



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina: Organização de Computadores**

**AP3 2º semestre de 2014.**

**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.
  6. Respostas não justificadas não serão consideradas.
- 

1. (2,5) Um computador, que apresenta uma arquitetura similar àquela apresentada ao longo do curso, possui uma capacidade máxima de memória principal de 128 M células, cada uma capaz de armazenar uma palavra de 16 bits. Em cada acesso à memória, realiza-se o acesso a uma célula. As instruções desta máquina são compostas de 2 campos: código de operação e endereço da célula de memória do operando. Sabe-se que o tamanho de uma instrução é 32 bits.

- a) (0,7) Calcule o número de códigos de operação diferentes que este computador pode ter.

*O barramento da memória (BE) deverá permitir endereçar as 128M células ( $N = 2^{27}$ ).*

*BE = 27 bits*

*Tamanho da instrução = 32 bits, então, Cód.oper. + Operando (Op) = 32 bits*

*Op é um endereço de memória, necessitando de no mínimo 27 bits.*

*Cód.oper = 32 bits – 27bits = 5 bits*

*Este computador pode ter 32 ( $2^5$ ) códigos diferentes (ou instruções diferentes)*

- b) (0,6) Indique o número de acessos à memória necessários para se obter uma instrução.

*Serão necessários 2 acessos à memória, já que uma instrução possui 32bits e*

ocupa 2 células (16bits em cada célula) e a cada acesso apenas uma célula é transferida no barramento

- c) (1,2) Descreva detalhadamente a execução da instrução **MUL Op.** e **JN Op.** nesta máquina. A instrução **MUL Op.** multiplica o conteúdo da célula de memória cujo endereço é **Op.** pelo conteúdo do acumulador e armazena o resultado na memória no endereço **Op.** A instrução **JN Op.**, quando executada, carrega CI com o valor de **Op.** se o conteúdo do Acumulador é menor que zero, e caso contrário carrega CI com CI+1.

### **MUL Op.**

- 1)  $RI \leftarrow (CI)$  , ou seja,  $RI \leftarrow$  recebe a instrução contida no endereço contido no CI
- 2)  $CI \leftarrow CI + 1$
- 3) Decodificação do código de operação
  - recebe os bits do código de operação
  - produz sinais para a execução da operação de subtração
- 4) Execução da operação
  - A UC emite sinais para que o valor do campo operando (Op.) seja transferido para a REM
  - A UC emite sinais para que o valor contido no REM seja transferido para o barramento de endereços
  - A UC ativa a linha READ do barramento de controle
  - Conteúdo da posição da memória, contido no barramento de endereços, é transferido através do barramento de dados para o RDM
  - UC emite sinais para transferir conteúdo do acumulador para UAL, ( $UAL \leftarrow ACC$ ) 1o. termo
  - O conteúdo do RDM é transferido para o registrador acumulador ( $ACC \leftarrow RDM$ )
  - UC emite sinais para transferir conteúdo do acumulador para UAL. ( $UAL \leftarrow ACC$ ) 2o. termo
  - A UC emite sinais para a UAL executar da operação de multiplicação
  - A UC emite sinais para a UAL liberar resultado e armazenar no acumulador ( $ACC \leftarrow UAL$ )

### **JN Op.**

- 1)  $RI \leftarrow (CI)$  , ou seja,  $RI \leftarrow$  recebe a Instrução contida no endereço contido no CI
- 2)  $CI \leftarrow CI + 1$
- 3) Decodificação do código de operação
  - recebe os bits do código de operação
  - produz sinais para a execução da operação de salto condicional
- 4) Execução da operação
  - UC emite sinal para transferir conteúdo do acumulador para UAL  
 $\rightarrow UAL \leftarrow ACC$
  - Executa operação de comparação  
 $\rightarrow$  Se Resultado = verdadeiro, isto é,  $ACC > 0$ ,  $CI \leftarrow 300$
- 5) Inicia o procedimento de leitura da instrução contida no endereço que consta em CI

2. (2,5) Considere o conjunto de 32 bits representado na base hexadecimal (DB480000)<sub>16</sub>. Mostre o que ele representa, **em decimal**, quando for interpretado como:

$$(DB480000)_{16} = (1101\ 1011\ 0100\ 1000\ 0000\ 0000\ 0000\ 0000)_2$$

- a) (0,3) um inteiro sem sinal.  
 $2^{31} + 2^{30} + 2^{28} + 2^{27} + 2^{25} + 2^{24} + 2^{22} + 2^{19} = 3.678.928.896$
- b) (0,3) um inteiro utilizando-se a representação sinal e magnitude.  
 $-(2^{30} + 2^{28} + 2^{27} + 2^{25} + 2^{24} + 2^{22} + 2^{19}) = -1.531.445.248$
- c) (0,5) um inteiro utilizando-se a representação em complemento a 2.  
 $-2^{31} + (2^{30} + 2^{28} + 2^{27} + 2^{25} + 2^{24} + 2^{22} + 2^{19}) = -616.038.400$
- d) (0,8) um número utilizando-se a representação ponto flutuante precisão simples IEEE 754 (1 bit de sinal, 8 bits para expoente em excesso de 127, 23 bits para mantissa).

$(1\ 10110110\ 100100000000000000000000)_2$

Sinal: 1 (negativo)

Expoente:  $10110110$  (excesso de 127) =  $182 - 127 = +55$

Mantissa:  $100100000000000000000000$

Resposta:  $-1, 100100000000000000000000 \times 2^{+55} = -(2^{+55} + 2^{+54} + 2^{+51})$

$(1\ 10110110\ 100100000000000000000000)_2 = -5,63 \times 10^{+16}$

- e) (0,6) Qual o menor e o maior valor positivo de números expressos na notação científica normalizada que podem ser representados utilizando-se a representação em ponto flutuante, descrita no item “d”, para este computador de 32 bits? Os valores devem ser representados em decimal.

