

Curso de Tecnologia em Sistemas de Computação
Disciplina: Organização de Computadores
GABARITO - AP1 1º semestre de 2018.

1. (3,5) Suponha que você deve projetar uma máquina com as seguintes especificações:

- ☐ Capaz de endereçar 1 M células de memória principal, sendo que cada célula armazena 2 bytes.
- ☐ Deve possuir os registradores RDM (utilizado para enviar e receber dados para/de o barramento de dados), REM (utilizado para enviar endereços no barramento de endereços), CI (utilizado para indicar o endereço da instrução a ser lida da memória) e RI (utilizado para armazenar uma instrução).
- ☐ Cada instrução deve conter um código de operação, dois operandos e um registrador como mostrado abaixo:

Cód. Oper	Operando 1	Operando 2	Reg.
-----------	------------	------------	------

onde o Operando 1 e o Operando 2 são endereços da memória principal e Reg. é o identificador de um Registrador, sendo que a máquina possui 7 registradores.

- ☐ Deve poder ter um máximo de 29 códigos de operação diferentes.
- a) (0,5) Indique qual deve ser o tamanho mínimo em bits do REM
Memória com 1M células => $N = 1M$ células
tamanho mínimo do REM será o tamanho do barramento de endereços necessário para endereçar toda a memória.
Barramento de endereços (BE) = $\log_2 N = \log_2 1M = 20$ bits
REM = tamanho do BE = 20 bits
- b) (0,4) Indique qual deve ser o tamanho mínimo em bits do do barramento de endereços.
tamanho do BE = 20 bits
- c) (0,6) Calcule o número de células que uma instrução necessita para ser armazenada.
Cada instrução = código de operação + 1o operando + 2o operando + 1 registrador
1o e 2o operando = endereço de uma célula = 20 bits
registrador = para atender pelo menos 7 regs. são necessários 3 bits
(total de $2^3 =$ até 8 registradores)
cod.operação = tamanho necessário para 29 códigos diferentes = necessários 5 bits
tamanho da instrução = $5 + 20 + 20 + 3 = 48$ bits
- d) (0,6) Indique o tamanho do RDM e do barramento de dados.
RDM = barramento de dados = tamanho necessário para transferir uma instrução
RDM = barramento de dados = 48 bits (3 células).
- e) (0,6) Quantos acessos a memória são necessários para que Unidade Central de Processamento obtenha uma instrução da memória, dado que cada acesso é lida uma célula da memória principal.
Como a instrução possui 48bits, ocupando 3 células e como em cada acesso uma célula é transferida, conclui-se, então, que serão necessários 3 acessos
- f) (0,8) Calcule a capacidade de armazenamento em bits dos registradores RI e CI, utilizando-se os valores calculados nos itens anteriores.
CI = tamanho necessário para endereçar toda a memória = 20 bits
RI = tamanho necessário para uma instrução = 48 bits

2 (2,5) Considere uma máquina que pode ter seu ciclo de busca e execução de uma instrução dividido em 5 estágios totalmente independentes: Busca de Instrução (BI), Decodificação (DI), Cálculo de Endereços de Operandos (CO), Execução (EX) e Escrita de Operandos (EO). Cada um dos estágios BI, EX e EO possui a duração de 10 ns e cada estágio DI e CO tem duração de 3 ns. Cada instrução desta máquina precisa executar os 5 estágios que serão sempre executados na sequência BI, DI, CO, EX e EO.

- a) (0,2) Uma implementação desta máquina foi realizada de modo que cada instrução deve ser completamente realizada em um único ciclo de relógio. Calcule a duração do ciclo de relógio que esta implementação deve possuir. Lembre-se que todas as instruções necessitam dos 5 estágios.

<i>BI</i> 10ns	<i>DI</i> 3ns	<i>CO</i> 3ns	<i>EX</i> 10ns	<i>EO</i> 10ns
-------------------	------------------	------------------	-------------------	-------------------

O ciclo de relógio deverá ter o tempo de 36ns

- b) (0,5) Como cada estágio é independente um do outro, deseja-se implementar uma nova arquitetura utilizando-se um pipeline de 5 estágios. Nesta nova implementação cada estágio do pipeline deve ser executado em um ciclo de relógio. Calcule a duração do ciclo de relógio que esta implementação pipeline deve possuir.

