

Sistemas Embarcados I – Laboratório 06: Robô de 1 eixo

Objetivos: Estudar o acionamento de um motor de passo. Construir códigos *assembly* para acionar pequenos automatismos.

O sistema é constituído por um motor de passo e seu circuito de acionamento, uma base em alumínio com uma régua impressa, um marcador de posição, um eixo sem, 2 chaves de fim de curso e um LED. Na Figura 1 se observa o diagrama esquemático e na Figura 2 a estrutura física.

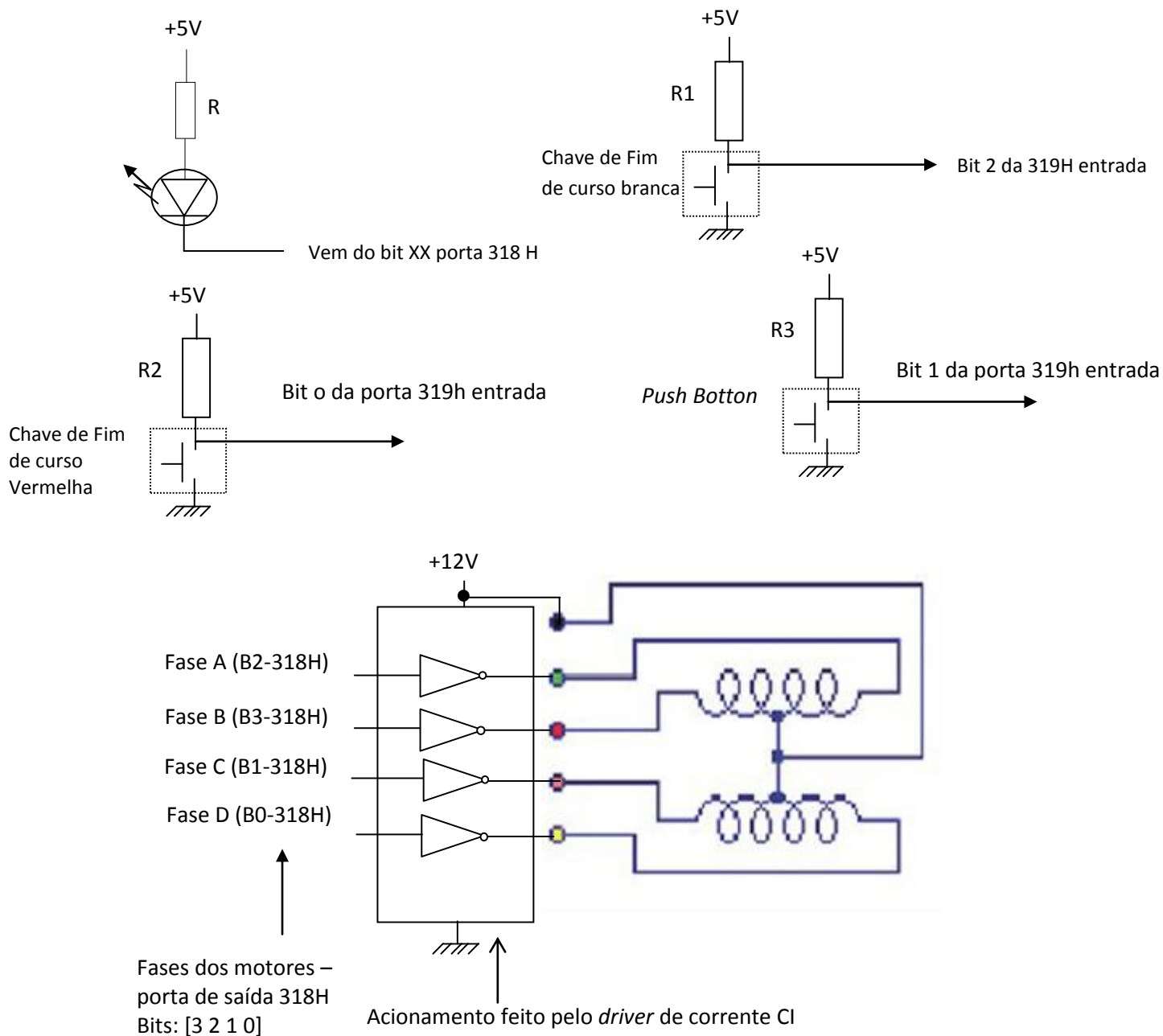
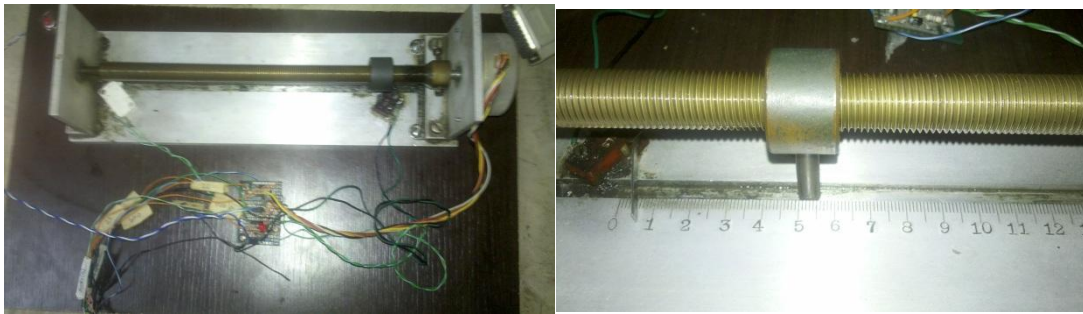


