

### Ficha 1 – Memória *cache*

1. Um CPU tem endereços de 18 bits e uma memória *cache* de mapeamento direto para palavras (32 bits). A memória *cache* tem 16 posições. A política de escrita em memória é *write-through*. O conteúdo inicial da memória *cache* está indicado na tabela (em hexadecimal).

	conteúdo	etiqueta	v
0	12345678	abc	1
1	6548feab	123	1
2	3c1f56fd	678	1
3	afd12498	567	0
4	6198fa34	b7c	1
5	1929aaaa	8d1	0
6	bbabeedd	cd3	1
7	1123aa56	456	1
8	7611432a	001	1
9	ffffeffe	877	1
10	ddedd556	777	0
11	4444cccc	198	1
12	7627abed	fdf	1
13	8768888a	479	1
14	71672912	655	0
15	22256733	111	1

- Indique qual é o espaço de endereçamento de um sistema baseado neste CPU e determine a capacidade máxima de memória.
- Determine o número de bits da etiqueta e do índice, assim como a capacidade de armazenamento da *cache*.
- Indique as alterações da memória *cache* para a seguinte sequência de acessos (L=leitura, E=escrita):

tipo	endereço	valor
L	2df10	–
L	23454	–
L	3f7b0	–
E	048c4	1212abab
E	3f7d0	00001111
E	1dde8	aaaabbbb

2. Assuma que o CPU do problema anterior é usado com uma memória *cache* também semelhante à anterior, mas que usa a política de escrita *write-back*. O conteúdo inicial da memória *cache* está indicado na tabela (em hexadecimal).

	conteúdo	etiqueta	v	d
0	12345678	abc	1	1
1	6548feab	123	0	0
2	3c1f56fd	678	1	0
3	afd12498	567	0	0
4	6198fa34	b7c	1	0
5	1929aaaa	8d1	0	1
6	bbabeedd	cd3	1	1
7	1123aa56	456	1	0
8	7611432a	001	1	1
9	ffffeffe	877	1	1
10	ddedd556	777	0	0
11	4444cccc	198	1	1
12	7627abed	fdf	0	1
13	8768888a	479	1	0
14	71672912	655	0	0
15	22256733	111	1	0

a) Explique a finalidade do campo d.

b) Indique as alterações da memória *cache* para a seguinte sequência de acessos (L=leitura, E=escrita):

tipo	endereço	valor
L	2df10	–
E	2df10	33334444
E	10c64	9999aaaa
L	23454	–
L	3f7b0	–
E	21de4	bbbb7777
E	2afdc	1212abab
E	3f7b0	00001111

3. Uma memória *cache* do tipo *write-through* com 4 blocos de 8 bytes é usada como *D-cache* num processador ARMv8.

O conteúdo da memória *cache* é o seguinte (em hexadecimal):

bloco	conteúdo								etiqueta	v
	7	6	5	4	3	2	1	0		
0	2a	5d	04	aa	78	00	9c	23	2900002	1
1	1b	b2	34	bb	7a	10	9f	a3	0000893	1
2	99	52	36	cc	88	b0	3c	2b	7bcd001	1
3	42	15	90	cd	71	ab	3f	6d	65abfff	0

Assuma que X0 = 0x52000044 e X1=0xf79a0034.

- a) Qual é o comprimento da etiqueta?
- b) Qual é o valor lido de memória por uma operação de 32 bits que use o endereço guardado em X0?
- c) Que alterações da memória *cache* são causadas pela escrita de um valor de 4 bytes da posição especificada por  $[X0, -4]$ ?
- d) Qual é o valor obtido por uma instrução de leitura de 1 byte do endereço da posição de memória especificada pelo conteúdo de X1?

4. Para cada um dos sistemas indicados a seguir considere que o tempo de acesso a memória principal é de 70 ns e que 36% das instruções acedem a dados em memória. O acesso a memória principal tem início após falta de acesso à memória *cache*.