1°. Estágio	2°. Estágio	3°. Estágio	4°. Estágio	5°. Estágio
<i>BI</i> 10ns	<i>DI</i> 10ns	<i>CO</i> 10ns	<i>EX</i> 10ns	<i>EO</i> 10ns

Ciclo de relógio será igual ao tempo para execução do estágio com maior tempo de execução = 10ns

- c) (0,5) Considere um programa que necessita executar 1.200 instruções. Calcule o tempo de execução deste programa na máquina do item a e na máquina do item b.

Seja T_{ex} = tempo de execução de uma instrução = número de estágios \times ciclo de relógio (determinado nos itens anteriores)

Para o item a (sem pipeline) :

$$T_{ex} = 1 \text{ estágio de } 36ns = 36ns$$

$$T_{total} = 1200 \text{ instruções} \times T_{ex} = \mathbf{43.200 \text{ ns}}$$

Para o item b (pipeline: 5 estágios) :

$$T_{ex} = 5 \text{ estágios de } 10ns \text{ cada} = 50ns$$

$$T_{total} = T_{ex} + 1199 \times \text{tempo de 1 estágio}$$

$$T_{total} = 50ns + 1199 \times 10ns = \mathbf{12.040 \text{ ns}}$$

- d) (0,5) Considere um programa que necessita executar 120.000 instruções. Calcule o tempo de execução deste programa na máquina do item a e na máquina do item b.

Seja T_{ex} = tempo de execução de uma instrução = número de estágios \times ciclo de relógio (determinado nos itens anteriores)

Para o item a (sem pipeline) :

$$T_{ex} = 1 \text{ estágio de } 36ns = 36ns$$

$$T_{total} = 120.000 \text{ instruções} \times T_{ex} = \mathbf{4.320.000 \text{ ns}}$$

Para o item b (pipeline: 5 estágios) :

$$\begin{aligned}
T_{ex} &= 5 \text{ est\u00e1gios de } 10\text{ns cada} = 50\text{ns} \\
T_{total} &= T_{ex} + 119.999 \times \text{tempo de 1 est\u00e1gio} \\
T_{total} &= 50\text{ns} + 119.999 \times 10\text{ns} = \underline{\underline{1.200.040 \text{ ns}}}
\end{aligned}$$

- e) (0,8) Compare as diferen\u00e7as dos tempos de execu\u00e7\u00e3o obtidos pela maquina do item a e a do item b para os programas dos itens c e d.

O uso do pipeline permite que mais de uma instru\u00e7\u00e3o seja executada ao mesmo tempo reduzindo, assim, o tempo total na execu\u00e7\u00e3o dos programas. Esta afirma\u00e7\u00e3o pode observada nos tempos de execu\u00e7\u00e3o calculados nos itens c e d para a m\u00e1quina do item b (com pipeline) que \u00e9 bem inferior quando comparada com a m\u00e1quina do item a (sem pipeline).

2. (1,5) Considere um sistema de computa\u00e7\u00e3o que possui uma mem\u00f3ria principal (RAM) com capacidade m\u00e1xima de endere\u00e7amento de 64 K c\u00e9lulas, sendo que cada c\u00e9lula armazena um byte de informa\u00e7\u00e3o. Para criar um sistema de controle e funcionamento da sua mem\u00f3ria cache, a mem\u00f3ria RAM \u00e9 constitu\u00edda de blocos de 8 bytes cada. A mem\u00f3ria cache do sistema \u00e9 do tipo mapeamento direto, contendo 32 linhas. Pergunta-se:

a) Como seria organizado o endere\u00e7o da MP (RAM) em termos de etiqueta (tag), n\u00famero de linha e do byte dentro da linha?