Figura 1: Esquemático do circuito implementado.



(a) Aspecto Físico.

(b) Marcador de posição e régua milimetrada.

Figura 2: Sistema físico.

Para acionar o motor de passo deve-se utilizar a seguinte tabela de fases:

Tabela de Fases			
A	B	C	D
1	0	1	0
0	1	1	0
0	1	0	1
1	0	0	1

Que deve ser declarada como:

segment data fasesdb 1010b, 0110b, 0101b, 1001b	; Tabela de fases.
--	--------------------

Portanto, pede-se realizar as seguintes tarefas:

- 1) Fazer um código que aciona o motor de passo de modo a, uma vez posto em marcha o motor, ao ser pressionada a tecla 'd' do teclado, o posicionador se mover para a direita; ao ser pressionada a tecla 'e', o posicionador se move para esquerda.
- 2) **Simular** com este sistema um portão de garagem com fechamento automático em cerca de 10 segs (observe que não se usa motor de passo para tal !). Vai ser necessário ler as 2 chaves de fim de curso e chave do tipo *push-botton*. Assuma portão completamente aberto quando o posicionador se encontra acionando a chave de fim de curso vermelha. Somente ao se "ligar" o circuito (logo após a execução do programa), para levar o portão para uma posição conhecida, o usuário deve permanecer pressionando o *push-botton* até que o portão feche completamente, quando então deverá liberá-lo. A partir deste ponto, todo o pedido de abertura do portão é comandado pelo *pushbotton*. Esta chave deve servir também como sinalizadora de presença de objeto/pessoa durante o fechamento (deslocamento do posicionador em direção à chave fim de curso branca). Uma vez que o portão esteja em movimento, o led deve ser aceso
- 3) Em seguida, descubra quantos passos do motor são necessários para andar 1 cm. Depois, faça um programa que simulará um robô de solda: posicione o sistema em 0 cm. Então simule que o posicionador deva parar nas posições 5 cm, 10 cm, e 15cm. Cada parada, simulando o tempo de soldagem, deve durar em torno de 1 seg. Durante a parada, o led deve acender simulando o processo de soldagem.

Atenção: A execução de uma instrução em um sistema microprocessado é, minimamente, milhares de vezes mais rápida que a dinâmica do sistema físico. Assim, entre um passo e outro você terá que introduzir um tempo de espera (*delay*). Portanto, use a rotina abaixo para fazê-lo (ajuste-a se for o caso). Você vai precisar também de outras versões desta rotina para simular os outros tempos.

Apêndice: Como acionar as portas de E/S usando instruções do 8086/8088

Em termos gerais, a comunicação do 8086/8088 com periféricos ocorre por meio de duas instruções:

- A instrução **in**, a qual realiza a **leitura** de dados;
- A instrução **out**, responsável pelo **envio** de dados.

