



## Ramo Estudantil IEEE - UEL



---

Angélica Carmelina de Souza (angelica.carmelina@uel.br)  
Arthur Marcon Obici (arthur.obici177@uel.br)  
Pedro Araújo Cordeiro (pedro.araujo.cordeiro@uel.br)  
Gleice Hellen Vidal dos Santos (gleice.hellen.vidal@uel.br)  
Bianca Rodrigues dos Santos de Almeida  
(bianca.almeida789@uel.br)

### RELATÓRIO FINAL: ALARME

Londrina  
2023



# Ramo Estudantil IEEE - UEL



---

Angélica Carmelina de Souza  
Arthur Marcon Obici  
Pedro Araújo Cordeiro  
Gleice Hellen Vidal dos Santos  
Bianca Rodrigues dos Santos de Almeida

## RELATÓRIO FINAL: ALARME

Relatório apresentado ao Ramo Estudantil  
IEEE da Universidade Estadual de Londrina.

**Diretor de Projetos:** Nathan Andreani Netzel  
**Gestores de Projetos:** Daniel Tresse Dourado, Levi Monteiro dos Santos

Londrina  
2023



## Ramo Estudantil IEEE - UEL



---

SOUZA, Angélica Carmelina. OBICI, Arthur Macon. CORDEIRO, Pedro Araújo. SANTOS, Gleice Hellen Vidal. ALMEIDA, Bianca Rodrigues dos Santos. **RELATÓRIO FINAL: ALARME**. 2023. Número total de folhas: 13. Relatório apresentado ao Ramo Estudantil IEEE da Universidade Estadual de Londrina, Londrina, 2023.

### RESUMO

O projeto Alarme é uma tecnologia produzida por meio do arduino, que é um microcontrolador. É utilizado como um objeto de segurança, onde é necessário a programação para que funcione adequadamente. Além disso, nele é utilizado o Led, que precisa ser instalado de maneira específica, juntamente com um resistor para evitar que queime. Há o sensor ultrassônico para medir distâncias. E para que o alarme emita os sinais sonoros, é utilizado um buzzer, que emite sons em diferentes frequências.

**Palavras-chave:** LED. Sensor Ultrassônico. Arduino. Buzzer.



---

## SUMÁRIO

### Sumário

1. INTRODUÇÃO.....	6
2. FUNDAMENTAÇÃO TEÓRICA .....	7
2.1 LED 2.1 .....	7
2.3 BUZZER 2.3.....	9
1. METODOLOGIA.....	10
4. RESULTADOS E DISCUSSÕES.....	15



## 1. INTRODUÇÃO

O presente projeto se trata de um equipamento de segurança, para advertir a respeito de algum sinal de ameaça ou perigo. O sistema é simples, onde usa equipamentos leves como Led, sensor ultrassônico e buzzer. É utilizado o microcontrolador Arduino UNO programado. O presente relatório abordará sobre os equipamentos utilizados, suas funções, a montagem do aparelho juntamente com seus resultados.

## OBJETIVOS

Fazer um sensor que ative a sirene e acenda o LED quando identificar movimento.  
Instalar um interruptor e um botão, para ligar/desligar o aparelho e desativar o alarme.

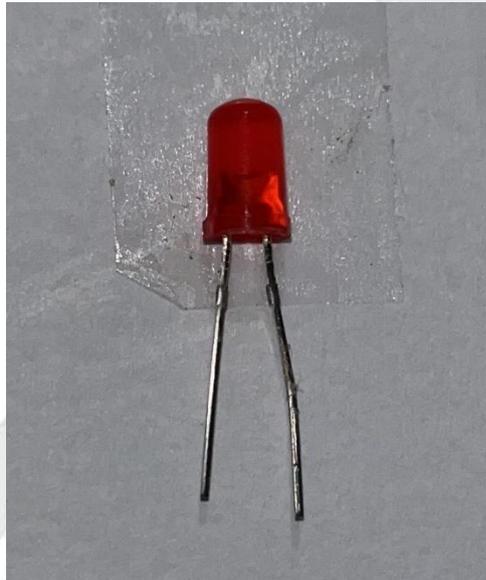
---

## 2. FUNDAMENTAÇÃO TEÓRICA

### 2.1 LED 2.1

É um diodo (componente eletrônico que permite a passagem da corrente elétrica em apenas um sentido) emissor de luz. Ele é utilizado para emitir luz em certos instrumentos. Ele é como uma lâmpada, só que não necessita de muita energia para funcionar. O Led possui polaridade, precisa ser instalado de maneira correta para que não queime. No caso, a parte positiva precisa estar na tensão mais alta, e a negativa, para o Terra. Ademais, ele precisa de um resistor para evitar a danificação.

