

# Lista de Exercícios 1

## Conceitos envolvidos:

- Contadores e Temporizadores
- Interface serial RS232
- Interrupções

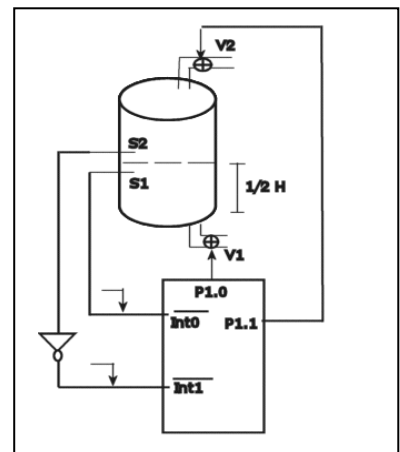
- Fazer um programa que utilize um timer interno do **8051** para criar um tempo de atraso de 0.05 segundos. Utilizando este programa como uma sub-rotina, escrever um programa que atrase 5 segundos.
- Um sistema baseado no 8051 utiliza as duas interrupções externas disponíveis e ainda a interrupção gerada por 1 dos Temporizadores/Contadores.

As condições em que se pretende que o sistema funcione são as seguintes:

- a interrupção externa 0 deve ser sempre atendida imediatamente e deve copiar o que está na posição de RAM externa 4000H para a posição 4200H;
- a interrupção externa 1 deve escrever o que está em 4200H na porta P1;
- a interrupção gerada pelo timer deve executar uma rotina que copie o que está na porta P2 para a posição 4000H da RAM externa;
- No caso de duas interrupções acontecerem simultaneamente, deve ser atendida a interrupção externa.

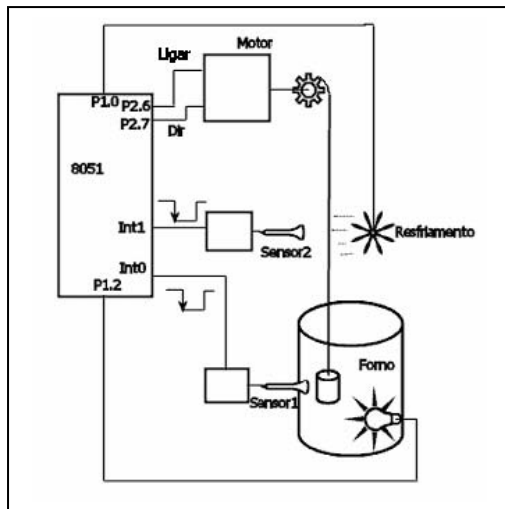
- Considere o Controlador de Nível da figura operando da seguinte maneira:

- Dois sensores S1 e S2 emitem nível lógico zero se estiverem fora do líquido e nível lógico 1 se estiverem imersos no líquido.
- Uma válvula V1, acionada pelo bit P1.0 de um microcontrolador 8051 drena o reservatório e uma válvula V2 acionada pelo bit P1.1 enche-o com líquido.
- Inicialmente o reservatório está vazio, ou seja, com os dois sensores em nível lógico zero.



Escrever um programa em Assembly do 8051 que mantenha o nível do líquido próximo à metade do reservatório (1/2 H) automaticamente, utilizando as entradas de Interrupções assinaladas, observando que as mesmas sentem a mudança de borda (descida) quando o líquido passa pelos sensores (S1 → Drenando e S2 → enchendo)

4. Escrever um programa em Assembly do 8051 que controle o dispositivo de teste



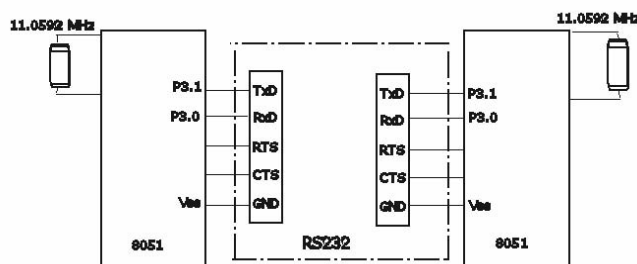
térmico de materiais, mostrado na figura. Um recipiente, com determinada substância sob teste, deve ser baixado ( $Dir = P2.7 = 1$ ) através de um Motor (Ligar =  $P2.6 = 1$ ), dentro de um forno. O Sensor1 detecta a presença do recipiente e envia uma descida de borda ao pino Int0 do microprocessador. O micro para o Motor (Ligar =  $P2.6 = 0$ ) e aciona o aquecimento do forno ( $P1.2 = 1$ ) durante aproximadamente 500 ms. Desliga o aquecimento, inverte o sentido do Motor (Ligar =  $P2.6 = 1$ ) ( $Dir = P2.7 = 0$ ), erguendo o recipiente

