

PROBLEMA 2: Sensor de distância com mostrador analógico

O problema do módulo 2 corresponde à um total de 11 pontos. O aluno deve mostrar o programa funcionando em sala de aula e enviar o código fonte até o final do dia.

Escreva uma aplicação embarcada que use o sensor ultrassônico HC-SR04 para medir a distância até um obstáculo e mostra o resultado num ponteiro analógico usando o servo motor, como na figura 1b. O mapeamento de distância para graus é mostrado na figura 1a.

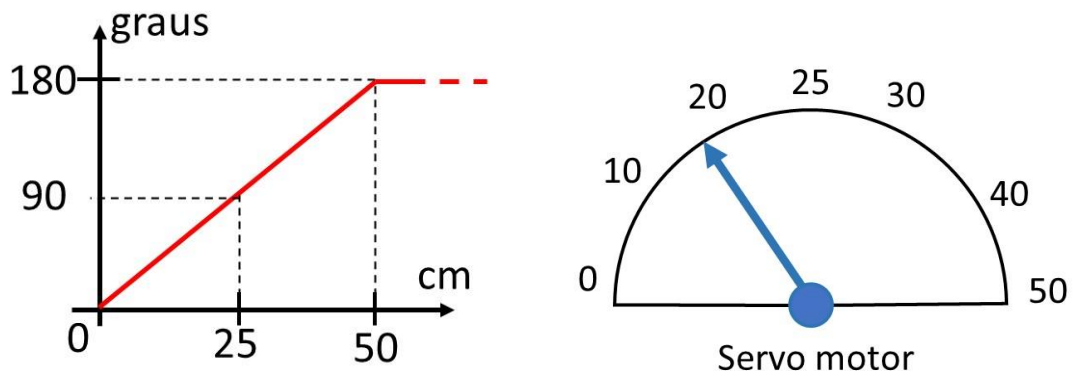


Figura 1a: Conversão de ângulo em graus Figura 1b: Display analógico de distância

Adicionalmente, o aluno pode mostrar a distância usando LEDs. Para isso, recomenda-se a sinalização mostrada na tabela 1.

Faixa de distância (cm)	Led 1 (P1.0) Vermelho	Led 2 (P4.7) Verde
Acima de 50	1	1
de 30 até 50	1	0
de 10 até 30	0	1
abaixo de 10	0	0

Tabela 1 – Sinalização dos LEDs em função da distância

A medição da distância deve acontecer na taxa de 10 ou de 20 medições por segundo. A sugestão é usar um sinal PWM para controlar o motor. A resposta do sensor de distância “echo” deve ser medida com o recurso de captura do timer. A cada medida, os LEDs e a posição angular do servo motor devem ser atualizados.

Dicas para elaboração de seu programa:

- 1) Prefira montar o sensor na vertical, um pouco afastado da superfície de sua mesa, pois ela pode refletir o sinal. Use um obstáculo grande que possa parar na vertical como um livro ou uma caixa. Em suma, tudo deve estar estável, do contrário você não terá certeza nas medições.
- 2) Elabore o programa por etapas. Teste cada uma delas em separado para ter certeza de seu correto funcionamento.
- 3) Inicie com a verificação do disparo do sensor e medição do sinal de eco. Prepare uma configuração com um obstáculo a uma distância conhecida e verifique se a medição está coerente. Varie a distância e confira. O recurso de breakpoint pode ajudar muito nesta etapa.
- 4) Depois, vá pouco a pouco adicionando as demais funções. Veja, então, que se deve quebrar o programa em pequenas funções.
- 5) Saiba que tentar escrever todo o programa de uma só vez é certeza de insucesso.
- 6) Sugestão para as funções, considerando que a duração do eco já está disponível:
 - a. `dist = calc_dist(echo);` Calcular a distância em cm
 - b. `leds(dist);` Acender os leds em função da distância
 - c. `prog_servo(dist);` Posicionar o eixo do servo

Se você já conseguiu fazer todo o problema, considere os aperfeiçoamentos abaixo.

- 7) Após ter tudo funcionando, você pode notar que as medições são ruidosas. É possível usar um filtro para reduzir o ruído. São sugestões o filtro de média móvel e o filtro mediana. Pode usar outros. Faça vários ensaios para ver se os resultados melhoram.
- 8) Algumas vezes surgem valores impossíveis. Por exemplo: uma medição resulta em 30 cm e a próxima medição resulta em 2m (é o teleporte!). Pense numa forma de eliminar valores absurdos.
- 9) A atualização de TA2 acontece a cada medida. Pense numa forma de realizar essa atualização de TA2 sem provocar flutuações no servo.

Sugestão de montagem:

