

# Título: Demolidor.

Nome da equipe: #include <coragem.h>  
 Problemática escolhida: Segurança Doméstica

**Resumo** – O projeto “Demolidor” (que faz referência ao super-herói da Marvel que possui um sonar altamente desenvolvido que atua como a sua visão) surgiu a partir da necessidade de promover uma maior segurança doméstica para pessoas com cegueira, seja ela de nascença, seja ela adquirida ao longo da vida. Tal projeto baseia-se no uso de sonares ultrassônicos que - ao encontrarem algum obstáculo dentro do seu raio de atuação - repassam essa informação para os motores de vibração, estes que são responsáveis por traduzir para o usuário os dados recebidos dos sonares por meio de sua intensidade de vibração.

**Palavras-chave** — *Arduino, linguagem C, Motor de Vibração, Sensor de Distância, distância, projeto.*

**Abstract** - *The “Daredevil” project (which references the Marvel superhero who has a highly developed sonar that acts as his vision) arose from the need to promote greater home security for people with blindness, either she from birth or she acquired it over a lifetime. This project is based on the use of ultrasonic sonars that - when encountering an obstacle within their radius of action - pass this information on to the vibration motors, which are responsible for translating the data received from the sonars through their vibration intensity.*

**Keywords** – *Arduino, C language, Vibration Motor, Distance Sensor, Distance, project.*

## I. OBJETIVO

O experimento tem por principal objetivo auxiliar pessoas cegas a se locomoverem em ambientes domésticos de forma segura.

É importante frisar que por “ambiente doméstico” entende-se qualquer ambiente particular de um

grupo restrito, geralmente familiar, podendo também ser de outro tipo. Com isso, o projeto visa viabilizar a livre circulação de pessoas com deficiência visual em âmbitos domésticos que frequentem menos, tais como a casa de amigos e vizinhos, já que - a depender do tempo que se tem essa condição, costuma-se conhecer bem certos ambientes, sobretudo o seu próprio lar - como também facilitar o processo de adaptação de pessoas que adquiriram cegueira recentemente à sua própria casa.

## II. MATERIAL UTILIZADO

1. 1 Arduino UNO;
2. 2 Módulos de Motor de Vibração;
3. 2 Módulos de Sensor de Distância Ultrassônico HC-SR04;
4. 1 Chave Gangorra On/Off MR-21S-2BB-F de 2 terminais;
5. 1 Bateria de 9V;
6. 1 Cabo Adaptador de Alimentação de Bateria de 9V;
7. 1 Case para Arduino UNO.

## III. RESUMO

### *Introdução*

O projeto “Demolidor” se inspira na ecolocalização dos morcegos com finalidade de expandir a percepção espacial de seu usuário. Para isso utiliza um sensor de distância que emite sinais ultrassônicos e por meio do eco desses sinais em objetos é capaz de mensurar distâncias.

O cálculo da distância de um objeto pode ser feito com base na velocidade do som no ar - aproximadamente 340 m/s - e o tempo entre a

emissão do pulso ultrassônico e a recepção de seu eco. Para isso basta usar a equação I.

$$\text{Distância} = (\text{Tempo} \times \text{velocidade do som}) / 2 \quad (\text{I})$$

Essa equação é a mesma da velocidade média - estudada no movimento uniforme. A divisão por dois é feita pois o tempo fornecido pelo sensor é para o trajeto de ida e volta do som. Se não for feita estaremos considerando duas vezes a distância.

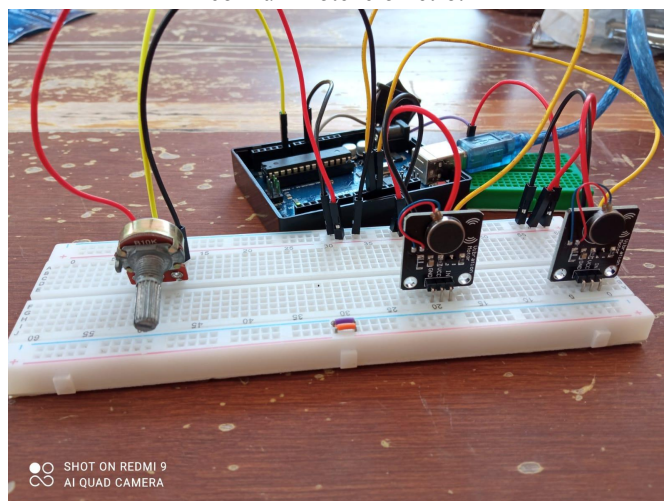
Para transmitir a informação da distância do objeto para o usuário é utilizado um motor de vibração. O arduino faz o cálculo dessa distância e emite um sinal PWM utilizado no motor. À medida que há aproximação do objeto, o duty cycle - porcentagem de tempo em que o sinal do PWM é nível alto - aumenta, isso faz com que o motor vibre com mais intensidade.

Em algum momento da vida já nos deparamos com a situação de estarmos em casa no período da noite e de repente acontecer uma falta de energia elétrica. Com isso tivemos a experiência de estarmos “cegos” em nosso ambiente doméstico e mesmo que bem familiarizados com o local ainda assim acabamos andando com dificuldades, trombamos em mesas, nos machucamos em quinas e alguns de nós podem até ter tropeçado e sofrido uma queda. Nesse contexto, a perda de visão e suas dificuldades são temporárias, mas para algumas pessoas - as que passaram por alguma forma de perda permanente da visão - isso é um desafio permanente e que requer tempo e esforço para adaptação. Tendo em vista isso, o projeto demolidor foi desenvolvido para auxiliar as pessoas com deficiência visual nesse processo visando uma locomoção com mais segurança e facilidade.

Para a implementação do projeto foram utilizados dois Sensores Ultrassônicos HC-SR04 e dois Módulos Motor de Vibração conectados diretamente aos pinos digitais do Arduino. Os pinos relacionados aos sensores HC-SR04 estão configurados como entradas e os módulos de vibração configurados como saída.

Inicialmente foi realizado um teste com os dois Módulos Motor de Vibração e um Potenciômetro, a fim de se observar os níveis de crescimento e decrescimento da intensidade de vibração dos módulos.

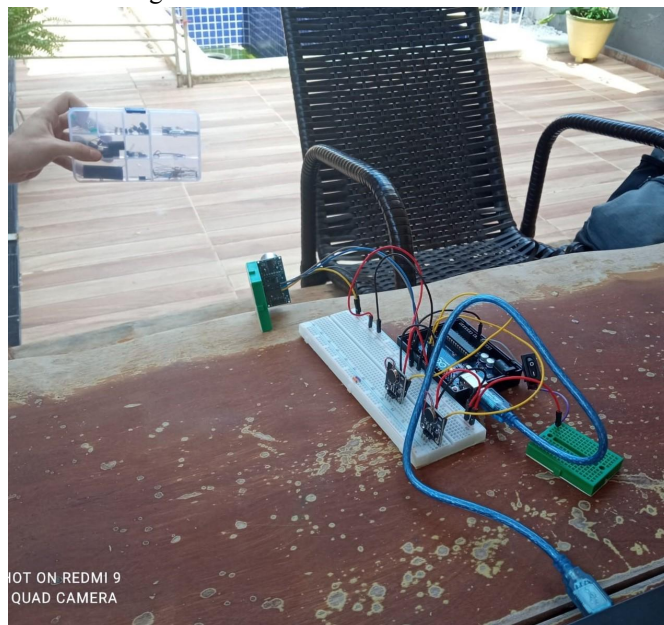
Figura 1. Teste dos Módulos Motor de Vibração com um Potenciômetro.



Fonte: imagem do acervo da equipe.

