



ORGANIZAÇÃO DE COMPUTADORES

LEIC

Conjunto de Exercícios V

Caches - Introdução

Versão 1.0

2022/2023

Exercício 1

1.1. For a direct-mapped cache design with a 32-bit address, the following bits of the address are used to access the cache:

Tag	Index	Offset
31-10	9-4	3-0

- (a) What is the cache line size (in 32-bit words)?
- (b) How many lines does the cache have?
- (c) What is the ratio between the total bits required to implement this cache over the number of data bits effectively stored in the cache?

1.2. Starting from power on, the following byte-addressed cache references are recorded:

Address	0x0	0x4	0x10	0x84	0xE8	0xA0	0x400	0x1E	0x8C	0xC1C	0xB4	0x884
---------	-----	-----	------	------	------	------	-------	------	------	-------	------	-------

- (a) How many blocks are replaced?
- (b) What is the hit ratio?

Exercício 2

For this exercise, refer to the following stream of 32-bit memory address references.

Address	3	180	43	2	191	88	190	14	181	44	186	253
---------	---	-----	----	---	-----	----	-----	----	-----	----	-----	-----

- (a) Show the final cache contents for a fully associative cache with two-word blocks and a total size of 8 words. Use LRU replacement. For each reference, identify the index bits, the tag bits, the block offset bits, and if it is a hit or a miss.
- (b) What is the miss rate for this cache using LRU replacement?

Exercício 3

Uma nova versão de um processador de 16 bits (cujos endereços também têm 16 bits) vai incluir uma pequena cache de instruções para melhorar o desempenho do processador. A cache, só para instruções, terá capacidade para 8 palavras, blocos de 2 palavras e mapeamento directo. A memória é endereçável à palavra (de 16 bits). As especificações dos sistemas de memória actual (sem cache) e futuro (com cache) são as seguintes:

	Sistema actual	Sistema futuro
Tempo de acesso a memória (directo)	100 ns	100 ns
Tempo de acesso à cache em <i>hit</i>	-	20 ns
Tempo de acesso à cache em <i>miss</i> (inclui verificação da cache e acesso à memória)	-	120 ns

Avalie o desempenho do processador ao executar um troço de código que gera a sequência de acessos a memória indicada na tabela seguinte (F – *instruction fetch*, R – *read data*, W – *write data*).

Endereço [h]	Tipo
8010	F
1000	R
8011	F
8012	F
8013	F
8014	F
8017	F
8018	F
2000	W
8019	F
801A	F
801B	F
801C	F
801D	F

- (a) Qual é a distribuição de leituras e escritas na memória no processador actual?
- (b) Calcule o tempo médio de acesso à memória do processador actual (sem cache).
- (c) Indique a estrutura dos endereços tal como são interpretados pela cache.
- (d) Calcule o *hit rate* na cache de instruções do processador futuro.
- (e) Calcule o tempo médio de acesso à memória do processador futuro.
- (f) Qual é o *speedup* do novo processador ao executar este programa? (Considere que o desempenho do processador resulta directamente do tempo de resposta do sistema de memória.)
- (g) É possível que, ao executar outro programa (que gere uma sequência de acessos diferente da indicada na tabela) o processador com cache tenha um desempenho inferior ao do processador original? Se sim, em que condições?

References

- [1] David Patterson and John Hennessy. *Computer Organization and Design: The Hardware/Software Interface*. Morgan Kaufmann, 4th edition, 2011.