



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Organização de Computadores

GABARITO - AP3 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. (1,5) Um computador, que apresenta uma arquitetura similar àquela apresentada ao longo do curso, pode endereçar 2M células de memória, sendo que cada célula de memória é capaz de armazenar uma palavra de 32 bits. Em cada acesso à memória, uma célula é acessada e realiza-se a transferência de uma palavra.

- a) (0,2) Qual é o tamanho do barramento de endereços deste sistema?

$N = 2M \text{ células} = 2^{21} \text{ células}$
 $\text{Como } e = \log_2 N = \log_2 2^{21} \Rightarrow e = 21 \text{ bits}$
Então, tamanho do barramento de endereços = 21 bits

- b) (0,3) Quantos bits podem ser armazenados no RDM (Registrador de Dados da Memória), no REM (Registrador de Endereços da Memória) e no CI (Contador de Instruções)?

RDM = tamanho da palavra transferida no processo de leitura de memória, como cada célula armazena uma palavra, então, RDM = tamanho da palavra = tamanho da célula = 32 bits

REM = barramento de endereços = 21 bits
CI = barramento de endereços = 21 bits

- c) (1,0) Descreva detalhadamente a execução da instrução **ADD Op.** nesta máquina, indicando como o REM, RDM, CI, barramentos de dados, de endereços e de controle são utilizados. A instrução **ADD Op.** soma o conteúdo da célula de memória cujo endereço é Op. com o valor armazenado no acumulador e coloca o resultado no acumulador.

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

Passo 2: A CPU aciona o sinal de leitura de memória no barramento de controle

Passo 3: A memória coloca o conteúdo do endereço Op no barramento de dados, que é transferido para o RDM da CPU (RDM <- MP(Op))

Passo 4: A CPU executa a soma do valor recebido no RDM com o valor armazenado 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. (2,0) Considere o conjunto de 32 bits representado na base hexadecimal $(80A00000)_{16}$. Mostre o que ele representa, **em decimal**, quando for interpretado como:

OBS: Não precisa fazer as contas, deixe-as indicadas.

$$(80A00000)_{16} = (10000000\ 10100000\ 00000000\ 00000000)_2$$

- a) (0,2) um inteiro sem sinal

$$2^{31} + 2^{23} + 2^{21} = 2.157.969.408$$

- b) (0,4) um inteiro utilizando-se a representação sinal e magnitude

$$-(2^{23} + 2^{21}) = -10.485.760$$

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

$$-2^{31} + (2^{23} + 2^{21}) = -2.136.997.888$$

- 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)

$$1\ 00000001\ 010000000000000000000000$$

Sendo:

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

$$\text{Expoente} = 00000001 = 1 - 127 \Rightarrow \text{expoente} = -126$$

$$\text{Mantissa} = 010000000000000000000000$$

$$\text{Temos então} \Rightarrow -1,010000000000000000000000 \times 2^{-126} = -1,01 \times 2^{-126} = -(2^{-126} + 2^{-128})_{10}$$

3. (1,5) Descreva os três possíveis métodos de comunicação entre uma interface de entrada e saída com a unidade central de processamento e memória principal: por programa (*polling*), por interrupção e por acesso direto à memória.

Por programa: A UCP indica à interface de entrada e saída que deseja realizar uma operação de transferência de dados e fica interrogando a interface para saber se ela está pronta para realizar a transferência de dados. Quando a UCP recebe uma resposta positiva da interface, ela realiza a transferência de dados. Para ler dados da interface e colocar os dados na memória, ela realiza operações de leitura de dados da interface e escrita na memória. Para escrever dados na interface, ela realiza operações de leitura da memória e escrita na interface.

Por interrupção: A UCP indica à interface de entrada e saída que deseja realizar uma operação de transferência de dados e realiza outras instruções que não se referenciam a esta operação, ou seja, a UCP não fica interrogando a interface para identificar quando ela está pronta. Quando a interface está pronta para realizar a transferência, ela gera um sinal de interrupção que é recebido pela UCP. A UCP ao receber este sinal, termina de realizar a instrução que estava sendo realizada, salva o contexto onde esta instrução estava sendo realizada, e executa as instruções para realizar a transferência de dados com a interface.

Por acesso direto à memória: Um controlador de acesso direto à memória realiza diretamente a transferência de dados entre a interface e a memória sem envolver a UCP nesta transferência. A UCP necessita enviar alguns parâmetros para o controlador: o endereço da interface, o tipo de transferência (escrita ou leitura de dados), o endereço de memória para ler ou escrever os dados e o número de bytes

a serem transferidos. O controlador de acesso direto à memória realiza toda a transferência de dados entre a interface e a memória e a UCP não necessita executar nenhuma instrução para realizar esta transferência. Quando a transferência acaba, o controlador gera um sinal de interrupção para a UCP indicando que a transferência foi realizada

4. (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 $N = 128 \text{ Mcélulas}$

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

$\text{Endereço} = E \Rightarrow N = 2^E \Rightarrow 128 \text{ Mcélulas} = 2^{27} \Rightarrow E = 27 \text{ bits}$

Memória Cache

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

Campos do endereço:

$\text{Tag} = B / Q = 16 \text{ Mblocos} / 2 \text{ Klinhas} = 8 \text{ K} = 13 \text{ bits}$

$\text{No. da linha} = Q = 2 \text{ K} = 11 \text{ bits}$

$\text{End da palavra} = 8 = 3 \text{ bits}$

Tag = 13 bits	No. linha = 11bits	End da palavra 3 bits
---------------	--------------------	--------------------------

Endereço = 27 bits

b) Mapeamento totalmente associativo.

Memória principal:

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

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

$\text{Endereço} = E \Rightarrow N = 2^E \Rightarrow 128 \text{ Mcélulas} = 2^{27} \Rightarrow E = 27 \text{ bits}$

Memória Cache

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

Campos do endereço:

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

$\text{End da palavra} = 8 = 3 \text{ bits}$

tag = 24 bits	End da palavra 3 bits
---------------	--------------------------

5. (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.

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

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.