

Sistemas Embebidos I

Semestre de Inverno de 2018/2019

1º teste - 16 de janeiro de 2019

Grupo I

Na sua folha de teste, indique a opção que considera correta como resposta a cada uma das seguintes questões de escolha múltipla. Note que a indicação de uma opção incorreta **desconta 50%** da cotação atribuída à questão em causa.

1. [1] A arquitetura ARMv7M presente no microcontrolador Cortex-M3 suporta o funcionamento ...

- A – ... exclusivamente em modo ARM.
- B – ... exclusivamente em modo Thumb.
- C – ... nos dois modos, Thumb ou ARM.

2. [1] O conjunto das instruções Thumb2 caracteriza-se por conter instruções codificadas...

- A – ... a 16 bit
- B – ... a 32 bit
- C – ... de forma variável (16 bit ou 32 bit)

3. [1] Para colocar no microcontrolador LPC1769 os pinos P1.0 e P1.1 a zero, utilizando as definições do CMSIS, deve ser feito:

- A – `LPC_GPIO1->FIOCLR &= ~3;`
- B – `LPC_GPIO1->FIOCLR |= 3;`
- C – `LPC_GPIO1->FIOCLR = 3;`

4. [1] Para que um programa autónomo para microcontrolador LPC1769 se execute corretamente o conteúdo da posição de memória 0x00000004 deve conter um endereço entre...

- A – 0x00000000 e 0x0007FFFC, inclusive.
- B – 0x10000000 e 0x10007FFC, inclusive.
- C – 0x00000000 e 0xFFFFF7FC, inclusive.

Grupo II

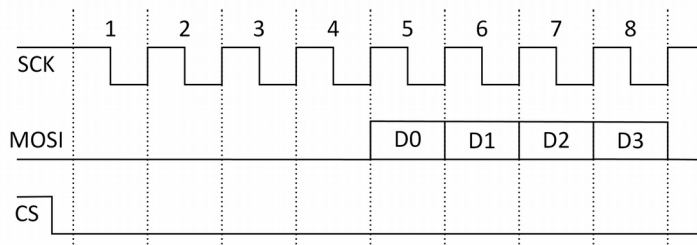
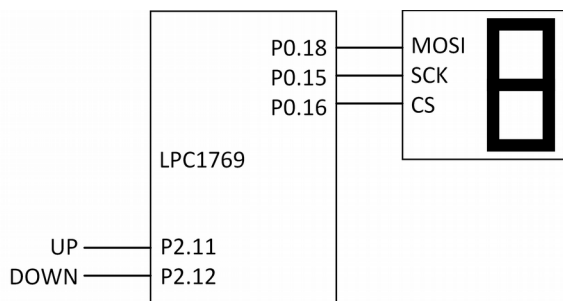
5. [2] Relativamente ao periférico System Tick da arquitetura Cortex-M3, justifique a seguinte afirmação: “Sendo o valor de PCLK igual a 100000000, não é possível ter interrupções periódicas com período de 1 seg.”

6. [2] Apresente, de forma sucinta, as razões que, no microcontrolador LPC1769, levam à execução do código do *bootloader*.

7. [3] Relativamente ao barramento SPI, comente, justificando, a seguinte afirmação: “Só por protocolo (*software*) é que é possível detetar se um *slave* não está presente no barramento.”

Grupo III

8. [9] A figura em baixo ilustra um sistema embebido baseado no microcontrolador LPC1769 que afixa num mostrador de 7 segmentos, com interface para barramento serie síncrono, o valor corrente do estado de um contador de 4 bits. As entradas UP e DOWN permitem, respetivamente, incrementar e decrementar o valor a afixar no mostrador entre os limites 1 e F. O valor da contagem é atualizado sempre que uma das entradas apresenta o valor lógico 1 isoladamente e reiniciado com o valor 0 sempre que as duas entradas apresentam o valor lógico 1 em simultâneo.



- [1] Indique, justificando, qual a programação do pino P0.16.
- [2] Programe a função `void SEG7_Init()`, responsável pela iniciação do mostrador de 7 segmentos como indicado no diagrama temporal. Admita a existência da função `SPI_Init(int bitsData, int CPOL, int CPHA)`, que inicia o controlador SPI para comunicar com uma palavra de dimensão `bitsData` e configura os valores de `CPOL` e `CPHA`.
- [2] Programe a função `unsigned int IN_Read()`, responsável pela leitura do estado corrente das entradas. Esta função não é bloqueante e devolve um de quatro valores possíveis: `NONE`, `UP`, `DOWN` ou `BOTH`.
- [2] Desenhe um diagrama de estados representativo do funcionamento do sistema, explicitando os objetos intervenientes e os eventos que o fazem evoluir.
- [2] Apresente o programa que implementa a solução descrita na alínea anterior.

Considere a existência da função `SEG7_Write(unsigned int value)`, que afixa no mostrador de 7 segmentos o valor de `value`.

Nota: Admita a existência das definições CMSIS para o microcontrolador LPC1769.

Pedro Sampaio