Menor positivo  $0\ 00000001\ 000000000000000000000000 =$

$1, 000000000000000000000000 \times 2^{-126} = +1,175 \times 10^{-38}$

Maior positivo  $0\ 11111110\ 111111111111111111111111 =$

$1, 111111111111111111111111 \times 2^{+127} = +3,40 \times 10^{+38}$

3. (2,5) Considere uma máquina que possa endereçar 128 Mbytes de memória física, utilizando endereço referenciando byte, e que tenha a sua memória organizada em blocos de 8 bytes. Ela possui uma memória cache que pode armazenar 2 K blocos, sendo um bloco por linha. Mostre o formato da memória cache, indicando os campos necessários (tag, bloco) e o número de bits para cada campo, e o formato de um endereço da memória principal, indicando os bits que referenciam os campos da cache, para os seguintes mapeamentos:

- ☞ a) Mapeamento direto.

Memória principal:

A máquina permite endereçar 128 Mbytes, como endereço referenciado a byte, temos  $N = 128\text{ Mcélulas}$

$B = \text{Total de blocos} = 128\text{ Mbytes} / 8\text{ bytes/bloco} = 16\text{ Mblocos}$

Tamanho do endereço da MP (E)  $\Rightarrow N = 2^E \Rightarrow 128\text{ Mcélulas} = 2^{27} \Rightarrow E = 27\text{ bits}$

Memória Cache

$Q = 2K\text{ blocos (1 bloco por linha)} = 2\text{ K linhas}$

Campos do endereço:

**Tag** =  $B / Q = 16 \text{ Mblocos} / 2 \text{ Linhas} = 8K \Rightarrow$  necessário 13 bits  
**Linha** = total de linhas =  $Q = 2K \Rightarrow$  necessário 11 bits  
**Palavra** = total de 8  $\Rightarrow$  necessário 3 bits

Tag = 13 bits	No.linha = 11bits	Palavra = 3 bits
---------------	-------------------	------------------

Endereço da MP = 27 bits

#### b) Mapeamento totalmente associativo.

Memória principal:

A máquina permite endereçar 128 Mbytes, como endereço referenciado a byte, temos  $N = 128 \text{ Mcélulas}$

$B = \text{Total de blocos} = 128 \text{ Mbytes} / 8 \text{ bytes/bloco} = 16 \text{ Mblocos}$

Tamanho do endereço da MP (E)  $\Rightarrow N = 2^E \Rightarrow 128 \text{ Mcélulas} = 2^{27} \Rightarrow E = 27 \text{ bits}$

Memória Cache

$Q = 2K \text{ blocos (1 bloco por linha)} = 2 \text{ K linhas}$

Campos do endereço:

**Tag** =  $B = 16 \text{ Mblocos} \Rightarrow$  necessário 24 bits  
**Palavra** = total de 8  $\Rightarrow$  necessário 3 bits

tag = 24 bits	Palavra = 3 bits
---------------	------------------

Endereço da MP = 27 bits

#### 4. (2,5) Descreva os modos de endereçamento de operandos, explicando suas vantagens e desvantagens.

**Imediato:** O campo operando contém o dado, desta forma o dado é transferido da memória juntamente com a instrução.

**Vantagem:** Rapidez na execução da instrução, pois não requer acesso à memória principal, apenas na busca da própria instrução.

**Desvantagem.** Limitação do tamanho do campo operando das instruções reduzindo o valor máximo do dado a ser manipulado. Trabalho excessivo para alteração de valores quando o programa é executado repetidamente e o conteúdo das variáveis serem diferentes em cada execução.

**Direto:** O campo operando da instrução contém o endereço onde se localiza o dado.

**Vantagem.** Flexibilidade no acesso a variáveis de valor diferente em cada execução do programa

**Desvantagem.** Limitação de memória a ser usada conforme o tamanho do operando.

**Indireto:** O campo de operando contém o endereço de uma célula, sendo o valor contido nesta célula o endereço do dado desejado.

**Vantagem:** Usar como "ponteiro". Elimina o problema do modo direto de limitação do valor do endereço do dado. Manuseio de vetores (quando o modo indexado não está disponível).

**Desvantagem:** Muitos acesso à MP para execução, requer pelo menos 2 acessos à memória principal.

**Por registrador:** característica semelhante aos modos direto e indireto, exceto que a célula (ou palavra) de memória referenciada na instrução é substituída por um dos registradores da UCP. O endereço mencionado na instrução passa a ser o de um dos registradores.

**Vantagens:** Menor quantidade de bits para endereçar os registradores, por consequência, redução da instrução. E o dado pode ser armazenado em um meio mais rápido (registrador).

**Desvantagens:** Devido ao número reduzido de registradores existentes na UCP causa uma dificuldade em se definir quais dados serão armazenados nos registradores e quais permanecerão na UCP.

**No modo indexado:** consiste em que o endereço do dado é a soma do valor do campo operando (que é fixo para todos os elementos de um dado vetor) e de um valor armazenado em um dos registradores da UCP (normalmente denominado registrador índice).

**Vantagem:** Rapidez de execução das instruções de acesso aos dados, visto que a alteração do endereço dos elementos é realizada na própria UCP

*Exemplo:*

*No modo de endereçamento base mais deslocamento o endereço é obtido da soma do campo de deslocamento com o conteúdo do registrador base. Este modo de endereçamento tem como principal objetivo permitir a modificação de endereço de programas ou módulos destes , bastando para isso alterar o registrador base.*

*Vantagem: Reduz o tamanho das instruções e facilita o processo de relocação de programas.*