**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA**

**FATEC CARAPICUÍBA**

**AMS – ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**GUSTAVO PEREIRA DOS SANTOS**

**JULIO LIMA DA SILVA**

**PROJETOS DE ROBÓTICA COM ARDUÍNO**

**SENSOR DE ESTACIONAMENTO COM BUZZER**

**Carapicuíba**

**2025**

**GUSTAVO PEREIRA DOS SANTOS**

**JULIO LIMA DA SILVA**

**PROJETOS DE ROBÓTICA COM ARDUÍNO**

**SENSOR DE ESTACIONAMENTO COM BUZZER**

Trabalho Acadêmico apresentado à Fatec de Carapicuíba como engajamento para a criação de um laboratório de robótica na unidade.

Orientador: Profº Jose Luis Barboza Lobianco

**Carapicuíba**

**2025**

1. **Sensor de Estacionamento com buzzer:**

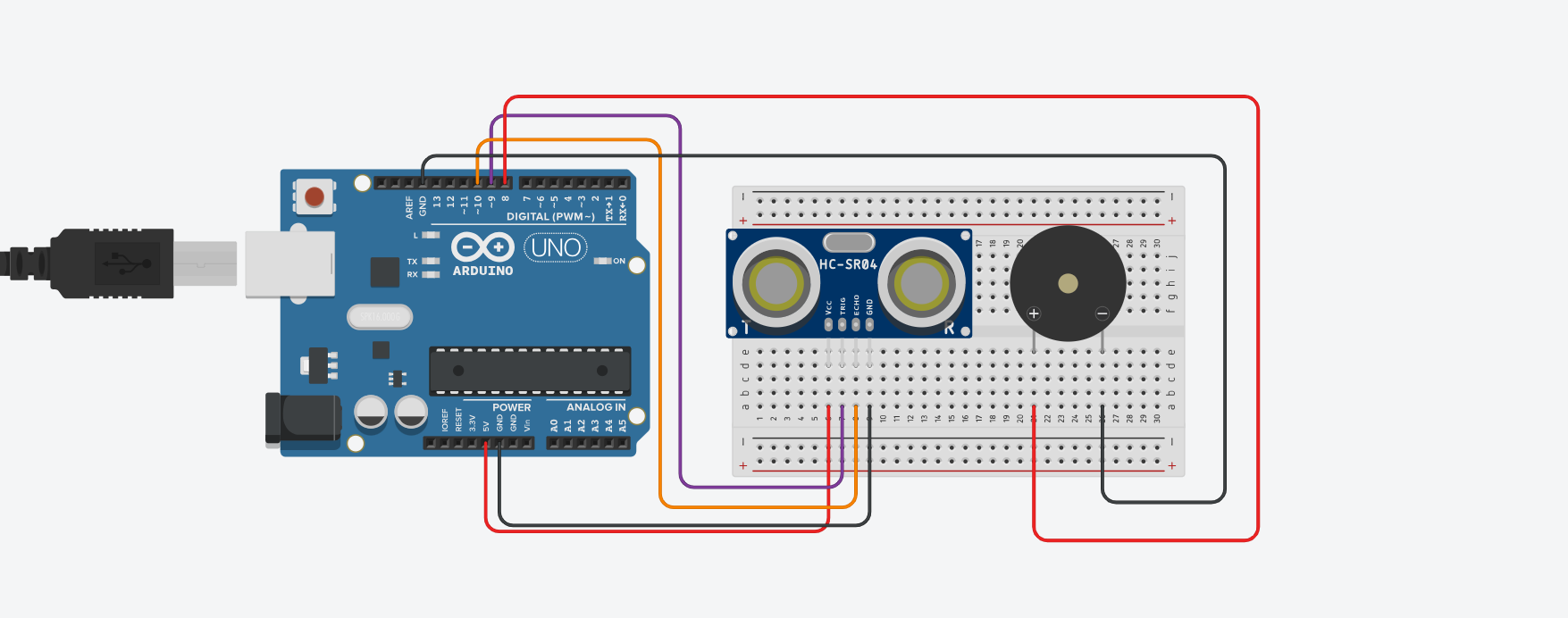
Este projeto simula um **sensor de estacionamento** como os encontrados em carros modernos. Ele utiliza um **sensor ultrassônico (HC-SR04)** para medir a distância de um obstáculo e um **buzzer** para emitir alertas sonoros com base na proximidade do objeto. É um ótimo exemplo de como aplicar sensores no dia a dia, além de introduzir conceitos como **medição de distância com som**, uso de **funções matemáticas** e **controle por PWM** ou sinal digital.

