

Lista de Exercícios 1

Conceitos envolvidos:

- a) Memória de Dados (interna e externa)
- b) Memória de Programa (interna e externa)
- c) Operações aritméticas e lógicas
- d) Portas
- e) Endereçamento a Bit
- f) Contadores e Temporizadores
- g) Interrupção
- h) Comunicação Serial RS232

1. Para o Microcontrolador da família MCS-51, responder aos itens abaixo de maneira objetiva:
 - a) Como acessar um dado armazenado na posição 1010h da RAM externa e copiar este dado no Registrador R7 ?
 - b) Quais os pinos da CPU do microcontrolador 8051 são utilizados para a operação do item anterior, e qual a função de cada um?
 - c) Se não for utilizada a memória de programa externa qual deve ser o nível lógico do pino EA?
 - d) Que instrução usar para acessar o endereço 8Bh da RAM interna de dados do microcontrolador?
2. Fazer um programa que escreva os números de 1 a 15H na memória interna de dados a partir do endereço 50H e na memória externa de dados a partir do endereço 2200H. Utilize modo de endereçamento indireto para escrita nas duas regiões de memória.
3. Fazer um programa que copie os dados da região de memória de dados externa de 2100H a 210FH para a região de memória de dados externa que inicia em 2300H.
4. Fazer um programa que copie os dados da área de memória de programa que devem estar armazenados a partir do endereço "TAB:" para a memória interna de dados a partir do endereço 30H. A seqüência de dados na memória de programa deve ser finalizada com o código 00. O programa deve contar o número de dados da seqüência, menos o último valor = 00, e armazenar o resultado no endereço 20h da RAM interna.
5. Escrever um programa que copie dados armazenados na RAM externa, com início na posição 8200H para a posição 8300H. A seqüência de dados deve ser finalizada com o

código 00. O programa deve contar o número de dados da seqüência, menos o último valor = 00, e armazenar o resultado no endereço 40h da RAM interna.

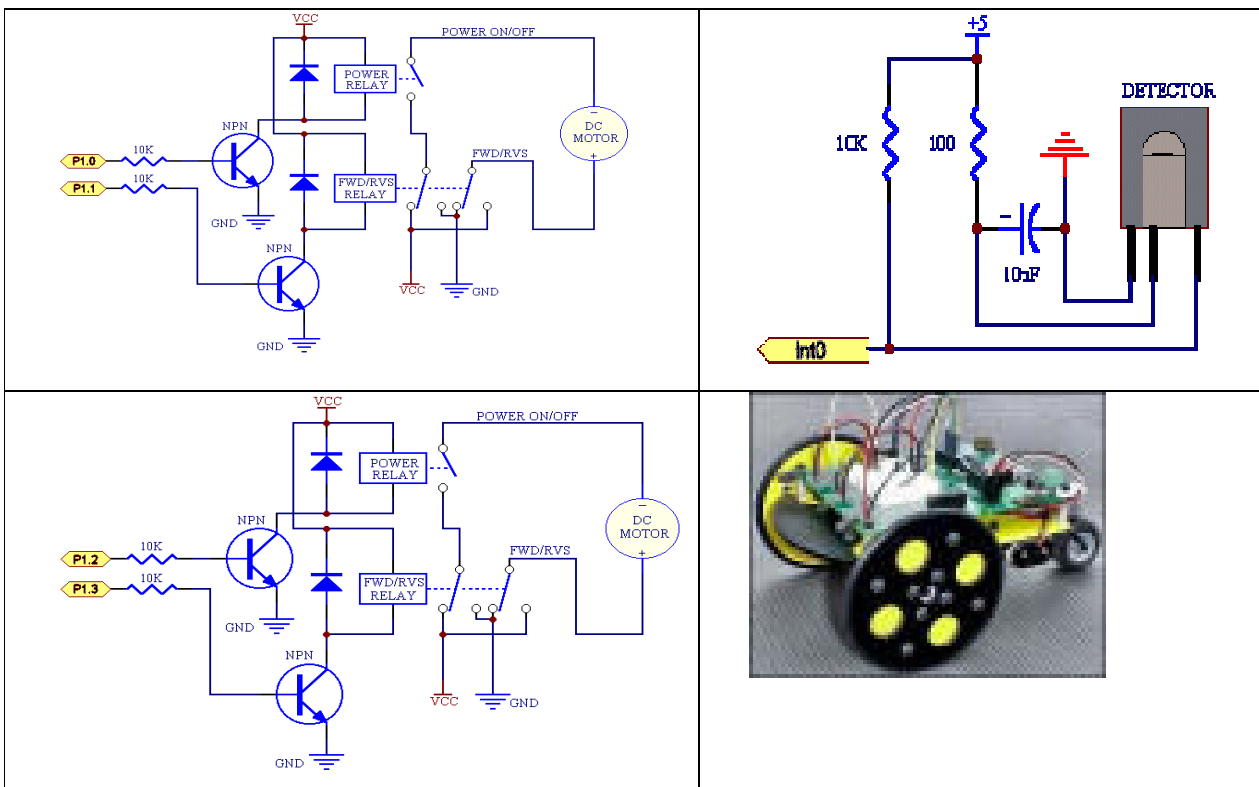
6. Escrever um programa que faça a soma de dois números de 24 bits (3 bytes) cada um. O primeiro número está armazenado nas posições **(R0+2)=MSB**, **(R0+1)**, **(R0)=LSB** da RAM externa. O segundo número está nas posições **(R1+2)=MSB**, **(R1+1)**, **(R1)=LSB**. O resultado deve ser colocado nas posições 42h=MSB, 41h e 40h=LSB da RAM interna.
7. Fazer um programa que armazene em ordem crescente, na RAM externa a partir do endereço 1000h, os elementos de uma tabela de bytes armazenada na Memória de Programa e terminada com o byte 00 (zero).
8. Fazer um programa que retorne no registrador **R6** o número de bits "UM" do registrador **R4**.
9. Usando instruções que endereçam bits no 8051, escrever um programa em Assembly que mova os bits do endereço de byte 20h para o endereço de byte 2Fh na ordem inversa, ou seja, o LSB do endereço 20h vai para o MSB do endereço 2Fh e assim sucessivamente até o MSB do endereço 20h ser movido para o LSB do endereço 2Fh.
10. Fazer um contador hexadecimal que coloque o valor de contagem na porta P1 em intervalos de 640 ciclos de máquina. Utilize o Timer 1.
11. Fazer um programa que gere uma onda quadrada na porta P1.7 com período de 2.56ms, considerando que o oscilador do microcontrolador é alimentado por um cristal de 12MHz. Utilize Timer 0 no Modo 0.
12. 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.
13. 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.

