

1. Considere que o PEPE é um processador com um pipeline de 4 estágios (cujos registos suportam *read-after-write*, isto é, os registos são escritos na primeira metade do sinal de relógio e os novos valores podem ser lidos já na segunda metade do mesmo ciclo de relógio). Os estágios são os seguintes:

- B (Busca de Instrução)
- D (Descodificação e Busca de Operandos)
- E (Execução da Instrução)
- W (Escrita do Resultado).

	T0	T1	T2	T3	T4	T5	T6	T7	T8
SUB R1, 04H	B	D	E	W					
JZ K		B	D	E					
AND R2, R1			B	D					
MOV R1, R2				B					
K: OR R1, R2					B	D	E	W	
ADD R2, R1						B	D	E	W

2. Suponha que a *cache* do PEPE (processador com 16 bits de endereço, endereçamento de byte) é de mapeamento directo, com uma capacidade de 128 palavras (blocos de 8 palavras).

- a) Indique o número de bits de cada um dos campos em que o endereço se divide para acesso à *cache*.

Etiqueta	8
Índice	4
Palavra dentro do bloco	3
Byte dentro da palavra	1

- b) Na execução de instruções do tipo MOV R1, [R2], o núcleo do PEPE verificou que nuns casos o valor de R1 demorava 3 ns a obter, noutros 20 ns, e que em média demorava 4,7 ns. Qual a taxa de insucesso (miss rate) da *cache* com este programa?

10 %

4. Imagine um processador com endereçamento de byte, capaz de endereçar um espaço virtual de 00000H até FFFFFH, enquanto o espaço de endereçamento físico vai de 0000H até FFFFH. As páginas virtuais têm uma dimensão de 100H bytes. A TLB é totalmente associativa de 8 entradas.

Posição da TLB	Bit de validade	N.º página virtual (hexadecimal)	N.º página física (hexadecimal)
0	1	C5E	18
1	1	3B	2B
2	0	C5E	6F
3	1	213	0E
4	0	6A	23
5	0	A02	2B
6	1	23	3B
7	0	45	2B

- a) Suponha que o processador executa estas instruções, assumindo-se que inicialmente R1=04H e R2=2FH. Assuma ainda que as instruções de salto alteram o PC (em caso de salto) na execução do estágio E. Cada um dos tempos T0 a T8 tem a duração de um ciclo de relógio. Indique (com a letra respectiva) em que tempo é executado cada estágio de cada instrução.

- b) Indique que conflitos de dados são visíveis neste programa, traçando um círculo à volta do registo que tem uma dependência (em conflito) de uma instrução anterior. Isto tem de ser válido em qualquer caso e não apenas no caso a).
- c) Suponha que só pode resolver os conflitos em software. Escreva uma nova versão do programa, sem conflitos (mas mantendo a funcionalidade).

SUB R1, 04H
JZ K
AND R2, R1
NOP
MOV R1, R2
NOP
K: OR R1, R2
NOP
ADD R2, R1

3. Suponha agora que o PEPE-8 (processador de 8 bits de dados, com 8 bits de endereço) tem uma *cache* de mapeamento directo com 8 entradas e está inicialmente vazia, sendo feitos os acessos em leitura aos endereços indicados na tabela e pela ordem com que aparecem. Para cada acesso, indique a posição (0 a 7) da *cache* que é usada, se foi *hit* (sim ou não) e qual o valor da etiqueta com que essa posição fica após esse acesso.

Endereço acedido	Posição da <i>cache</i>	Hit (S ou N)	Etiqueta (em binário)
3AH	2	N	00111
54H	4	N	01010
62H	2	N	01100
54H	4	S	01010
3AH	2	N	00111
54H	4	S	01010

Após alguns acessos, a TLB ficou como indicada. Preencha a tabela seguinte para este exemplo concreto.

Dimensão do espaço físico (em Bytes)	64K
Dimensão do espaço virtual (em Bytes)	1 M
N.º páginas do espaço virtual	4K
N.º <u>actual</u> (neste instante) de páginas virtuais cuja tradução virtual/físico pode ser feita sem aceder à tabela de páginas	4
Endereço físico que corresponde ao endereço virtual 3B18H	2B18H
Endereço virtual que corresponde ao endereço físico 182BH	C5E2BH