

## Parte Teórica

**Cotações:**

1- 0,5; 2- 0,5; 3- 2,5; 4- 2,5; 5a- 1,5 b- 0,5; 6a- 0,5 b- 1,5 c- 2,5; 7a- 0,5 b- 2 c- 2; 8a- 2, 8b- 1

**Nota: 1 e 2 descontados 0,5 por cada resposta errada (resposta no enunciado)**

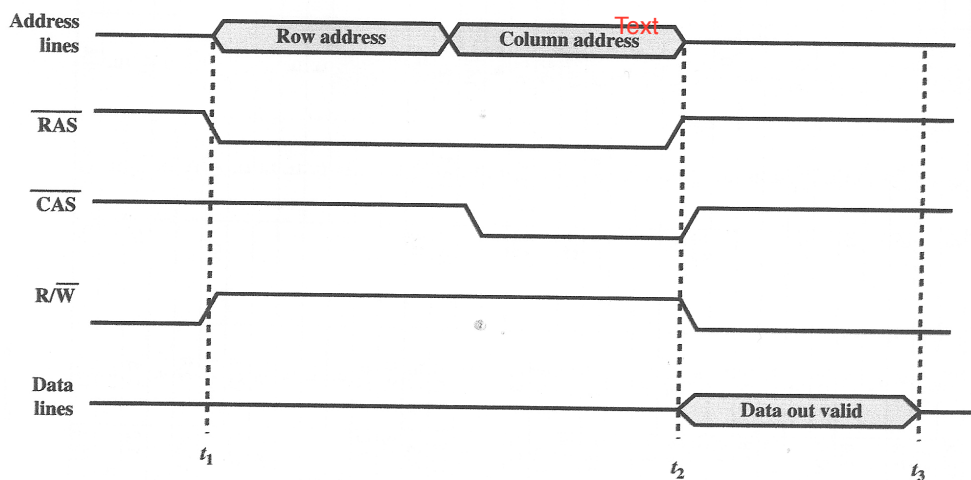
1. O I2C é um bus:
  - a. Síncrono
  - b. Assíncrono
2. Qual o tipo de memória usado nos SSD:
  - a. SRAM
  - b. Flash Memory**
  - c. EPROM
3. Como funciona um sistema de interrupções vetorizadas? Quais as diferenças relativamente a interrupções não-vetorizadas? Descreva o sistema de interrupções do PIC32.
4. *Carrier Sense Multiple Access* é usado por Ethernet e por CAN. Explique em que consiste. Porque é que CAN usa uma estratégia diferente da usada por Ethernet na resolução de conflitos no acesso ao bus?

## Parte Teórica

5. A figura representa um diagrama temporal simplificado de um ciclo de leitura de uma DRAM. O tempo de acesso é o intervalo  $[t_1, t_2]$ . O tempo de restabelecimento é o intervalo  $[t_2, t_3]$ , durante o qual a DRAM tem de restaurar a carga antes de o processador possa efetuar um novo acesso. Assuma que:  $t_2 - t_1 = 25\text{ns}$  e  $t_3 - t_2 = 15\text{ns}$

a. Qual é o tempo de ciclo? Qual a máxima taxa de transferência que esta DRAM pode assegurar assumindo que se trata de uma 64Mx4?

b. Se se construir uma memória x32 usando esta componente qual a taxa de transferência?



6. Pretende-se projetar um sistema de memória que permita detetar erros nos bytes armazenados.
- Se apenas se pretender detetar erros de um bit, que solução adotaria?
  - Se se pretender detetar erros e corrigir os erros de 1 bit, que código de deteção e correção de erros adotaria? O código de quantos bits adicionais para deteção e correção de erros seriam necessários por cada byte?
  - Para o byte 01101101 gere os bits de deteção e correção de erros. Mostre que o código identifica corretamente um erro no bit 4.

## Parte Teórica

7. Um sistema de 32-bits com a memória byte-addressable tem uma cache direct-mapped de 4kBytes com blocos (linhas) de 4 palavras de 32 bits.
- Quantas linhas tem a cache? **Linha: 4 palavras -> 16 bytes; N° linhas =  $4 \cdot 2^{10} / 16 = 2^8 = 256$**
  - Quais os campos em que estão divididos os endereços da memória e qual o número de bits de cada um deles?

- O conteúdo da posição de memória 101B2A5C foi transferido para a cache. Em que linha da cache e em que posição na linha se encontra? Quais os endereços das outras palavras armazenadas na mesma linha da cache? Qual o conteúdo do campo **Tag** correspondente a essa linha?

**Linha da cache =  $A5_{16} = 165$       Posição da palavra na linha:  $11_2 = 3$       Posição do byte:  $C_{16} = 12$**   
**Endereços das outras palavras armazenadas na mesma linha: 101B2A50, 101B2A54, 101B2A58**  
**Tag = 101B2**

8. Uma unidade de disco tem as seguintes características: *Seek time* = 10ms; Velocidade de rotação = 5400rpm; *Transfer Rate* = 8 MB/s.
- Calcule o tempo médio requerido para transferir 32kB do disco para a memória, assumindo que esses dados se encontram armazenados em sectores contíguos.

$$T_{av} = t_{seek} + t_{rot}/2 + t_{data} = 10 + 0,5 \cdot (60/5400) \cdot 10^3 + (32 \cdot 2^{10} / 8 \cdot 2^{20}) \cdot 10^3 \text{ ms}$$
$$T_{av} \approx 10 + 5,56 + 0,03 = 15,59 \text{ ms}$$

- As transferências entre o disco e a memória fazem-se por DMA. Porquê?

**As transferências de informação de/para o disco envolvem não itens individuais (bytes ou palavras) mas blocos de dados (no mínimo o conteúdo de um sector do disco, cuja dimensão nas atuais unidades de disco é no mínimo de 512 Bytes e pode ir até 8kB), com velocidades de transferência**

## Parte Teórica

elevadas (8 MB/s no caso da unidade de disco aqui referida). Se o processador tivesse de efetuar essas transferências:

disco -> registo do processador -> memória na leitura

memória -> registo do processador -> disco na escrita

o tempo de processador gasto nas comunicações com o disco seria considerável. Para libertar o processador da tarefa da transferência de dados, o sistema de I/O dos computadores inclui um módulo (o controlador de DMA) com capacidade de endereçar a memória e de fazer a transferência de dados diretamente entre memória e os periféricos em que a transferência de dados se faz em blocos (*block devices*) e não carater a carater (*character devices*).

O controlador de DMA atua assim como um processador auxiliar controlado pelo processador central que lhe indica o endereço base do bloco de memória para/de onde transferir os dados e a dimensão do bloco de dados. Quando termina a transferência o controlador de DMA gera uma interrupção para informar o processador de que terminou a tarefa.

No caso do PIC32 como os módulos de I/O estão integrados no mesmo chip que o processador, os módulos que comunicam com *block devices* (Ethernet, CAN, High Speed USB,...) incorporam controlador de DMA próprio, existindo para além disso um controlador de DMA autónomo que permite assegurar a capacidade de acesso direto à memória de outros dispositivos externos.