



CURSO: TADS

SEMESTRE: 1º SEMESTRE TURMA:1B

COMPONENTE CURRICULAR / TEMA: APLICANDO CONHECIMENTO - AULA 6

COMPONENTES DO GRUPO DE TRABALHO:

NOME:ROBERTO ALEXANDRE DE OLIVEIRA TIA: 21510768. POLO: HIGIENÓPOLIS

NOME:ANDRÉ DE SOUZA OCLECIANO. TIA: 21505071. POLO: SAÚDE

NOME: GEOVANNA NUNES DE MATOS SOUZA. TIA: 21511624 POLO: OSASCO

NOME DO PROFESSOR: EDUARDO FERREIRA

(1) Qual é a finalidade da memória cache?

É uma memória intermediária entre o processador e a MP com elevada velocidade de transferência.

Ele é o intermediário entre o processador e a memória principal devido à diferença de velocidade, evitando que o processador fique ocioso.

Auxilia no comportamento dos programas - princípios da localidade

(2) Como é o mapeamento dos blocos da MP para a cache?

Existem três funções de mapeamento:

- Mapeamento direto
- Mapeamento associativo
- Mapeamento associativo por conjunto



(3) Como é feita a atualização da cache? Qual é a vantagem e a desvantagem de cada abordagem?

WRITE THROUGH (ou escrita direta ou escrita em ambas)

Todas as operações de escrita são feitas tanto na memória cache quanto na MP. Dessa forma, consegue-se assegurar que a MP estará sempre com dados válidos.

Desvantagem: considerável tráfego de informação com a MP é gerado, podendo criar um gargalo no sistema, a MP é considerada uma memória lenta. Pode haver uma grande quantidade de escritas desnecessárias na MP e, conseqüentemente, uma redução de desempenho no sistema.

WRITE BACK (ou escrita de volta ou escrita somente no retorno)

As escritas do processador acontecem apenas na memória cache. Quando é feita uma atualização do processador, em uma linha da cache, é atribuído o valor 1 ao bit de atualização (um dos bits de controle na linha da cache) associado à linha atualizada na cache. Quando um bloco vai ser substituído na cache, ele apenas é escrito de volta na MP se seu bit de atualização tiver valor 1.

Desta forma, minimiza-se o número de operações de escrita na MP (que era um gargalo na política write through).

Desvantagem: partes da MP podem ficar inválidas por um tempo e, portanto, o acesso à MP pelos dispositivos de E/S só poderá ser efetuado por meio da memória cache. O acesso direto à cache requer um conjunto de circuitos mais complexo e custoso.

(4) Qual é a finalidade dos algoritmos de substituição? Descreva cada um deles.

Quando um dos blocos da MP precisa ser armazenado na memória cache e ela está cheia, o algoritmo de substituição define a linha da memória cache que deve ser substituída.

LRU (Least Recently Used)

A linha a ser substituída é a que está há mais tempo sem ser referenciada na memória cache

LFU (Least Frequently Used)

A linha a ser substituída é a que foi referenciada menos vezes pelo processador

FIFO (First In First Out)

A linha a ser substituída é a que está há mais tempo na cache, independente de estar sendo usada ou não com frequência pelo

Processador

ALEATÓRIO

A substituição é realizada aleatoriamente e não é baseada em nenhum histórico de uso.



(5) Diferencie cache hit de cache miss.

Cache Hit é quando o conteúdo solicitado pela CPU é encontrado no Cache. Nesse caso a busca pelo conteúdo não consome memória devido o conteúdo já estar no cache, tendo uma resposta mais rápida.

Cache Miss é quando o conteúdo solicitado pelo servidor não está no Cache. Nesse caso a CPU vai tentar buscar a resposta em outros níveis de cache e caso não encontre, diretamente na memória da aplicação. Tendo uma resposta mais lenta e consumindo processamento da aplicação.

(6) Código Hamming

(a) Para a palavra a seguir, calcule a palavra final a ser encaminhada utilizando:

01010111

$$R1 = D1 \wedge D2 \wedge D4 \wedge D5 \wedge D7 = 1 \wedge 1 \wedge 0 \wedge 1 \wedge 1 = 0$$

$$R2 = D1 \wedge D3 \wedge D4 \wedge D6 \wedge D7 = 1 \wedge 1 \wedge 0 \wedge 0 \wedge 1 = 1$$

$$R3 = D2 \wedge D3 \wedge D4 \wedge D8 = 1 \wedge 1 \wedge 0 \wedge 0 = 0$$

$$R4 = D5 \wedge D6 \wedge D7 \wedge D8 = 1 \wedge 0 \wedge 1 \wedge 0 = 0$$

Palavra final: 010100110110.

(b) Considerando o item (a), caso o D4 do código fosse alterado em um armazenamento, mostre a veracidade do código Hamming para descobrir o erro.

$$R1 = 1 \wedge 1 \wedge 1 \wedge 1 \wedge 1 = 1$$

$$R2 = 1 \wedge 1 \wedge 1 \wedge 0 \wedge 1 = 0$$

$$R3 = 1 \wedge 1 \wedge 1 \wedge 0 = 1$$

$$R4 = 1 \wedge 0 \wedge 1 \wedge 0 = 0$$

R4 R3 R2 R1

0 0 1 0

0 1 0 1

0 1 1 1 = posição 7 => D7