Processador	Tamanho	Taxa de faltas	Tempo de acerto
P1	1 KiB	11,4%	0,62 ns
P2	2 KiB	8,0%	0,66 ns

- a) Assumindo que é o tempo de acerto que determina o período de relógio, determine as frequências de operação dos dois sistemas.
- b) Determine o tempo médio de acesso à memória para os dois casos.
- c) Assumindo um CPI básico de 1, qual é o CPI de cada um dos processadores? Qual é o processador mais rápido?

5. Um CPU (com  $F=1$  GHz) está equipado com memórias *cache* para instruções e para dados, cujas taxas de faltas são, respetivamente, 5 % e 10 %. O tempo de acesso a memória principal é 80 ns (a acrescentar ao tempo de acesso a memória *cache*).

Em média, 40 % das instruções de um programa acedem a dados (i.e., são *load* ou *store*).

- a) Determine a taxa de faltas global da memória *cache* em número de faltas por 1000 instruções.
- b) Suponha que se pretendia equipar o CPU com uma memória *cache* unificada. Determine a máxima taxa de faltas desta alternativa para que ela apresente o mesmo desempenho que a versão *split cache*.
- c) Assuma que, na ausência de faltas de *cache*, o CPU tem  $CPI_{ideal}=1,2$ . Determine o CPI efetivo para os seguintes casos:
  - i. sistema sem memória *cache*;
  - ii. sistema com memória *cache*.

6. A tabela seguinte apresenta o conteúdo (em hexadecimal) de uma memória *cache* do tipo *write-back* com 8 blocos de 8 bytes usada como *D-cache* num CPU com endereços de 16 bits.

bloco	conteúdo								etiqueta	v	d
	7	6	5	4	3	2	1	0			
0	aa	cc	de	hf	34	33	11	01	235	1	0
1	bb	ad	45	4f	af	de	21	99	391	1	1
2	cc	34	ab	1f	56	cd	ff	ff	023	1	1
3	dd	67	22	2b	32	56	32	21	198	0	1
4	ee	32	11	9f	aa	ba	ab	bb	311	1	0
5	ff	10	00	04	01	02	03	04	278	0	0
6	11	03	41	32	cc	dd	ee	ff	212	1	1
7	22	01	65	01	05	06	07	08	387	0	1

- Como é decomposto o endereço para acesso à memória *cache*? Justifique.
- Indique (se possível) o valor (byte) em memória principal no endereço 0xc467. Justifique.
- Explique quais as alterações que ocorrem na *cache* e na memória principal durante a leitura do valor (byte) residente no endereço 0xe48d.

7. Um CPU tem endereços de 16 bits e uma memória *cache* de mapeamento direto para dados (palavras, 32 bits). A memória *cache* é do tipo *write-back*, possui uma palavra por bloco e a respetiva indexação é formada por 4 bits.

A tabela seguinte mostra parcialmente o conteúdo da *cache* (conteúdo e etiqueta em hexadecimal).

	conteúdo	etiqueta	v	d
...	...	...	...	...
5	7BCDBCD7	241	0	0
6	76543210	1F3	1	0
7	FFFFFFFF	241	1	1
8	80000000	240	1	0
9	26E111A4	241	1	1
10	13012020	199	1	0
...	...	...	...	...

- Determine quantos blocos tem a *cache* e quantas palavras em memória podem ser mapeadas em cada entrada da *cache*.
- Determine o conteúdo de memória nos endereços 0x9020 e 0x9024.
- Mostre quantos acessos a memória ocorrem como consequência das seguintes operações:
  - ler Mem[0x7CDC]
  - escrever 0x12345678 em Mem[0xE020]
- Assuma que a taxa de faltas da *cache* é 1/9 da taxa de acertos, a penalidade de falta é 80 ciclos e o CPI devido a protelamento no acesso a dados é 2. Nestas condições, determine a percentagem de instruções executadas que acedem a memória.

Fim