Mem\u00f3ria Principal

=> Tamanho da mem\u00f3ria (em bytes) = 64Kc\u00e9lulas, cada 1 c\u00e9lula armazena 1 byte, temos $N = 64K$ c\u00e9lulas (ou 64 Kbytes)

=> Ser\u00e1 organizada em blocos de 8bytes, como 1c\u00e9lula = 1byte, temos cada bloco = 8 c\u00e9lulas, $K = 8$

=> $N = 64K$ c\u00e9lulas e $K = 8$ c\u00e9lulas por bloco, o total de blocos da MP (B) ser\u00e1:

Total de blocos: $B = N / K$ => $B = 64K\text{c\u00e9lulas} / 8 \text{ c\u00e9lulas por bloco}$ => $B = 8K\text{c\u00e9lulas}$

Mem\u00f3ria Cache

=> O K (quantidade de c\u00e9lulas/bloco) tem de ser igual a MP.

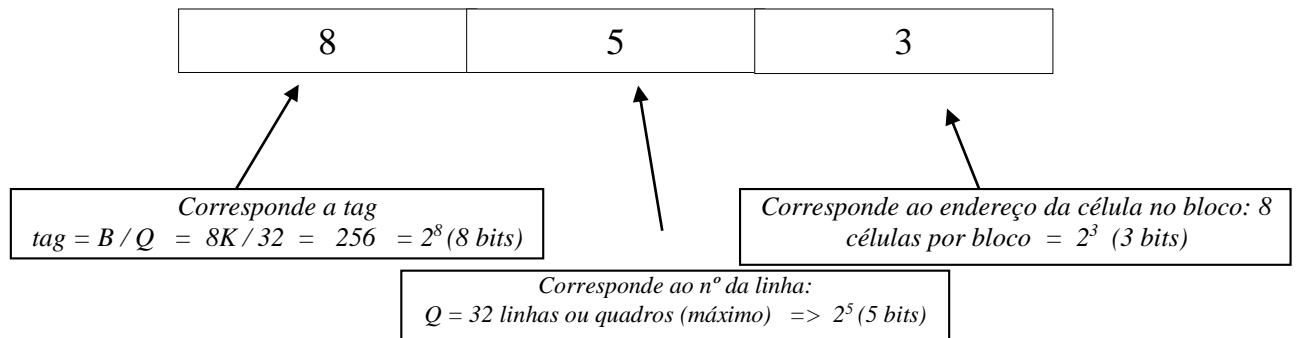
=> Tamanho da mem\u00f3ria cache (em blocos ou linhas) => $Q = 32$ linhas

=> Tamanho da mem\u00f3ria cach\u00ea em c\u00e9lulas = $Q \times K = 32 \text{ linhas} \times 8 \text{ c\u00e9lulas/linha} = 256 \text{ c\u00e9lulas}$ (ou 256 bytes)