até a posição do Sensor2, que opera da mesma forma que o Sensor1, mas usando a Interrupção Int1. Quando Int1 receber uma descida de borda, o micro deve parar o Motor (Ligar =  $P2.6 = 0$ ) e acionar o resfriamento ( $P1.0 = 1$ ) durante aproximadamente 500 ms. O ciclo deve ser repetido 3 vezes e parar. Considerar o Cristal da CPU de 12 MHz.

- Fornecer o programa em Assembly
- Fornecer os parâmetros de programação dos tempos envolvidos.

5. Dois microcontroladores 8051 estão se comunicando através de uma interface padrão RS232 com *handshaking* via RTS e CTS. No esquema abaixo conectar os fios do cabo de comunicação corretamente e responder aos itens:

- Qual o valor de TH1 em ambos os micros se a taxa de comunicação é de 19200 *bauds*?
- Qual o valor de tensão na linha de comunicação quando esta está em repouso?
- Qual o tamanho, em microssegundos, do *Start Bit*?

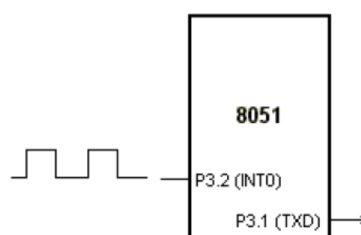


6. Um sistema baseado no Microcontrolador 8051 utiliza as duas interrupções externas e as interrupções geradas pelos Timers/Counters.

Escrever um programa em Assembly tal que:

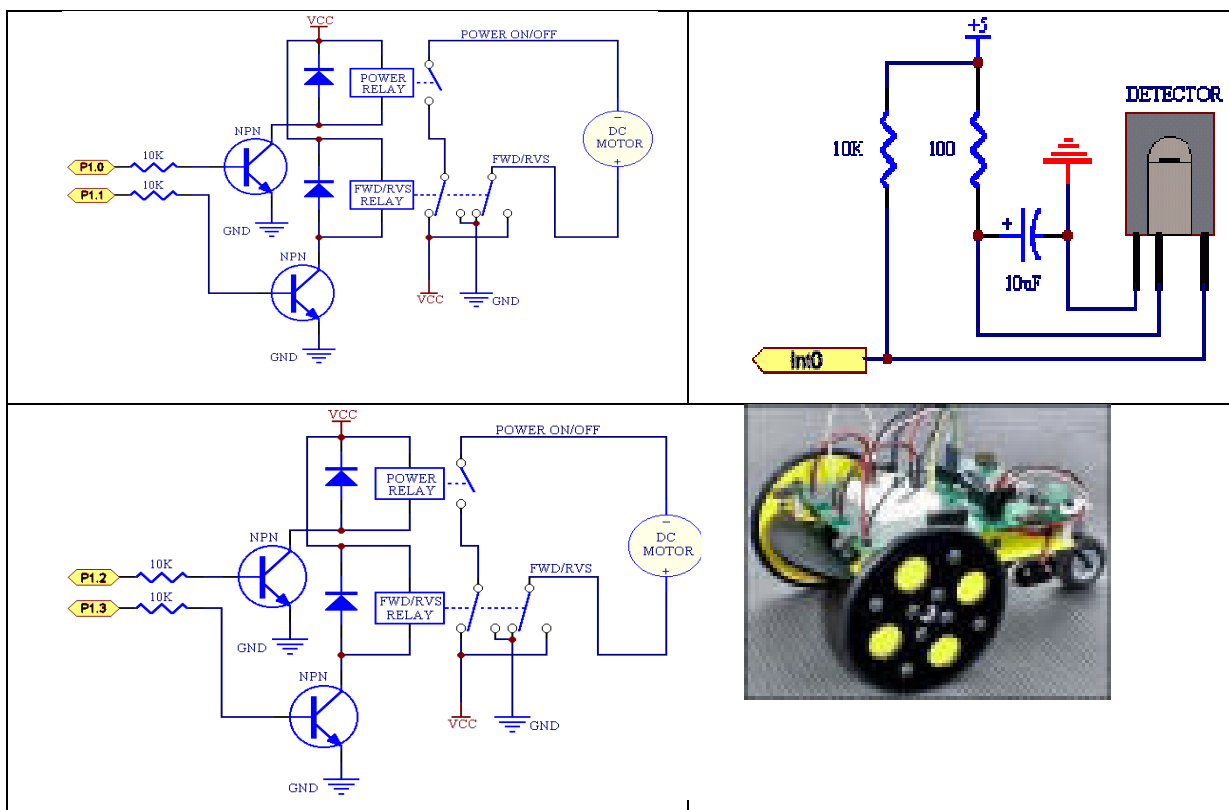
- A interrupção externa 0, atendida prioritariamente, deve trocar o que está na Porta P1 com o que está contido na posição de RAM externa 5000h.
- A interrupção externa 1, com baixa prioridade, deve transferir o que está armazenado na RAM externa do endereço 5000h para a RAM interna no endereço 7Fh;
- A interrupção gerada pelo T/C 0 (a cada 10 ms) com alta prioridade, deve executar uma rotina que copie o que está no endereço da RAM interna 7Fh para a posição 5200h da RAM externa;
- A cada 60 milissegundos (aproximadamente) e controlado pelo T/C 1 com interrupção de baixa prioridade, o dado armazenado na RAM externa no endereço 5200h deve ser enviado para a posição de memória externa 5000h.
  - O programa deve ficar em Loop infinito sempre executando o algoritmo acima.
  - Considerar o cristal do oscilador de 12 MHz.