**Figura 1 - Led**



**Fonte:** próprio autor.

## 2.2 SENSOR ULTRASSÔNICO

É um dispositivo que transforma a energia em ondas ultrassônicas. Ele mede distâncias de 2cm a 4m, enviando um sinal sonoro que colide com o obstáculo. Ele é conectado com o arduino e é utilizado um programa para que o sensor seja efetivado.

**Figura 2 – Sensor Ultrassônico**

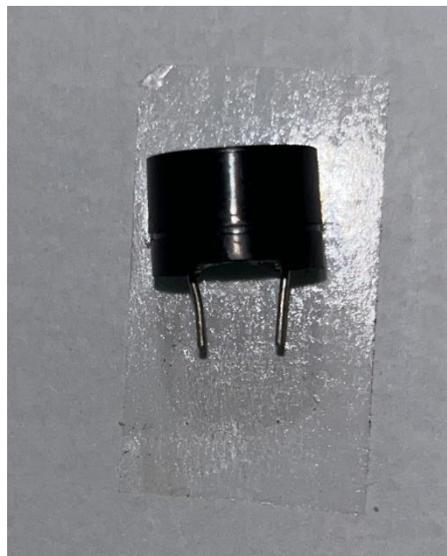


**Fonte:** <https://www.smartkits.com.br/sensor-ultrassonico-hc-sr04>

## 2.3 BUZZER

O buzzer é uma campainha. Ele emite sons em uma certa frequência. Ele é conectado ao arduino com um resistor de 100 ohm. Ele também tem polaridade, então precisa ser instalado corretamente.

**Figura 3 - Buzzer**



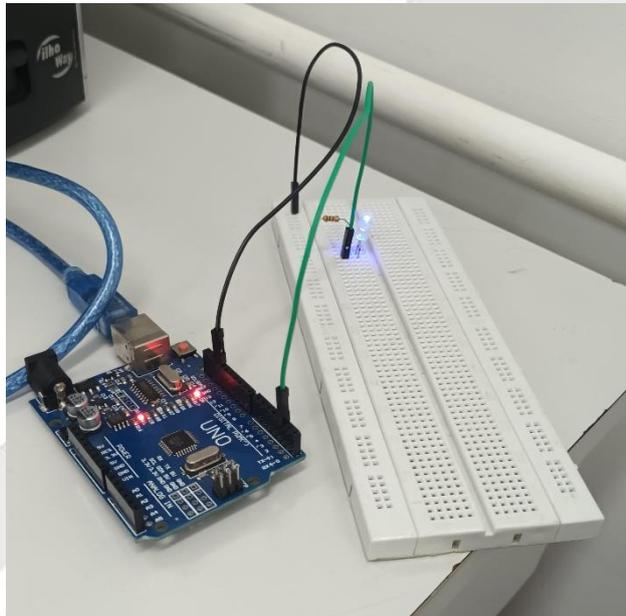
**Fonte:** Próprio autor.

## 1. METODOLOGIA

### 1.1 MONTAGEM

Começa-se pela montagem do LED no protoboard, o lado positivo para a tensão mais alta e o lado negativo para o terra do circuito. Coloca-se um resistor ao lado do LED, de 100ohms, para limitar a corrente passando pelo circuito. Depois, conecta-se o led no protoboard no arduino com o lado negativo no Digital Ground e o lado positivo no Digital I/O Pins 2 com a devida programação que será citada posteriormente.