14. 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:

a) Inicialmente movimentar o robô à frente.

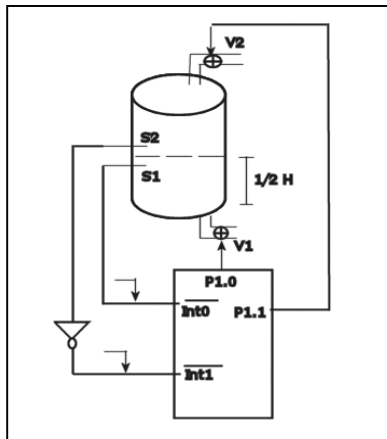
b) Quando o primeiro obstáculo for detectado o robô deve voltar atrás por 2 segundos e virar à direita por 2 segundos. A frequência do oscilador do microcontrolador é de 12 MHz.

c) 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,.....).

d) 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.

15. Um Microcontrolador 8051, com um oscilador de 11.0592 MHz comunica-se serialmente com um teclado a 4800 BPS e com uma impressora a 9600 BAUDS. Fazer um programa em Assembly que leia o dado do teclado e envie para a impressora, considerando as velocidades de comunicação de cada periférico.

16. 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)

17. Desenvolver um projeto com um Microcontrolador 8051 que controle a cancela de uma passagem de trem. Fornecer o software (em assembly) e o hardware em blocos com as seguintes especificações:

a) Os Pinos de controle da cancela são:

P1.0 = 1 → abre a cancela

P1.0 = 0 → fecha a cancela

b) A cancela possui os seguintes sensores:

P1.1 = 0 → cancela está fechada

P1.1 = 1 → cancela está aberta

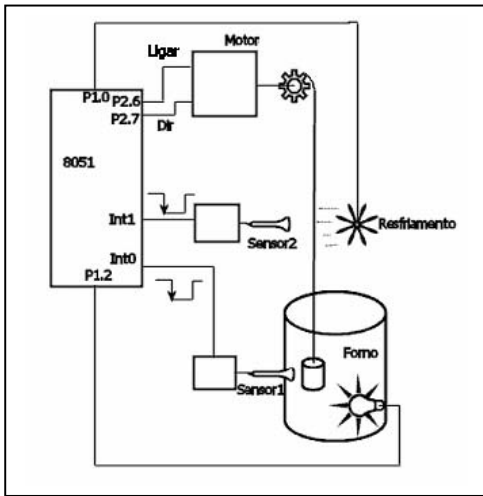
O sensor de presença de automóvel querendo passar está conectado à Interrupção Externa 0 ativada na descida de borda → Carro em frente à cancela fechada querendo passar.

A cancela deve ser aberta se não tiver trem passando. Caso contrário, a cancela deve ficar fechada.

c) O pino para o sensor de presença de trem passando é :

Interrupção Externa 1 ativada em nível baixo → trem passando, a cancela deve estar fechada, mesmo que ocorra presença de carro querendo passar. Esta interrupção deve ser de mais alta prioridade que a Interrupção externa 0, ou seja, se ocorrer deve sempre fechar a cancela.

18. 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 pára 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.

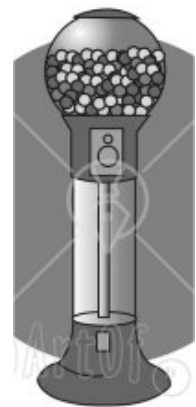
- Fornecer o programa em Assembly
- Fornecer os parâmetros de programação dos tempos envolvidos.

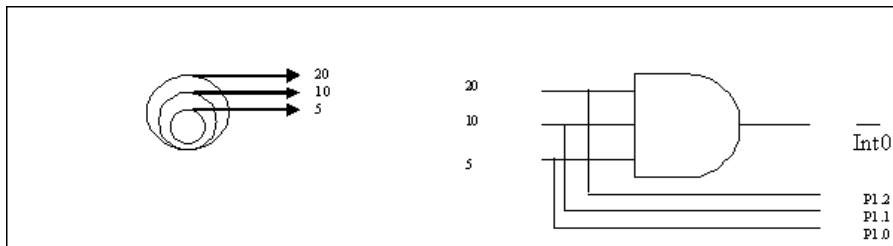
19. Automatizar uma Máquina de Doces com o Microcontrolador 89S52.

A máquina deve fornecer em cada operação, somente um doce que custa 20 centavos. A cada operação o programa re-inicia e espera nova seqüência 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 seqüências 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