O formato de ambas as instruções é similar. É preciso indicar sempre o **endereço da porta** onde os dados serão escritos ou de onde serão captados. Caso o endereço da porta esteja entre 0 e 255, ele pode ser escrito diretamente na instrução. Caso contrário, deve ser carregado previamente em DX.

Para ambas, é necessário, também, **um operando de 8 ou 16 bits** que servirá como fonte (para o caso de uma escrita na porta) ou destino dos dados (para uma leitura).

As partes de código a seguir apresentam exemplos do uso dessas instruções:

```
IN AL, 200h    ;Lê os dados da porta de endereço 200h e coloca o resultado em AL
ADD AL, 8h     ;Soma AL com 8
OUT 203h, AL   ;Escreve o conteúdo de AL na porta de endereço 203h

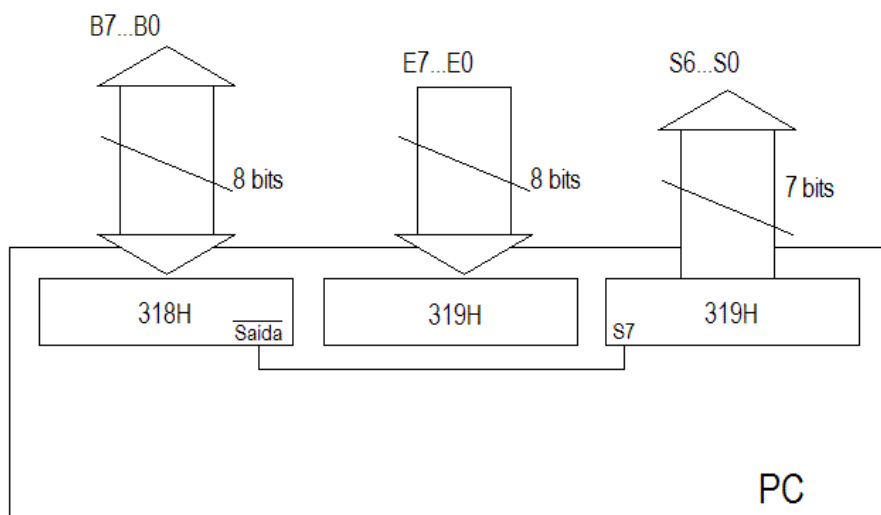
MOV DX, 318h   ;Carrega o endereço da porta em DX, que é maior que 255
IN AL, DX      ;Lê os dados e escreve o resultado em AL

MOV AL, 0xFF
OUT DX, AL     ;Escreve o conteúdo de AL na porta 318h
```

Observação: os operandos de fonte e destino de dados podem ser de 16 bits, mas esse uso não será abordado nos experimentos da disciplina.

Hardware

Alguns PCs do laboratório contam com uma placa especial, a qual provê duas portas de comunicação de 8 bits de largura cada. Além disso, duas tensões de alimentação diferentes (5VDC e 12 VDC) estão disponíveis para uso nas maquetes. A figura a seguir exibe as portas de comunicações e seus respectivos endereços.



A porta 319h possui dois barramentos diferentes. Um é empregado exclusivamente para entrada (leitura) de dados. O outro, independente, é utilizado apenas para escrita (saída) de dados. Desta

forma, instruções para leitura de dados “enxergarão” somente o estado dos bits E7...E0 do barramento de entrada, assim como a escrita dos dados afetará, exclusivamente, as linhas S7...S0.

A porta 318h, entretanto, é bidirecional. Antes de realizar uma instrução IN ou OUT para esta porta, é preciso certificar-se de que o sentido do barramento está selecionado corretamente. O bit 7 da porta 319h de saída é responsável por esse ajuste. Quando ele é 0 o barramento é saída, quando 1 o barramento é entrada. Só mude a direção do barramento para saída quando necessário, voltando para entrada assim que possível.

O código a seguir escreve o byte 0xF1 na porta 318h.

MOV DX, 319h	; Carrega o endereço da porta 319h
MOV AL, 0x00	
OUT DX, AL	; Ajusta 318h para saída (bit 7 = 0)
DEC DX	; Porta 318h
MOV AL, 0xF1	
OUT DX, AL	; Escreve o byte 0xF1 na porta 318h