**Figura 4 – Montagem do LED**



**Fonte:** Próprio autor



Já a montagem do sensor ultrassônico, seu VCC, Trig, GND e Echo é conectado ao 5 Volt Power Pin, Digital I/O Pins 4 e 5 e Ground Pin do arduino respectivamente. A biblioteca utilizada em sua programação é a ultrasonic.h.

Para a montagem do buzzer, é necessário tomar cuidado com sua polaridade, para ser conectado corretamente. Seu lado positivo deve ser conectado ao Digital I/O Pin 10, com um resistor 100ohms, e o lado negativo no GND do arduino.

## 1.2 PROGRAMAÇÃO

Para o funcionamento do projeto, foi elaborada uma programação que define o sensor com suas variáveis para armazenar a distância em cm, programado com distâncias de menos 20cm, led com suas funções de ligar e desligar com tempo de 300ms, buzzer para tocar em 1000Hz, e verificações de acionamento dos botões e a chave seletora.

### Apresentação do programa:

```
// Define os pinos do sensor ultrassônico
const int trigPin = 7;
const int echoPin = 6;

// Define os pinos do LED e do buzzer
const int ledPin = 8;
const int buzzerPin = 9;

// Define os pinos da chave seletora e do botão push
const int switchPin1 = 10;
```



## Ramo Estudantil IEEE - UEL



```
const int switchPin2 = 11;
const int buttonPin = 12;

// Define a variável para armazenar a distância medida
long distancia_cm;

// Inicialmente, o alarme está desligado
bool alarmeAtivo = false;
bool botaoPressionado = false; // Estado do botão pressionado

// Inicializa os pinos
void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(ledPin, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
  pinMode(switchPin1, INPUT_PULLUP);
  pinMode(switchPin2, INPUT_PULLUP);
  pinMode(buttonPin, INPUT_PULLUP);

  // Inicializa a porta serial para depuração
  Serial.begin(9600);
}

void loop() {
  // Verifica a posição da chave seletora
  int switchPosition1 = digitalRead(switchPin1);
  int switchPosition2 = digitalRead(switchPin2);
```



```
// Verifica se o botão push foi pressionado
if (digitalRead(buttonPin) == LOW && !botaoPressionado) {
    botaoPressionado = true;
    alarmeAtivo = !alarmeAtivo; // Inverte o estado do alarme
    delay(100); // Debounce para evitar leituras falsas
}

// Verifica se o botão foi liberado
if (digitalRead(buttonPin) == HIGH) {
    botaoPressionado = false;
}

// Verifica se a chave seletora voltou à posição de alarme ativo
if (switchPosition1 == LOW && switchPosition2 == HIGH && alarmeAtivo) {
    // Gera um pulso no pino TRIG do sensor ultrassônico
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    // Lê o tempo de retorno do pulso no pino ECHO
    distancia_cm = pulseIn(echoPin, HIGH) / 58; // Calcula a distância em centímetros

    // Imprime a distância no monitor serial em cm
    Serial.print("Distancia: ");
    Serial.print(distancia_cm);
    Serial.println(" cm");
}
```



## Ramo Estudantil IEEE - UEL



```
if (distancia_cm < 20) {  
    digitalWrite(ledPin, HIGH); // Liga o LED  
    delay(300); // Delay de 300 ms  
    digitalWrite(ledPin, LOW); // Desliga o LED  
    delay(300); // Delay de 300 ms  
    tone(buzzerPin, 1000); // Toca o buzzer em 1000 Hz  
} else {  
    digitalWrite(ledPin, LOW); // Desliga o LED  
    noTone(buzzerPin); // Para o som do buzzer  
}  
} else {  
    // Se a chave seletora estiver em outra posição (alarme desligado)  
    digitalWrite(ledPin, LOW); // Desliga o LED  
    noTone(buzzerPin); // Para o som do buzzer  
}  
}
```



---

### 4. RESULTADOS E DISCUSSÕES

Componentes e programação funcionando conforme o esperado no final do projeto. O Led emitiu luz e o alarme foi acionado conforme a identificação de aproximação de algum objeto a menos de 20cm. Apesar de alguns desafios com a programação e equipamentos, a montagem do alarme foi comprovadamente executado.