dados, um byte para o registrador do controlador com endereço 72. Este procedimento é repetido para enviar cada byte dos 4999 bytes seguintes a serem enviados para o controlador do disco.

E/S por interrupção:

A UCP envia o comando 20 para o registrador com endereço 70. Depois, vai executar outras instruções. Quando o controlador de disco rígido está pronto para receber o byte, ele envia um sinal de interrupção para a UCP. A UCP, ao receber o sinal de interrupção, finaliza a instrução que estiver executando, salva o contexto do programa que estava sendo executado e atende a interrupção, encaminhando o byte a ser gravado no disco para o registrador de endereço 72 do controlador de disco. Este procedimento é repetido para enviar cada byte dos 4999 bytes seguintes a serem enviados para o disco rígido.

E/S por acesso direto à memória:

Neste caso, existe um controlador de DMA (Direct Access Memory) que é responsável pela transferência dos bytes. A UCP informa a este controlador que a operação de transferência deverá ser realizada com o controlador do disco rígido, o endereço inicial da memória onde devem ser lidos os bytes a serem enviados, o número de bytes a serem enviados (5000 bytes) e a indicação de que o controlador de DMA deve transferir os bytes da memória para o controlador do disco rígido. Após receber estas informações, o controlador de DMA realiza a transferência dos bytes entre a memória e o controlador do disco rígido sem a intervenção da UCP. Após a transferência de todos os bytes, o controlador de DMA avisa o fim da operação para a UCP através de um sinal de interrupção.