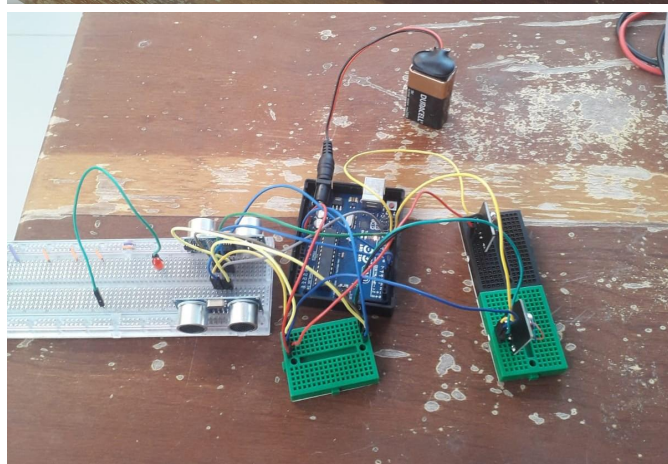
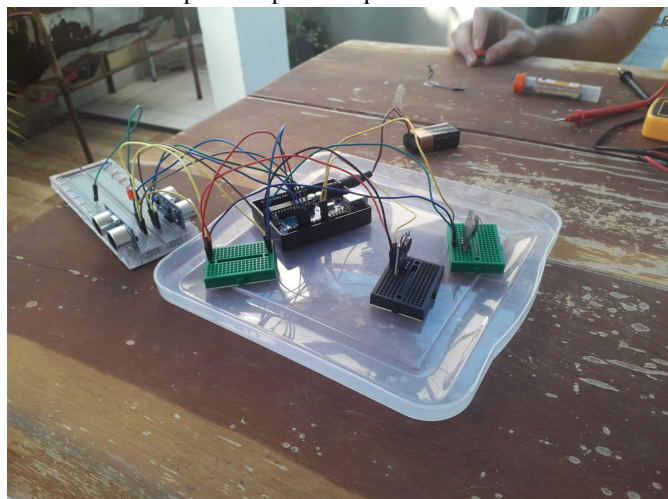
Em seguida, foi realizado um teste com o Sensor Ultrassônico e os módulos, com o objetivo de analisar e registrar informações referentes tanto aos limites de percepção de obstáculos do sensor, quanto à distância mínima - a priori - necessária para ativar os módulos.

Figura 2. Teste do Sensor Ultrassônico.



Fonte: imagem do acervo da equipe.

Figura 3. Projeto “Demolidor” montado por completo na protoboard.



Fonte: imagem do acervo da equipe.

O Sensor Ultrassônico se caracteriza pela ótima precisão e baixo custo, sendo possível medir distâncias no raio de 2 cm à 4 m, com precisão de aproximadamente 3 mm. O módulo de vibração é capaz de emitir diversos padrões de vibração com diferentes intensidades, suportando sinais PWM. Portanto, ambos os componentes foram considerados, devido às suas características, os mais adequados para realização do projeto.

#### IV. JUSTIFICATIVA

No experimento realizado, ainda na forma de protótipo, testou-se a sensibilidade de um usuário aos motores de vibração quanto a existência de algum objeto no raio de atuação dos sensores ultrassônicos - na frente de sensores foi colocado a mão de um integrante do grupo e o usuário deveria dizer em que sentido estava a mão, esquerda ou direita, sem usar a visão. Além disso, foi testado colocar duas mãos, uma em frente a cada sensor, e o usuário disse qual delas estava mais próxima do sensor de acordo com a intensidade de vibração dos dois motores. Para visualizar o experimento veja o [vídeo](#).

##### Montagens

##### a) Descrição do Funcionamento:

TABELA I

Tabela de conexões de cada componente do projeto.

Componente	Conexões
Bateria de 9V	O polo negativo está ligado diretamente ao pino adaptador para a alimentação; enquanto o polo positivo está ligado à chave On/Off, esta está ligada ao pino adaptador.
Chave On/Off	O 1º terminal está ligado ao polo positivo da bateria; enquanto o 2º terminal está ligado ao pino adaptador.

