

# Sistemas Digitais I

## 2ª Repetição Escrita de 2002/03

Leia atentamente o enunciado. Seja breve nas respostas, mas justifique-as convenientemente. **Por favor**, use letra legível ! Com os melhores desejos de um feliz Natal e bom trabalho nesta repetição ...

### I

Lembrar-se-á decerto do problema apresentado no grupo II da última repetição, que é transcrita aqui:

Recentemente, alguns dos alunos que frequentam esta cadeira visitaram o N.R.P. Côte Real. Para entrarem no navio tiveram que passar por umas comportas que isolam a "cidadela" do meio exterior. A "cidadela" do navio é constituída pelas zonas interiores no navio que estão a uma pressão superior à pressão ambiente, de modo a garantir que caso haja um ataque NBQ, não haverá ar contaminado a entrar para o navio (o ar comprimido passa por filtros que garantem a descontaminação).

Cada comporta tem uma porta exterior e uma porta interior. Cada uma dessas portas tem um sensor que envia um sinal digital "0" quando a porta está fechada, e um sinal "1" quando a porta está aberta. Embora de facto existam mais, assuma que existem 8 comportas de acesso à cidadela. Para manter a pressão interior superior à exterior, é necessário ter um sensor de pressão externa, outro de pressão interna, e um compressor. Os sensores de pressão (por hipótese) medem pressões entre 0 e 65000 milibares, com uma precisão de +/- 1 milibar, disponibilizando em cada instante o valor da pressão como um número binário de 16 bits. O compressor é comandado através 2 linhas digitais: a primeira liga (quando é 1) ou desliga (quando é 0) o compressor, enquanto a segunda põe o compressor em modo "normal" (quando é 0), ou modo "rápido" (quando é 1).

Por razões que veremos mais tarde, o sistema deverá estar ainda ligado a um termómetro, que indica a temperatura no interior no navio, em graus Celsius, como um número de 8 bits em notação de complemento para 2. Para evitar que a pressão interna suba demasiado, é necessário ter uma válvula de escape, que abre quando recebe um sinal digital "1", e fecha quando recebe um "0".

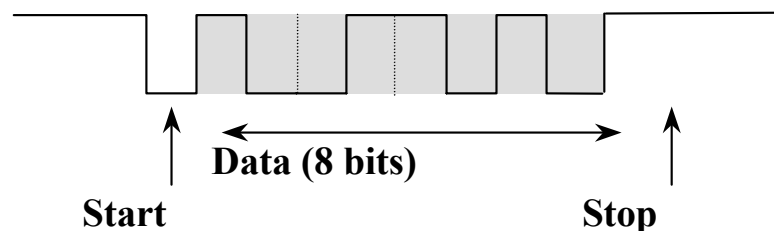
Finalmente, deverá haver um interruptor para ligar e desligar o sistema de compensação de pressão. Neste momento, o controlo das comportas, sensores, e compressor, é feita pelo sistema Nautos. No entanto, seria interessante (por exemplo para os novos NPO 2000) ter um sistema descentralizado, onde este controlo seria efectuado por um dispositivo autónomo que depois trocava informação com os outros sistemas de bordo. A troca de informação com os outros sistemas pode ser feita usando um protocolo série (implementado num sistema 8085 usando os pinos SID e SOD).

Imagine que está a trabalhar na empresa D.N. (Desenvolvimentos Náuticos Ltd), e que lhe é pedido que proponha um protótipo de sistema para controlar as comportas da cidadela e o sistema de pressurização. Genericamente, pretende-se que o sistema mantenha uma pressão interior ligeiramente superior à exterior, pretende-se manter estatísticas sobre a utilização das comportas, e pretende-se detectar situações anómalas. Devido à situação de crise em que o país se encontra, pretende-se desenhar o sistema em torno de um microprocessador barato e simples, e optou-se por usar um 8085, usando um clock (clock out) de 1MHz. Prevê-se que o código do programa não ocupe mais do que 16K, e que os dados necessários não passem os 32K. Por razões várias pretende-se que existe memória RAM disponíveis nos endereços A000 a A0FF. Prevê-se também que haja certas acções que deverão ser tomados de segundo a segundo.

Para ajudar a sincronizar essas acções está disponível um "relógio" digital que gera pulsos curtos com uma frequência de 1Hz.