7. Fazer um programa em Assembly do 8051 que calcule o valor da frequência de uma onda quadrada entrando pelo pino da Interrupção Externa 0 e envie-o para a interface serial RS232 a uma taxa de 4800,N,8,1. Considerar o cristal da CPU de 11,0592 MHz. Utilizar a interrupção Int0 sensível à descida de borda. O valor da frequência a ser enviada para a interface serial é um número hexadecimal de 16 Bits.



8. Escrever um programa em Assembly do 8051 que armazene uma tabela com 10 números na memória de programa. Ler cada um dos números e se o número for par, enviar para a porta P1 e para a interface serial RS232 a uma velocidade de 9600,N,8,1. Se o número for ímpar enviar para a porta P2 e para a interface serial RS232, a uma taxa de 4800,N,8,1. Contar o número de números pares e ímpares encontrados e armazenar o valor na RAM externa, nos seguintes endereços: (2030h) = quantidade de números pares e (2031h) = quantidade de números ímpares. O cristal é de 11.0592 MHz.

9. Um robô como mostrado na figura é acionado por dois motores de corrente contínua, um para cada roda, conforme o esquema, e possui um sensor localizado na parte da frente que tem a função de detectar a presença de obstáculos. Desenvolver um programa em Assembly do 8051 que controle o robô fazendo-o navegar por uma sala onde diversos obstáculos podem ser encontrados, de tal forma que ele não colida com nenhum.



O circuito do sensor está ligado na entrada de interrupção Int0 que gera um pulso negativo quando um obstáculo é detectado.

Os motores são acionados da seguinte maneira, conforme mostra o esquema eletrônico:

P1.0 = 1 → liga a alimentação do motor da roda da esquerda (P1.0 = 0 → desliga)

P1.2 = 1 → liga a alimentação do motor da roda da direita (P1.2 = 0 → desliga)

O movimento do robô é dado pela tabela:

P1.1	P1.3	Movimento
0	0	Robô movimenta-se para trás
0	1	Robô vira para a esquerda (roda esquerda pra trás e roda direita à frente)

1	0	Robô vira para a direita (roda esquerda à frente e roda direita para trás)
1	1	Robô movimenta-se à frente

O programa deve:

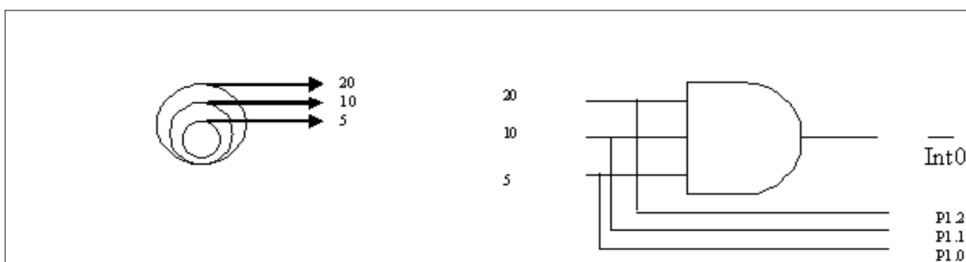
- Inicialmente movimentar o robô à frente.
- Quando o primeiro obstáculo for detectado o robô deve voltar atrás por 2 segundos e virar à direita por 2 segundos. A frequência do oscilador do microcontrolador é de 12 MHz.
- A cada obstáculo detectado o robô deve movimentar-se para trás por 2 segundos e inverter a última direção durante 2 segundos (direita, 2s, esquerda, 2s, direita, 2s, esquerda, 2s,.....).
- Após cada inversão de direção, o robô deve ser movimentado para frente até que novo obstáculo seja encontrado. Durante o movimento para trás e direita/esquerda a Int0 deve ser desabilitada.

#### 10. Automatizar uma Máquina de Doces com o Microcontrolador 89S52.

A máquina deve fornecer em cada operação, somente um doce que custa 20 centavos. A cada operação o programa reinicia e espera nova sequência de moedas. As moedas aceitas pela máquina são de 5 centavos, 10 centavos e 20 centavos.



Como cada moeda tem um tamanho diferente, um sensor óptico alinhado com o coletor de moedas determina qual moeda foi inserida. Apenas uma moeda pode ser inserida por vez. A inserção de uma moeda é detectada através da Interrupção Int0. O circuito de reconhecimento de moedas é mostrado abaixo e sua operação é de acordo com a Tabela 1.



P1.2 = 20 cents	P1.1 = 10 cents	P1.0 = 5 cents	Moeda inserida
1	1	1	Nenhuma (Int0 = 1)
1	1	0	5 centavos (Int0 = 0)
1	0	0	10 centavos (Int0 = 0)
0	0	0	20 centavos (Int0 = 0)

**Tabela 1 – Sinal nos pinos do 89S52 quando as moedas são inseridas.**

A Máquina de Doces tem três controles:

P2.0 = 1 → Fornecer o Doce

P2.1 = 1 → Devolver 5 centavos de troco

P2.2 = 1 → Devolver 10 centavos de troco

A operação da Máquina de Doce pode ser vista na Tabela 2 com todas as sequencias possíveis (S1 a S11) de inserção de moedas pelo usuário e com as ações a serem tomadas pelo microcontrolador.

	Seqüência de moedas inseridas	Ações de controle	
S1	$5 + 5 + 5 + 5 = 20$	Doce → P2.0 = 1	Troco = 0
S2	$5 + 5 + 5 + 10 = 25$	Doce → P2.0 = 1	Troco = 5 cents
S3	$5 + 5 + 5 + 20 = 35$	Doce → P2.0 = 1	Troco = 15 ( 10 + 5)
S4	$5 + 5 + 10 = 20$	Doce → P2.0 = 1	Troco = 0
S5	$5 + 5 + 20 = 30$	Doce → P2.0 = 1	Troco = 10
S6	$5 + 10 + 10 = 25$	Doce → P2.0 = 1	Troco = 5
S7	$5 + 10 + 20 = 35$	Doce → P2.0 = 1	Troco = 15 (10 + 5)
S8	$5 + 20 = 25$	Doce → P2.0 = 1	Troco = 5
S9	$10 + 10 = 20$	Doce → P2.0 = 1	Troco = 0
S10	$10 + 20 = 30$	Doce → P2.0 = 1	Troco = 10
S11	$20 = 20$	Doce → P2.0 = 1	Troco = 0

**Tabela 2 - Operação da Máquina de Doces**