Sensor de Distância mais à esquerda	Terminal GND ligado ao pino GND do Arduino; Terminal ECHO ligado ao pino digital 2 do Arduino; Terminal TRIG ligado ao pino digital 3 do Arduino; Terminal VCC ligado ao pino 5V do Arduino.
Sensor de Distância mais à direita	Terminal GND ligado ao terminal GND do Sensor Ultrassônico mais à esquerda; Terminal ECHO ligado ao pino digital 4 do Arduino; Terminal TRIG ligado ao pino digital 5 do Arduino; Terminal VCC ligado ao pino terminal VCC Sensor Ultrassônico mais à esquerda.
Módulo Motor de Vibração mais acima	Terminal IN ligado ao pino digital PWM 9 do Arduino; Terminal VCC ligado ao terminal VCC do Módulo de Vibração mais abaixo; Terminal GND ligado ao terminal GND do Módulo de Vibração mais abaixo.

Módulo Motor de Vibração mais abaixo	Terminal IN ligado ao pino digital PWM 10 do Arduino; Terminal VCC ligado ao terminal VCC do Sensor Ultrassônico mais à direita; Terminal GND ligado ao pino GND do Arduino;
--------------------------------------	--

Tomando por base a Tabela I acima referente às conexões existentes em todo o projeto, pode-se inferir que o funcionamento do mesmo pode ser dividido em três grandes áreas que estão interligadas entre si, são elas: **alimentação**, **Sensores de Distância** e **Módulos Motor de Vibração**.

Todo o sistema é alimentado por uma *bateria de 9V* que o deixa portátil - acoplada com um *cabo adaptador de alimentação* - por meio de uma *chave gangorra On/Off* que, quando desativada, não permite a passagem de corrente elétrica entre a bateria e a placa Arduino, deixando o projeto desligado.

Quando a *chave On/Off* está ativada, é permitido um fluxo de corrente elétrica entre a bateria e a placa Arduino, ligando o projeto. Com isso, os dois *Sensores Ultrassônicos* passam a funcionar, emitindo uma onda ultrassônica a cada 10 $\mu$ s (dez microssegundos) cada um. Se essa onda é captada pelo receptor quer dizer que há uma barreira física dentro do raio de funcionamento do sensor que possibilitou o ricocheteio da mesma e, a partir do tempo entre a emissão e a captação de tal onda, é calculado via software a distância entre o sensor e a barreira.

Os raios de funcionamento dos *sensores* foram limitados a dois metros via software para fins de praticidade e viabilidade do projeto. Portanto, apenas os dados captados dentro dessa faixa são considerados e repassados para os *Módulos Motor de Vibração*. Tais módulos estão ligados cada um em uma entrada digital PWM da placa Arduino, sendo possível - então - controlar a intensidade de vibração dos mesmos de acordo com as distâncias

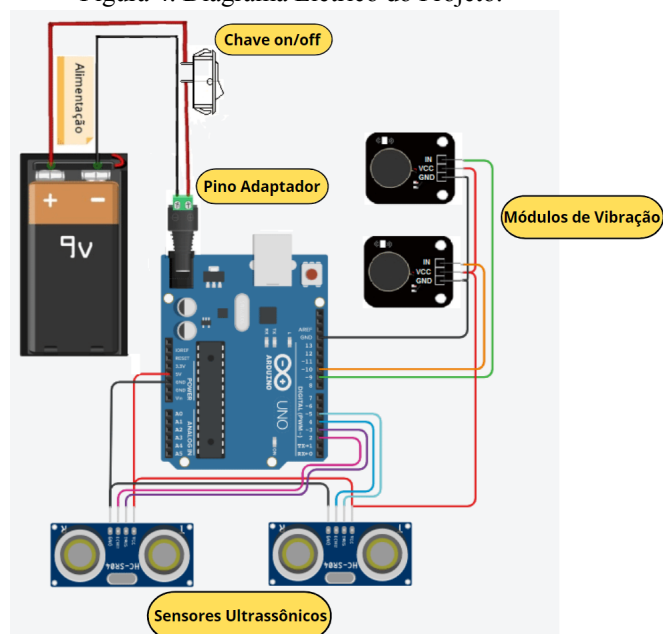


calculadas com os dados dos *Sensores de Distância*. Com isso, à medida que o usuário do projeto “Demolidor” aproxima-se de uma barreira, a intensidade de vibração vai aumentando gradativamente.

