

GABARITO DA AP3 - 1º sem de 2016 ORGANIZAÇÃO DE COMPUTADORES

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 a transferência de uma palavra. 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) (0,2) Qual é o maior endereço em decimal desta memória?

$$\text{Maior endereço: } N-1 = 128M - 1 = 134.217.728 - 1 = 134.217.727$$

- b) (0,3) Qual é o tamanho do barramento de endereços deste sistema?

$$\text{Tamanho do barramento} = e$$

$$e = \log_2 N = \log_2 128M \text{ células} = \log_2 2^{27} = 27 \text{ bits}$$

$$\text{Então, tamanho do barramento de endereços} = 27 \text{ bits}$$

- c) (0,6) Quantos bits são armazenados no RDM (Registrador de Dados da Memória), no REM (Registrador de Endereços da Memória) e no CI (Contador de Instruções) ?

$$\text{REM} = \text{barramento de endereços (BE)} = 27 \text{ bits}$$

$$\text{RDM} = \text{tamanho da palavra} = 16 \text{ bits}$$

$$\text{CI} = \text{quantidade de bits para endereçar toda a memória} = \text{BE} = 27 \text{ bits}$$

- d) (0,5) Calcule o número de códigos de operação diferentes que este computador pode ter

$$\text{operando} = \text{endereço de uma célula} = 27 \text{ bits}$$

$$\text{Tamanho da instrução} = \text{código de operação} + 1 \text{ operando} \Rightarrow$$

$$32 = \text{código de operação} + 27 \text{ bits}$$

$$\text{cod.operação} = 5 \text{ bits (permite até 32 códigos diferentes)}$$

- e) (0,2) Qual é o número máximo de bits que pode existir na memória?

$$\text{Total de bits} = T$$

$$T = N \times M \Rightarrow T = 128M \text{ células} \times 16 \text{ bits/célula} \Rightarrow T = 2048 \text{ Mbits ou } 2 \text{ Gbits}$$

- f) (0,7) Descreva detalhadamente a execução da instrução **ADD Op.** nesta máquina. A instrução **ADD Op.** soma o conteúdo do acumulador com o conteúdo da célula de memória cujo endereço é **Op.** e armazena o resultado na memória no endereço **Op.**

Passo 1: A CPU coloca no REM o valor do operando (REM <- Op), e é disponibilizado no barramento de endereço

Passo 2: A CPU aciona pelo barramento de controle a leitura de memória

Passo 3: A memória coloca o valor no barramento de dados, e por consequência no RDM da CPU (RDM <- MP(Op))

Passo 4: A CPU executa a soma do valor recebido com o acumulador armazenando no acumulador; ACC <- ACC + RDM (ou ACC <- ACC + MP(Op))

Passo 5: CI é incrementado (CI <- CI + 1) para apontar para a próxima instrução a ser lida

2. (2,5) Considere uma máquina que utiliza 32 bits para representar números em ponto fixo e em ponto flutuante.

- a) (0,7) Mostre a representação de -245,0 utilizando-se a representação ponto flutuante precisão dupla IEEE 754 (1 bit de sinal, 8 bits para expoente em excesso de 127, e 23 bits para mantissa)

$$-245,0_{10} = -11110101,0_2 = -1,11101010 \times 2^{+7}$$

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

$$\text{Expoente} = +7 + 127 = 134_{10} = 10000110_2$$

$$\text{Mantissa} = ,11101010$$

$$\text{Temos então: } 1 \ 10000110 \ 111010100000000000000000$$

- b) Para o conjunto de bits obtido no item anterior, indique o que ele representa na base 10, considerando-se as seguintes representações: **(Não precisa fazer as contas, deixe-as indicadas)**:

$$(11000011011101010000000000000000)$$

- i. (0,4) um inteiro sem sinal

$$2^{31} + 2^{30} + 2^{25} + 2^{24} + 2^{22} + 2^{21} + 2^{20} + 2^{18} + 2^{16} = 3.279.224.832$$

- ii. (0,6) um inteiro utilizando-se a representação em complemento a 2

$$-2^{31} + (2^{30} + 2^{25} + 2^{24} + 2^{22} + 2^{21} + 2^{20} + 2^{18} + 2^{16}) = -1.015.742.464$$