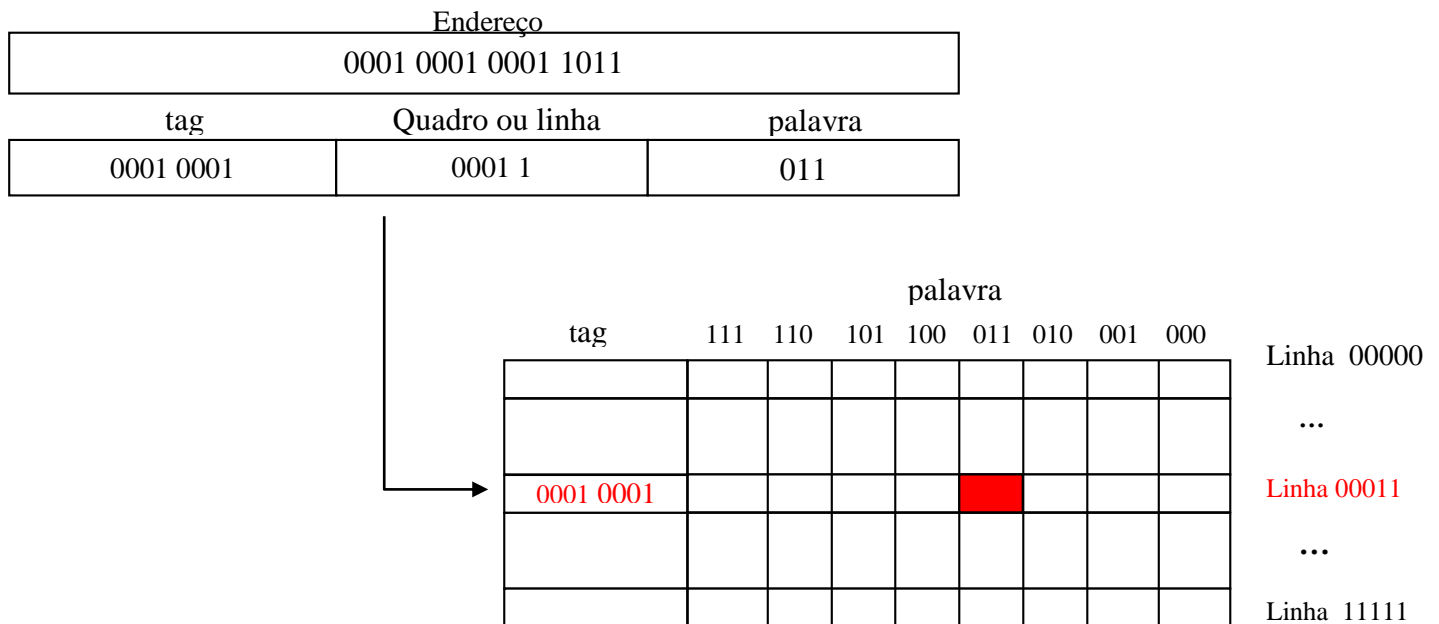
Mem\u00f3ria principal

=> Para endere\u00e7armos toda a MP precisamos da seguinte quantidade de bits (E)

sendo $N = 2^E$ => $N = 64K\text{c\u00e9lulas}$ => $N = 2^{16}$ => $E = 16$ bits



b) Em que linha estaria contido o byte armazenado no seguinte endereço da MP: 0001 0001 0001 1011?



c) Qual é capacidade da memória cache em bytes?

Capacidade da cache = $Q \times K = 32 \text{ linhas} \times 8 \text{ palavras/linha}$, considerando neste problema, tamanho da palavra = tamanho da célula e tamanho da célula = 1 byte, então:

Capacidade da cache = $32 \text{ linhas} \times 8 \text{ bytes} = 256 \text{ bytes}$ ou 2^8 bytes

3. (2,5) Descreva detalhadamente a hierarquia de memória nos sistemas computacionais atuais.

Podemos ilustrar essa hierarquia de memória na forma de uma pirâmide dividida em 4 níveis. No topo da pirâmide teríamos os registradores, que são pequenas unidades de memória que armazenam dados na UCP. São dispositivos de maior velocidade com tempo de acesso em torno de 1 ciclo de memória, menor capacidade de armazenamento além de armazenar as informações por muito pouco tempo.

Em um nível abaixo teríamos a memória cache, cuja função é acelerar a velocidade de transferência das informações entre UCP e MP e, com isso, aumentar o desempenho do sistema. A UCP procura informações primeiro na Cache. Caso não as encontre, as mesmas são transferidas da MP para a Cache. A cache possui tempo de acesso menor que a da Memória principal, porém com capacidade inferior a esta, mas superior ao dos registradores e o suficiente para armazenar uma apreciável quantidade de informações, sendo o tempo de permanência do dado menor do que o tempo de duração do programa a que pertence.

Abaixo da memória cache teríamos a memória básica de um sistema de computação, que é a memória principal. Dispositivo onde o programa (e seus dados) que vai ser executado é armazenado para que a UCP busque instrução por instrução para executá-las. A MP são mais lentas que a cache e mais rápidas que a memória secundária, possui capacidade bem superior ao da cache e os dados ou instruções permanecem na MP enquanto durar a execução do programa.

Finalmente, na base da pirâmide teríamos a memória secundária, memória auxiliar ou memória de massa, que fornece garantia de armazenamento mais permanente aos dados e programas do usuário. Alguns dispositivos são diretamente ligados: disco rígido, outros são conectados quando necessário: disquetes, fitas de armazenamento, CD-ROM. São os mais lentos em comparação com os outros níveis de memória, mas possuem a maior capacidade de armazenamento e armazenam os dados de forma permanente.