



UNICAMP - Faculdade de Engenharia Elétrica e computação

EA960 - Organização de Computadores

## Soluções/Respostas : Prova 1

**Monitor:** Caio Hoffman  
**Professor:** Ivan Luiz Marques Ricarte

Maio de 2010

---

**Questão 1**

Solução:

---

**a)**

$$T_{pf}^i = \alpha T_{pf}$$

$$\begin{aligned} T_{pf}^{ii} &= (1 - \sigma)T_{pf} + \beta\sigma T_{pf} \\ &= (1 - \sigma + \beta\sigma)T_{pf} \\ &= \left(1 + \sigma(\beta - 1)\right)T_{pf} \end{aligned}$$

Sendo  $\alpha, \beta < 1$ . Para que  $i$  resulte em melhor desempenho  $\Rightarrow$

$$\begin{aligned} T_{pf}^i &< T_{pf}^{ii} \Leftrightarrow \\ \alpha T_{pf} &< \left(1 + \sigma(\beta - 1)\right) T_{pf} \\ \sigma(1 - \beta) &< 1 - \alpha \\ \sigma &< \frac{1 - \alpha}{1 - \beta} \end{aligned}$$

Resposta:  $\sigma < \frac{1 - \alpha}{1 - \beta}$

**b)**

$$\begin{aligned} T_f &= 4 = \sigma T_{pf} = \sigma 10 \Leftrightarrow \\ \sigma &= 0,4 \\ \frac{0,25}{1 - \beta} &< 0,4 \\ -0,15 &< -0,4\beta \Leftrightarrow \\ \frac{0,15}{0,4} &> \beta \end{aligned}$$

Resposta:  $\beta < 0,375$

---

---

**Questão 2**

Solução:

**a)**

- 32 Bytes por linha  $\Rightarrow$  5 bits de identificação de palavra na linha (*offset*)

- $256 \text{ KiBytes} = 2^{18} \text{ bytes} \Rightarrow \frac{2^{18} \text{ bytes}}{2^5 \text{ bytes/linha}} = 2^{13} \text{ linhas}$

- Associatividade 8  $\Rightarrow$  8linhas/grupo  $\therefore \frac{2^{13} \text{ linhas}}{2^3 \text{ linhas/grupo}} = 2^{10}$  grupos, desse modo 10 bits de índice

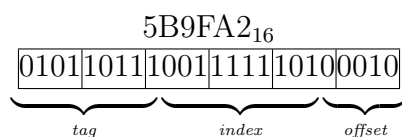
- Sobram 9 bits que vão de  $000000000_2$  à  $111111111_2$ .

Resposta: Faixa de valores é de  $000_{16}$  à  $1FF_{16}$ .**b)**

Considerando que não há bits adicionais como de LRU, validade, escrita, etc. Temos:

$$(8 \text{ tags/grupo} \cdot 9 \text{ bits/tag}) \cdot 2^{10} \text{ grupos} = 72 \text{ Kibits}$$

Resposta: 72 Kibits.

**c)**Resposta: O endereço seria mapeado no rótulo  $0B7_{16}$  e no conjunto  $0FD_{16}$ .**d)**

A desvantagem é a tradução necessária de endereços virtuais em endereços físicos. Normalmente, um endereço virtual requisitado pelo processador passa pela TLB que faz a tradução para endereço físico e envia à memória cache.

*Comentário:* Uma vantagem de uma cache que opera com endereços físicos sobre uma que opera com endereços virtuais é que ela não gera *aliasing*. Por exemplo, uma troca de página pode fazer com que a cache fique com um endereço virtual correspondente a um quadro na memória e, depois da troca, ganhe um outro endereço virtual que aponte para o mesmo quadro na memória.

---

---

**Questão 3**

Solução:

---

a)

Penalidade de escrita é o atraso ocasionado pelo cálculo e escrita da paridade no sistema de discos.

b)

O esquema RMW não diminuíra o atraso na escrita, num sistema RAID 4, devido o “gargalo” ser o disco de paridade. Independentemente se for apenas um bloco, ou vários, o cálculo da paridade depende do disco de paridade. Portanto, como em ambos esquemas RW e RMW, há centralização da escrita de paridade num único disco, não existe maneiras de haver paralelismo na escrita de blocos, consequentemente, não haverá melhorias modificando a forma que é calculada a paridade.

c)

Num sistema de discos, quando uma falha ocorre, ela é detectada pelo controlador de discos, que sabe exatamente onde ela ocorreu. Desse modo, não é necessário localizá-la apenas corrigi-la, tarefa que a paridade é capaz de fazer. Já na memória principal, não existe essa possibilidade, portanto, se faz necessário o uso de um sistema que detecte e corrija erros, exatamente o que código de Hamming é capaz.

---

---

**Questão 4**

Solução:

---

a)

O multiplicador de Booth executa em média menos operações de soma e subtração que o que multiplicador comum, pois há iterações do algoritmo que só ocorrem deslocamentos, sem qualquer operação de soma ou subtração.

b)

Multiplicando	Resultado/Multiplicador	Q	Etapas
0110	0000 110 <b>1</b>	<b>0</b>	Início
"	1010 110 <b>1</b>	<b>0</b>	Subtraindo
"	1101 011 <b>0</b>	<b>1</b>	Deslocando
"	0011 011 <b>0</b>	<b>1</b>	Somando
"	0001 101 <b>1</b>	<b>0</b>	Deslocando
"	1011 101 <b>1</b>	<b>0</b>	Subtraindo
"	1101 110 <b>1</b>	<b>1</b>	Deslocando
"	1110 111 <b>0</b>	<b>1</b>	Final

---

**1110 1110 = -18**

---