- c) (0,8) 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 “a”, para este computador? Os valores devem ser representados em decimal (pode deixar as contas indicadas).

$$\begin{aligned} \text{maior valor positivo: } & 0 \ 1111110 \ 111111111111111111111111 = \\ & = 1, 111111111111111111111111 \times 2^{+127} = +3,40 \times 10^{+38} \end{aligned}$$

$$\begin{aligned} \text{menor valor positivo: } & 0 \ 00000001 \ 000000000000000000000000 = \\ & = 1, 000000000000000000000000 \times 2^{-126} = +1,18 \times 10^{-38} \end{aligned}$$

3.(2,5) Considere uma máquina que possa endereçar 256 Mbytes de memória física, utilizando endereço referenciando byte, e que tenha a sua memória organizada em blocos de 16 bytes. Ela possui uma memória cache que pode armazenar 4K 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 256 Mbytes, como endereço referenciado a byte, temos $N = 256 \text{ Mcélulas}$

$$B = \text{Total de blocos} = 256 \text{ Mbytes} / 16 \text{ bytes/bloco} = 16 \text{ Mblocos}$$

$$\text{Tamanho do endereço da MP (E)} \Rightarrow N = 2^E \Rightarrow 256 \text{ Mcélulas} = 2^{28} \Rightarrow E = 28 \text{ bits}$$

Memória Cache

$$Q = 4K \text{ blocos (1 bloco por linha)} = 4K \text{ linhas}$$

Campos do endereço:

$$\text{Tag} = B / Q = 16 \text{ Mblocos} / 4K \text{ linhas} = 4K \Rightarrow \text{necessário } 12 \text{ bits}$$

$$\text{Linha} = \text{total de linhas} = Q = 4K \Rightarrow \text{necessário } 12 \text{ bits}$$

$$\text{Palavra} = \text{total de } 16 \Rightarrow \text{necessário } 4 \text{ bits}$$

Tag = 12 bits	No.linha = 12bits	Palavra = 4 bits
---------------	-------------------	------------------

Endereço da MP = 28 bits

- b) Mapeamento totalmente associativo.

Memória principal:

A máquina permite endereçar 256 Mbytes, como endereço referenciado a byte, temos $N = 256 \text{ Mcélulas}$

$$B = \text{Total de blocos} = 256 \text{ Mbytes} / 16 \text{ bytes/bloco} = 16 \text{ Mblocos}$$

$$\text{Tamanho do endereço da MP (E)} \Rightarrow N = 2^E \Rightarrow 256 \text{ Mcélulas} = 2^{28} \Rightarrow E = 28 \text{ bits}$$

Memória Cache

$$Q = 4K \text{ blocos (1 bloco por linha)} = 4K \text{ linhas}$$

Campos do endereço:

$$\text{Tag} = B = 16 \text{ Mblocos} \Rightarrow \text{necessário } 24 \text{ bits}$$

$$\text{Palavra} = \text{total de } 16 \Rightarrow \text{necessário } 4 \text{ bits}$$

tag = 24 bits	Palavra = 4 bits
---------------	------------------

Endereço da MP = 28 bits

4.(2,5) Considerando os diversos tipos de endereçamentos de instruções:

- a) Projete um mecanismo de endereçamento que permita que um conjunto arbitrário de 64 endereços, não necessariamente contíguos, em um grande espaço de endereçamento, seja especificável em um campo de 6 bits.

A solução será a utilização do endereçamento por registrador base mais deslocamento. Por exemplo, teríamos 2 bits para especificar um registrador e 4 bits para especificar um deslocamento. Poderíamos, assim, usar 4 registradores, cada um com até 16 deslocamentos possíveis, fornecendo 64 endereços diferentes.

- b) Analise os modos de endereçamento base mais deslocamento e indexado, estabelecendo diferenças de desempenho, 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.

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 mantém o índice do array.

O modo indexado possui como vantagem a rapidez de execução das instruções de acesso aos dados, visto que a alteração do endereço dos elementos é realizada no próprio processador. Possui como desvantagem o uso de um registrador específico como registrador-índice e a limitação do tamanho do vetor de índice.

O modo base mais deslocamento tem a vantagem na redução do tamanho das instruções economizando memória, e facilitando o processo de relocação dinâmica de programas. Possui a desvantagem do uso de um registrador específico como registrador base.