

### Parte Teórica (sem consulta)

- (2) 1. Descreva *sucintamente* as principais diferenças entre a arquitectura Intel x86 e a arquitectura MIPS. Baseado na sua experiência de programação em linguagem *assembly* nessas duas arquitecturas faça uma avaliação das respectivas vantagens e inconvenientes.

**Notas:**

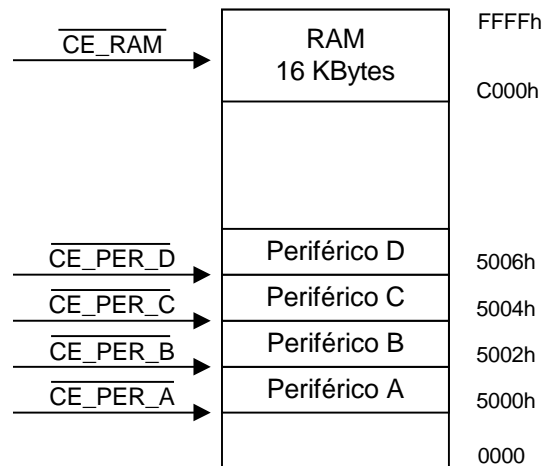
- Na sua resposta não deixe de abordar os seguintes pontos:
    - Modos de endereçamento
    - Instruções aritméticas e lógicas
    - Passagem de parâmetros para subrotinas
  - *Não se pretende* que a resposta seja uma listagem das instruções das duas arquitecturas a comparar
- (2) 2. Um dado computador tem 256 Mbyte de memória, utilizando módulos de 64 MByte construídos com pastilhas de DRAM de 64MX1 bits. Desenhe descodificadores de endereços para esses módulos supondo que eles estão:
- “interleaved”
  - “non-interleaved”
- Indique em ambos os casos quais os endereços que correspondem a cada um dos módulos
- (3) 3. a. Explique a diferença entre entradas/saídas programadas e entradas/saídas sob interrupção
- b. Explique o processamento de interrupções em sistemas Intel. Porque se dizem as interrupções vectorizadas?
- (3) 4. Considere uma operação de escrita em I/O pelo 80188 vista nos barramentos do próprio microprocessador, ou seja, sem se tomar em consideração a “latch” da parte menos significativa do barramento de endereços.
- a) Trace o diagrama temporal da operação, indicando os sinais A15-8 (*Addresses*, parte mais significativa), AD7-0, ALE (*Address latch enable*) e WR (*Write*). Justifique.
- b) Caracterize o protocolo do ponto de vista da sincronização quer na transferência de endereços quer na escrita de dados. Justifique.
- c) Caracterize o protocolo do ponto de vista da coordenação entre o endereçamento e transferência de dados. Justifique.
- d) Considere agora o protocolo visto após a “latch” dos endereços menos significativos. Responda à alínea c) nesta nova perspectiva.
- (2,5) 5. Os elementos ligados ao barramento PCI podem ser designados por *Initiators* ou *Targets*.
- a) Qual o papel de cada um desses elementos?
- b) A que se destinam os sinais de IRDY# e TRDY#? Quando é que podem ser utilizados e quais são as consequências? Explique com auxílio de diagrama temporal.
- c) Como é que no barramento PCI são sinalizadas as operações de transferência de dados (Leitura, Escrita, ...)? Que sinais participam nessa sinalização?
- d) Qual é a máxima taxa de transferência em Bytes/seg suportada pelo barramento PCI (se não se lembrar dos valores exactos relevantes arbitre com razoabilidade)? Justifique.
- (1) 6. Na folha de características de um disco duro actual pode-se observar a indicação dos seguintes parâmetros de desempenho:

*Average Latency:* 4,16 msec    *Spindle Speed:* 7.200 RPM

Indique qual o significado dos mesmos e demonstre a relação entre ambos.

### Parte Prática

Um sistema baseado num  $\mu\text{C}$  80188 apresenta o espaço de endereçamento de periféricos ocupado do modo que a figura seguinte ilustra:



#### 1.

a) Admita que o  $\mu\text{C}80188$  disponibiliza o sinal "M/IO" que permite diferenciar os acessos aos espaços de endereçamento de memória e de periféricos ( $\text{M/IO} = 0 \rightarrow$  acesso ao espaço de endereçamento de periféricos). Apresente a lógica de descodificação para a memória RAM.

b) Utilizando como base a linha de selecção  $\text{PCS2}$ , apresente a lógica de descodificação mínima para cada um dos quatro periféricos (i.e., geração dos sinais  $\text{CE\_PER\_A}$ ,  $\text{CE\_PER\_B}$ ,  $\text{CE\_PER\_C}$  e  $\text{CE\_PER\_D}$ )

c) Construa uma rotina em Assembly para a programação da linha  $\text{PCS2}$ , supondo a utilização de 2 *wait states*. Justifique os valores apresentados.

2. Os periféricos A e B são portos de entrada, enquanto que os periféricos C e D são portos de saída. O periférico A tem a capacidade de gerar uma interrupção sempre que o dispositivo externo a que está ligado aí coloca alguma informação. Suponha que está a ser utilizada, para esse efeito, a linha de interrupção INT0.

a) Construa um procedimento em linguagem C que programe:

- o vector de interrupção (admita que a rotina de serviço se designa "RSI\_INT0");
- o controlador de interrupções, por forma a que a linha INT0 seja sensível ao nível, não esteja mascarada e tenha a prioridade mínima.

b) Implemente em linguagem C o procedimento de serviço à interrupção (RSI\_INT0), de modo a que leia um valor do porto de entrada (periférico A) e, se se tratar de um carácter alfabético (códigos ASCII entre 41h e 5Ah e entre 61h e 7Ah), o coloque no porto de saída (periférico C).

3. De modo a verificar o respectivo estado, pretende-se fazer *polling* sobre o periférico B a intervalos regulares. Apresente um procedimento em linguagem C que faça a programação dos *timers* do  $\mu\text{C}80188$ , por forma a que o *timer* 1 gere uma interrupção de 5 em 5 segundos.

(Sugestão: utilize o *timer* 2 como pré-divisor do *timer* 1).

**Cotação:** 1a-1.5, 1b-1.5, 1c-0.5, 2a-1, 2b-1, 3-1