Além disso, é importante ressaltar que cada *Sensor Ultrassônico* está apontado para uma direção diferente, ou seja, um está voltado mais para a esquerda, enquanto o outro está voltado mais para a direita. Sendo assim, cada *Módulo de Vibração* - que está associado a um sensor distinto - é capaz de repassar para o usuário do projeto também um senso de direção, já que estarão posicionados na região da barriga, estando um mais à esquerda e o outro mais à direita.

### b) Diagrama Elétrico:

Figura 4. Diagrama Elétrico do Projeto.



Fonte: imagem produzida pela equipe.

### c) Código para implementação do projeto:

Figura 5. Imagens do código por partes.

```

1  #include <Arduino.h>
2
3  // Pino usado para disparar os pulsos do sensor
4  #define PinTRIG1 3
5  #define PinTriG2 5
6  // Pino usado para ler a saída do sensor
7  #define PinECHO1 2
8  #define PinECHO2 4
9  // Pino usado para disparar o motor de vibração (PWM)
10 const int PinVibra1 = 9; // PWM
11 const int PinVibra2 = 10; // PWM
12
13 double TempoEcho1 = 0;
14 double TempoEcho2 = 3;
15
16 // Em metros por microsegundo
17 const double VelocidadeSom mporus = 0.000340;
18
19 ///////////////////////////////////////////////////
20 // Configuração do setup //
21 ///////////////////////////////////////////////////
22
23 void setup()
24 {
25     // Configuração do pino TRIG como saída
26     // Inicializado em nível baixo
27     pinMode(PinTRIG1, OUTPUT);
28     pinMode(PinTriG2, OUTPUT);
29     digitalWrite(PinTRIG1, LOW);
30     digitalWrite(PinTriG2, LOW);
31
32     // Configuração do pino ECHO como entrada
33     pinMode(PinECHO1, INPUT);
34     pinMode(PinECHO2, INPUT);
35
36     // Inicializa a porta serial
37     //Serial.begin(9600);
38 }
39
40 ///////////////////////////////////////////////////
41 // Função que envia o pulso de TRIG //
42 ///////////////////////////////////////////////////
43
44 void DisparaPulsoUltrassonico1()
45 {
46     // Para fazer o HC-SR04 enviar um pulso ultrassônico, nos temos
47     // que enviar para o pino de trigger um sinal de nível alto
48     // com pelo menos 10us de duração
49     digitalWrite(PinTRIG1, HIGH);
50     delayMicroseconds(10);
51     digitalWrite(PinTRIG1, LOW);
52 }
53 void DisparaPulsoUltrassonico2()
54 {
55     // Para fazer o HC-SR04 enviar um pulso ultrassônico, nos temos
56     // que enviar para o pino de trigger um sinal de nível alto
57     // com pelo menos 10us de duração
58     digitalWrite(PinTriG2, HIGH);
59     delayMicroseconds(10);
60     digitalWrite(PinTriG2, LOW);
61 }
62

