***PROJECT 1***

**1.0: Arduino - LDR Sensor and LED (Coding and Hardware connection)**

This example demonstrates the use of if() statements. It reads the state of a potentiometer (an analog input) and turns on an LED and buzzer if the potentiometer goes above a certain threshold level. It prints the analog value regardless of the level.

**Hardware Required**

Arduino UNO

LED

220 ohm resistor

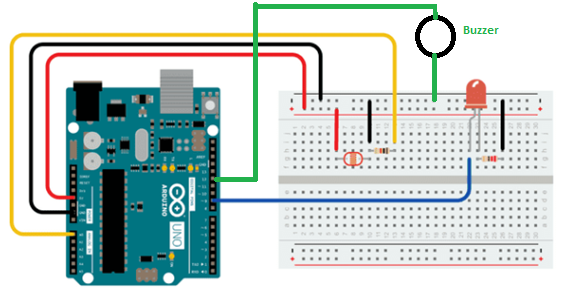
analog sensor (a photoresistor will do)

10k ohm resistor

hook-up wires

breadboard

**Schematic**



**Code**

#include <stdio.h>

const int analogPin = A0; // pin that the sensor is attached to

const int ledPin = 13; // pin that the LED is attached to

const int buzzer = 12; // pin that the buzzer is attached to

void setup() {

// initialize the LED pin as an output:

pinMode(ledPin, OUTPUT);

pinMode(buzzer, OUTPUT);

// initialize serial communications:

Serial.begin(9600);

}

void loop() {

// read the value of the potentiometer:

int analogValue = analogRead(analogPin);

// if the analog value is high enough, turn on the LED:

if (analogValue < 600) {

digitalWrite(ledPin, HIGH);

delay(100);

digitalWrite(ledPin, LOW);

delay(100);

digitalWrite(ledPin, HIGH);

delay(100);

digitalWrite(ledPin, LOW);

delay(100);

digitalWrite(ledPin, HIGH);

delay(100);

for (int i = 0; i <= 2; i++) {

tone(buzzer, 1000); // Send 1KHz sound signal...

delay(1000);

tone(buzzer, 2000);

delay(500);

noTone(buzzer); // Stop sound...

delay(1000); // ...for 1sec

}

} else {

digitalWrite(ledPin, LOW);

}

// print the analog value:

Serial.println(analogValue);

delay(1); // delay in between reads for stability

}

**1.1: Arduino – Buzzer and Ultrasonic Sensors (Coding and Hardware connection)**

This example demonstrates the use of if() statements. It reads the state of a ultrasonic sensor (an analog input) and turns on an LED and buzzer if the ultrasonic goes above a certain threshold level. It prints the analog value regardless of the level.

**Hardware Required**

Arduino UNO

LED

220 ohm resistor

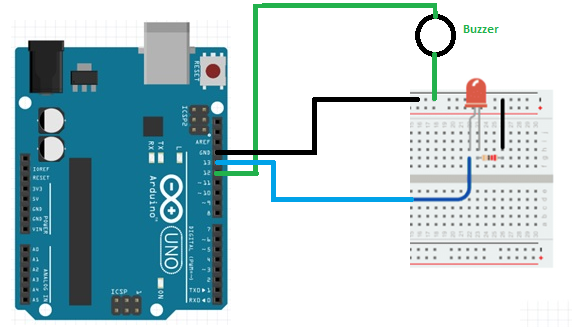
analog sensor (a ultrasonic will do)

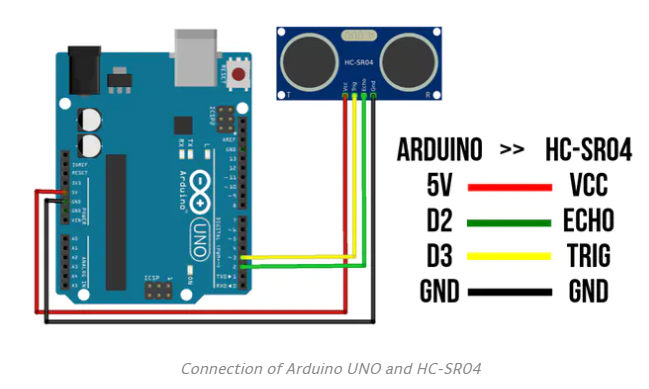
10k ohm resistor

hook-up wires

breadboard

**Schematic**

****



**Code**

#define echoPin 2 // attach pin D2 Arduino to pin Echo of HC-SR04

#define trigPin 3 //attach pin D3 Arduino to pin Trig of HC-SR04

// These constants won't change:

const int ledPin = 13; // pin that the LED is attached to

const int buzzer = 12; pin that the buzzer is attached to

long duration; // variable for the duration of sound wave travel

int distance; // variable for the distance measurement

void setup() {

// initialize the LED pin as an output:

pinMode(ledPin, OUTPUT);

pinMode(buzzer, OUTPUT);

pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT

pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT

// initialize serial communications:

Serial.begin(9600);

Serial.println("Ultrasonic Sensor HC-SR04 Test"); // print some text in Serial Monitor

Serial.println("with Arduino UNO R3");

}

void loop() {

// Clears the trigPin condition

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin HIGH (ACTIVE) for 10 microseconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculating the distance

distance = duration \* 0.034 / 2; // Speed of sound wave divided by 2 (go and back)

// Displays the distance on the Serial Monitor

Serial.print("Distance: ");

Serial.print(distance);

Serial.println(" cm");

// if the analog value is high enough, turn on the LED:

if (distance < 30) {

digitalWrite(ledPin, HIGH);

delay(100);

digitalWrite(ledPin, LOW);

delay(100);

digitalWrite(ledPin, HIGH);

delay(100);

digitalWrite(ledPin, LOW);

delay(100);

digitalWrite(ledPin, HIGH);

delay(100);

for (int i = 0; i <= 3; i++) {

tone(buzzer, 1000); // Send 1KHz sound signal...

delay(1000);

tone(buzzer, 2000);

delay(500);

noTone(buzzer); // Stop sound...

delay(1000); // ...for 1sec

}

} else {

digitalWrite(ledPin, LOW);

}

delay(1); // delay in between reads for stability

}