

Projeto Exame
Trabalho Individual
CES- 25 Arquiteturas para Alto Desempenho - 2016
Prof. Paulo André Castro
www.comp.ita.br/~pauloac/ces25/

1. Objetivo

Exercitar e fixar conhecimentos adquiridos sobre sistemas de hierarquia de memória durante o curso de Arquiteturas para Alto Desempenho no 1º. Bimestre

2. Descrição do Trabalho

Implementar um simulador de sistema de memória com dois níveis de cache similares ao do Core 2 Duo T7200 para determinar as taxas de acerto utilizando um programa de benchmark (gcc). L1 é o nome da cache preferencial da CPU, caso não encontre em L1, busca-se na cache L2 e depois na memória principal. Considere que a busca nas caches é feita em paralelo. Implemente a aproximação de LRU através de fila (FIFO).

Tabela 1. Hierarquia de Cache on-chip do Core 2 Duo T7200

Tamanho da Cache L1	32KB Dados (+32KB de Instruções)
Tamanho do bloco (ou linha) em L1	64 bytes
Política de Gravação L1	Write Through
Política de Substituição L1	FIFO
Associatividade L1	Associativo de 8 vias (conjuntos com 8 blocos)
Tempo de Acesso de 1 Palavra em L1	2(clocks)
Tempo de Comparação de 1 Tag	1 (clock)
Outras características L1	1. Write Allocate; 2. considere o tamanho <i>declarado</i> disponível para dados, porém explicito no relatório o espaço necessário para as informações de controle e o tamanho total de cada nível
Tamanho da Cache L2	4096KB
Tamanho do bloco em L2	64 bytes
Política de Gravação L2	Write Back
Política de Substituição L2	FIFO
Associatividade L2	Associativo de 16 vias
Tempo de Acesso (clocks) L2	4 (clocks)
Tempo de Comparação de 1 Tag	2 (clocks)
Outras características L2	1. Write Not Allocate, 2. considere o tamanho <i>declarado</i> disponível para dados, porém explicito no relatório o espaço necessário para as informações de controle e o tamanho total de cada nível.
Tempo de Ac. (clocks) Memória	60

Tabela 1. Dados sobre a Hierarquia de Memória. Considere tempos iguais para acessos de leitura ou escrita e sem falhas de memória principal.

3. Base de Dados

Esta disponível no site da matéria (arquivo gcc.zip), uma base de dados com endereços de **acesso a dados** gerado por um simulador com base na execução de um programa de benchmark conhecido como gcc. O formato do arquivo é dado por um endereço hexadecimal de 32 bits e um indicador se é um acesso de leitura (R) ou escrita (W). Por exemplo:

```
0041f7a0 R
004758a0 R
31348900 W
004a30e0 R
```

Especifique os tempos necessários a realização de cada operação de acordo com a situação, veja as tabelas abaixo. Sempre que possível deve-se considerar operações em paralelo, porém respeitando impossibilidades por uso simultâneo de hardware ou uso de informações ainda não disponíveis. As tabelas abaixo devidamente preenchidas com suas decisões sobre o tempo necessário para cada situação devem constar do relatório do projeto. Pode haver situações indicadas que não ocorrem em um determinado sistema de memória, neste caso apenas marque que a situação não ocorre. Caso encontre alguma situação relevante para a contabilização não indicada nas tabelas abaixo, acrescente-a e indique o tempo gasto nesta situação.

Tabela 2. Operações causadas por acesso ao Sistema de Memória

Operação causada por acesso ao Sist.Mem.	Situação		
	Acerto em L1	Acerto em L2	Acerto na Memória
Leitura	1+2=3		
Leitura	Não(*A)	2+4=6	
Leitura	Não(*B)	Não(*C)	60
Escrita			
Escrita			
Escrita			

Tabela 3. Operações causadas por subida de de bloco

Operação causada por subida de bloco (*ID)	Situação	
	Descrição	Tempo Adicional
(*A)	Atualização apenas em L1; L2 entrega dados a CPU e a L1	0
(*B)		
(*C)		

(*ID) Subida de bloco determinada por situação indicada na Tabela 2.

4. Resultados Esperados e Alteração Proposta

Determine a taxa de acerto das caches L1 e L2 para a base de dados de teste com a configuração dada e o tempo de memória no acesso aos dados em clocks.

Sugira uma mudança no sistema (exceto aumento de tamanho da cache L1), por exemplo, alteração na associatividade, política de gravação ou política de substituição que julgue possa melhorar as taxas de acerto. Não há necessidade de

obter uma melhoria de desempenho, mas o intuito deve ser este. Caso sugira um aumento de L2 considere que $x\%$ de aumento do tamanho, causa um aumento de $x\%$ no tempo de acesso de L2. (Geralmente, caches maiores tem tempos maiores de acesso). Simule o sistema com sua alteração e determine a taxa de acerto das caches L1 e L2 e o tempo gasto com memória usando a mesma base de dados

5. Análise Crítica

Discuta os resultados obtidos por sua alteração e os resultados obtidos com a configuração original.

6. Material a ser Entregue e Prazo

Relatório do Projeto (formato .doc ou .pdf) com:

Objetivo: objetivo do trabalho, citar a linguagem utilizada para implementar o simulador.

Descrição: detalhamento das premissas adotadas (tempo em caso de acerto/falha em cada nível, em caso de substituição, etc.), da implementação do LRU e a da alteração sugerida para o sistema. Explicitar gasto de memória com informações de controle nas caches. Incluir as Tabelas 2 e 3 desta especificação no relatório com os dados preenchidos.

Resultados Obtidos: resultados obtidos com a simulação nas duas configurações.

Análise dos Resultados: discussão dos resultados (ganho, perda ou manutenção) do desempenho em termos de taxas de acerto e tempo de memória e a confirmação ou refutação da sua expectativa em relação a sua proposta de alteração.

Conclusões: Comentários sobre o trabalho (complexidade/facilidade, sugestões, etc.) e contribuições do trabalho para o entendimento sobre sistemas de memória.

Código-fonte e Executável do Simulador (em C, C++, Java ou Python).

Prazo de Entrega: 30 de junho de 2016
(Via e-mail em arquivo .zip).

Bom Trabalho!
Prof. Paulo André Castro
pauloac@ita.br