Por hipótese, alguém do grupo de trabalho já definiu que os endereços de 0000H a 3FFFH são ocupados com ROM, e os endereços 8000H a FFFFH são ocupados com RAM. No entanto, as ligações com os periféricos ainda estão por fazer.

1. Pretende-se fazer a interface com o sensor de pressão, compressor, e válvula de escape usando um único integrado, mais concretamente o 8755 (é a versão que tem EPROM).
  - 1.1 - Apresente um logograma que indique TODAS as ligações entre o microprocessador o 8755 e os dispositivos em causa.
  - 1.2 - Escreva o código máquina que deverá ser incluído nas rotinas de inicialização do sistema para programar o 8755.
  - 1.3 - Escreva uma rotina que lê o valor da pressão, e deixa o resultado no par de registos DE. Essa rotina será chamada LER\_PRESSAO.
  - 1.4 - Escreva uma rotina de "comando" da válvula de escape, de acordo com um byte que lhe é passada no acumulador. Se esse byte fôr 0, a válvula deverá ser fechada, caso contrário a válvula deverá ser aberta. Essa rotina deverá ser chamada VALVULA.
  - 1.5 - Escreva uma rotina de "comando" do compressor, de acordo com um byte de comando que lhe é passada no acumulador. Se esse byte tiver o valor 0, a rotina deverá parar o compressor. Se fôr 1, deverá ligar o compressor na potência mínima, e se fôr 2 deverá ligá-lo na potência máxima. Essa rotina será chamada COMPRESSOR.
  
2. Escreva a rotina que receba um byte de dados através da porta série (SID) com o formato lógico do RS232. Como se lembrará este formato lógico consiste em enviar um start bit a 0, 8 bits de dados, e um stop bit a 1 (como se vê na figura). Admita que está a usar um bit rate de 150, i.e., cada bit demora 1/150 de segundo. A sua rotina deverá estar constantemente a ler o pino SID (que está "normalmente" a 1), à espera que este "caia" para 0. Quando isso acontecer, a rotina deverá esperar 1/300s e ler novamente o pino SID. Isto faz com que o "start bit" seja lido exactamente a meio. Se essa leitura não fôr 0, isso quer dizer que a mudança de 1 para 0 no pino SID não foi o início de uma transmissão mas sim um "glitch". Nesse caso, a rotina deverá terminar, deixando o acumulador com o valor 0. Caso o start bit continue de facto a 0, então a rotina deverá esperar mais 1/150s para ler o 1º bit de dados (que será o bit MENOS SIGNIFICATIVO do dado). A partir daí a rotina deverá ler todos os bits de dados (esperando sempre 1/150s entre leituras consecutivas). Após ter lido o último bit, a rotina deverá terminar, deixando no acumulador o dado lido. Como o relógio de 1Hz não pode gerar os sincronismos necessários, pode usar um outro relógio de 1.8kHz. Para além de TODO o código necessário, indique também que ligações (hardware) é que seria necessário fazer. A rotina de leitura será chamada LER\_RS232



3. Pretende-se usar a rotina desenvolvida na pergunta 2 para permitir que o computador central (o sistema Nautos) envie para o nosso sistema a pressão que pretende estabelecer. O sistema Nautos enviará primeiro o byte mais significativo, e depois o menos significativo. Escreva uma rotina chamada LER\_COMPUTADOR que usando a rotina LER\_RS232 deixe no par DE o valor da pressão pretendida.

4. Um camarada seu que não sabe programar em assembler, escreveu a seguinte rotina em PASCAL (a linguagem que tive como cadete do 1º ano da Escola Naval):

```
PROGRAM KeepPressure
VAR
    Alvo : INTEGER;
    Actual : INTEGER;
BEGIN
    alvo = LER_COMPUTADOR();
    actual = LER_PRESSAO();
    IF alvo > actual THEN
        BEGIN
            WHILE alvo > actual DO
                COMPRESSOR( 2 );
            COMPRESSOR( 0 );
        END
    ELSE
        BEGIN
            WHILE alvo < actual DO
                VALVULA( 1 );
            VALVULA( 0 )
        END;
    END.
```

Assumindo que o tipo de dados INTEGER ocupa 2 bytes, passe para assembler do 8085 a rotina apresentada. Indique claramente onde são guardadas as diversas variáveis em jogo.

Boa sorte !