```

```

64
65 ///////////////////////////////////////////////////
66 // Função que calcula a distância em metros //
67 ///////////////////////////////////////////////////
68
69 double CalculaDistancia(double tempo_us)
70 {
71     return((tempo_us*VelocidadeSom_mporus)/2);
72 }
73
74 ///////////////////////////////////////////////////
75 // Função que controla as vibrações //
76 ///////////////////////////////////////////////////
77
78 void Vibracao(double distancia, int pino)
79 {
80     /*
81     Utilizamos a função map para transformar uma escalar de 0 a 200 cm
82     em uma tensão de 5V a 3,7V - 3,7 V é a tensão mínima para funciona-
83     mento do módulo de vibração. No Arduino, para saídas analógicas, o
84     número 255 representa 5V e o número 188 representa 3,7V - 188 é ob-
85     tido por regra de três simples: (3,7/5)*255 = 188.
86     */
87     double conversao = 0;
88     conversao = map(distancia, 0, 200, 255, 188);
89     analogWrite(pino, conversao);
90 }
91
92 ///////////////////////////////////////////////////
93 // FUNÇÃO PRINCIPAL //
94 ///////////////////////////////////////////////////
95
96 void loop()
97 {
98     // Envia pulso para disparar os sensores
99     DisparaPulsoUltrassonico1();
100
101     // Mede o tempo de duração do sinal no pino de leitura(us)/entrada
102
103     TempoEcho1 = pulseIn(PinECHO1, HIGH);
104
105     //condicional para ativar primeiro motor apenas se a distância para
106     //uma barreira for menor que 2m.
107     if((CalculaDistancia(TempoEcho1)*100) <= 200)
108     {
109         Vibracao(CalculaDistancia(TempoEcho1)*100, PinVibra1);
110         //Serial.print("Distancia 1: ");
111         //Serial.println(CalculaDistancia(TempoEcho1)*100);
112     }
113
114 }
115
116 else{
117     Serial.println("Fora de Alcance 1!");
118     digitalWrite(PinVibra1, LOW);
119 }
120
121
122 //condicional para ativar segundo motor apenas se a distância para
123 //uma barreira for menor que 2m.
124 DisparaPulsoUltrassonico2();
125 TempoEcho2 = pulseIn(PinECHO2, HIGH);
126 if((CalculaDistancia(TempoEcho2)*100) <= 200)
127 {
128     Vibracao(CalculaDistancia(TempoEcho2)*100, PinVibra2);
129     //Serial.print("    Distancia 2: ");
130     //Serial.println(CalculaDistancia(TempoEcho2)*100);
131 }
132
133
134 else{
135     Serial.println("Fora de Alcance 2!");
136     digitalWrite(PinVibra2, LOW);
137 }
138 }
139

```

Fonte: material produzido pela equipe.

[Código no GitHub](#)

## V. DIFICULDADES ENCONTRADAS NA IMPLEMENTAÇÃO DO PROJETO

A maior dificuldade enfrentada na montagem do protótipo foi na leitura dos dois sensores ultrassônicos de forma simultânea. Primeiramente tentou-se gerar um único pulso de Trig para os dois sensores, porém, com a função *pulseIn* só conseguiu-se ler o sinal do pino Echo de um dos sensores. Isso acontecia porque com esse pulso de Trig comum aos dois sensores mandavam um pulso ultrassônico ao mesmo tempo e recebiam seu respectivo eco quase no mesmo instante. Como o Arduino só processa um comando de cada vez, enquanto o sinal do pino Echo de um sensor era lido, o outro era perdido. A solução para esse problema foi criar funções independentes de pulso Trig para cada sensor (linhas 44 a 52 para o sensor 1 e linhas 53 a 61 para o sensor 2) e logo depois que cada uma é executada, é feita a leitura do pino Echo correspondente. Podemos ver isso nas linhas 100 e 104 para o sensor 1 e nas linhas 124 e 125 para o sensor 2.

## VI. REFERÊNCIAS

1. [ARDUINO EDUCATION](#)
2. [CURSO DE ARDUINO - UDEMY](#)
3. [DOCUMENTAÇÃO DE REFERÊNCIA DO ARDUINO](#)
4. [HC-SR04 DATASHEET\(PDF\) - LIST OF UNCLASSIFIED MANUFACTURERS](#)
5. BANZI, MASSIMO; SHILOH, MICHAEL. **PRIMEIROS PASSOS COM ARDUINO: A PLATAFORMA DE PROTOTIPAGEM ELETRÔNICA OPEN SOURCE**. SÃO PAULO: NOVATEC EDITORA LTDA, 2015.