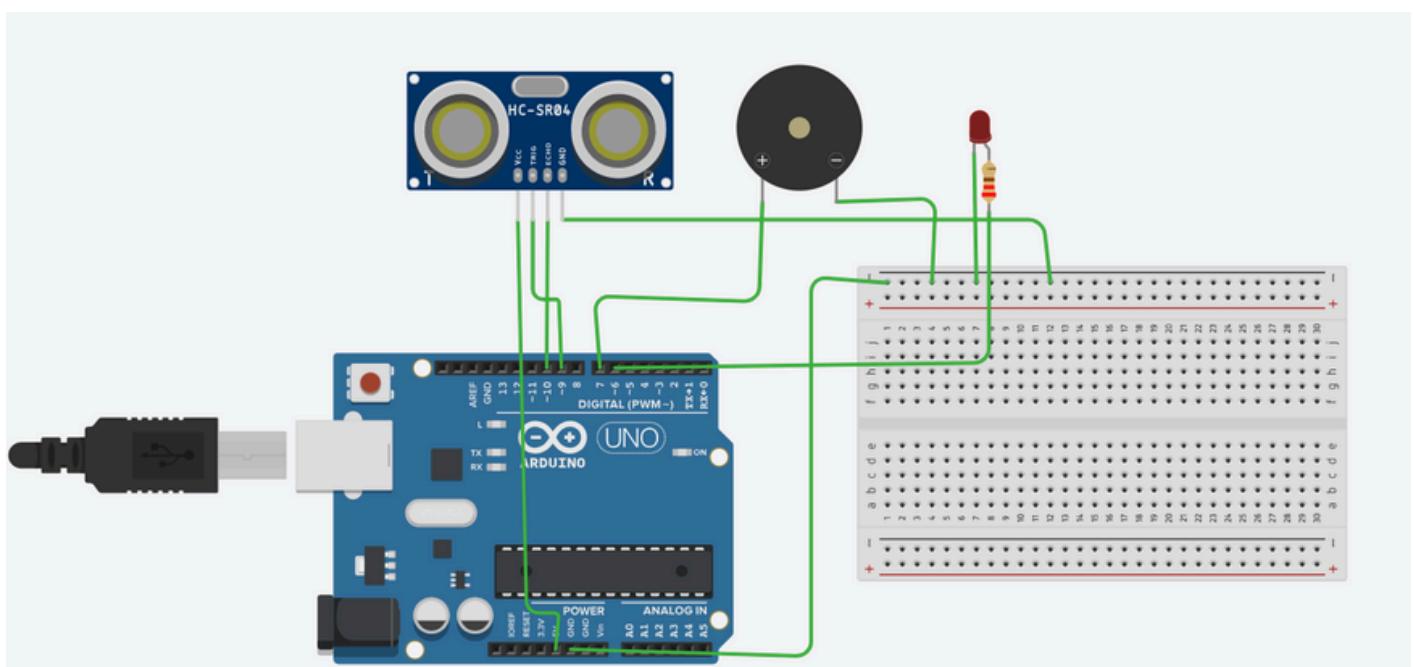


Distance Measurement Using Ultrasonic Sensor (HC-SR04) and Arduino

circuit connection



introduction

This project uses an ultrasonic sensor to send sound waves, measures the time for the echo to return, and calculates distance; the Arduino processes this data, triggering a buzzer and LED to signal proximity or distance thresholds, forming a non-contact proximity/distance system for applications like parking assist or liquid level sensing, using components like HC-SR04, Arduino, LED, and buzzer.

Working Principle

1. Trigger: The Arduino sends a short HIGH pulse (e.g., 10 μ s) to the sensor's Trigger (Trig) pin.
2. Emit: The sensor emits eight ultrasonic bursts (40kHz).
3. Echo: Sound waves hit an object and reflect back.
4. Receive: The sensor's Echo (Echo) pin goes HIGH when sound is sent and LOW when the echo returns.
5. Measure Time: The function measures this HIGH duration (travel time).
6. Calculate Distance: . The time is halved for the round trip.
7. Alert: If distance is within a set range, the LED might light up; if outside, the buzzer sounds, indicating an obstacle or range breach

Components

- * Arduino Board (e.g., Uno): The microcontroller brain.
- * HC-SR04 Ultrasonic Sensor: Emits and receives sound waves (VCC, GND, Trig, Echo pins).
- * Buzzer: Auditory alert.
- * LED: Visual alert (with a current-limiting resistor).
- * Breadboard & Jumper Wires: For connections.
- * USB Cable: For programming

Connection

- * Sensor VCC - Arduino 5V.
- * Sensor GND - Arduino GND.
- * Sensor Trig - Arduino Digital Pin 9.
- * Sensor Echo - Arduino Digital Pin 10.
- * Buzzer (positive) - Arduino Digital Pin 8.
- * Buzzer (negative) - Arduino GND.
- * LED (anode, with resistor) - Arduino Digital Pin 11.
- * LED (cathode) - Arduino GND.

Applications

- * Proximity Sensing: Detecting objects for robotics.
- * Parking Sensors: Cars use this for obstacle detection.
- * Liquid Level Measurement: In tanks.
- * Anti-Collision Systems: In automated guided vehicles (AGVs).

code

```
// Ultrasonic Sensor with LED and Buzzer

const int trigPin = 9;
const int echoPin = 10;
const int ledPin = 6;
const int buzzerPin = 7;

long duration;
int distance;

void setup() {
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(ledPin, OUTPUT);
    pinMode(buzzerPin, OUTPUT);
    Serial.begin(9600);
}

void loop() {
    // Trigger ultrasonic pulse
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    // Read echo
    duration = pulseIn(echoPin, HIGH);

    // Calculate distance (cm)
    distance = duration * 0.034 / 2;

    // Display distance
    Serial.print("Distance: ");
    Serial.print(distance);
    Serial.println(" cm");

    // Alert condition
    if (distance > 0 && distance < 20) {
        digitalWrite(ledPin, HIGH);
        digitalWrite(buzzerPin, HIGH);
    } else {
        digitalWrite(ledPin, LOW);
        digitalWrite(buzzerPin, LOW);
    }

    delay(500);
}
```