1. **Componentes:**

|  |  |
| --- | --- |
| **Componente** | **Quantidade** |
| Arduino Uno (ou similar) | 1 |
| Sensor Ultrassônico HC-SR04 | 1 |
| Buzzer ativo (ou passivo) | 1 |
| Protoboard | 1 |
| Jumpers (fios de conexão) | 1 kit |
| Cabo USB para o Arduino | 1 |

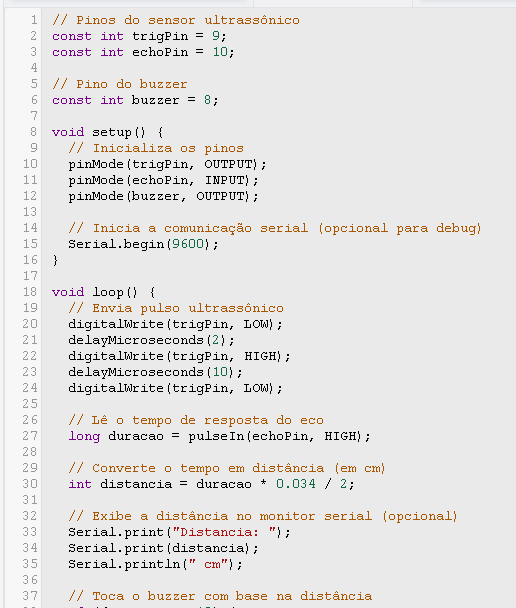
1. **Montagem do Circuito:**
   1. **Conecte o sensor ultrassônico:**

* VCC → 5v do Arduino
* GND → GND do Arduino
* TRIG → pino digital 9
* ECHO → pino digital 10
  1. **Conecte o buzzer:**
* Pino positivo do buzzer → pino digital 8
* Pino negativo do buzzer → GND
  1. **Imagem do Circuito:**

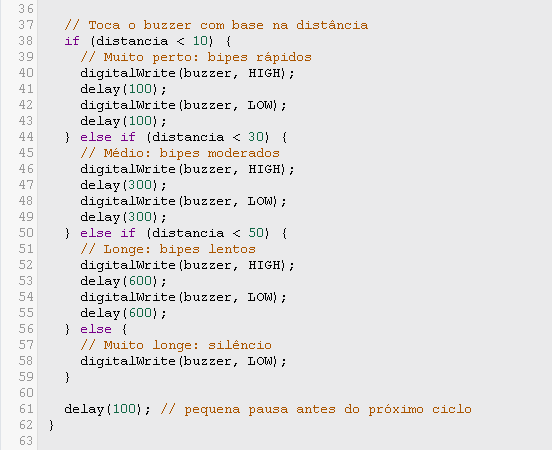
****

(Acervo Pessoal)

1. **Funcionamento do projeto:**
   1. **Imagem do Código:**

****

(Acervo pessoal)



(Acervo pessoal)

* 1. **Descrição do Código:**

O **sensor HC-SR04** envia pulsos ultrassônicos e mede o tempo de retorno para calcular a distância de um objeto à sua frente. Com base nessa distância, o Arduino controla o buzzer para emitir bipes:

* 1. > 50 cm: sem som;
  2. 30 a 50 cm: bipes lentos;
  3. 10 a 30 cm: bipes médios;
  4. < 10 cm: bipes rápidos.

Esse sistema ajuda a simular como os carros “avisam” quando estão chegando perto de um obstáculo – ideal para demonstrar aplicações reais de sensores.