Ardunio Codes

LED Blinking

int led = 13;

void setup() {

  // initialize digital pin LED\_BUILTIN as an output.

  pinMode(led, OUTPUT);

}

// the loop function runs over and over again forever

void loop() {

  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the voltage level)

  delay(1000);                       // wait for a second

  digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW

  delay(1000);                       // wait for a second

}

Gas Sensor

int ledPin\_1 = 2;  // BUZZER POSITIVE

int ledPin\_2 = 3;  // BUZZER NEAGATIVE

int ledPin\_3 = 13;  // green led positive

int ledPin\_4 = 12;  /// green neGATIVE

int ledPin\_5 = 6;  // red led positive

int ledPin\_6 = 7;  // red neagtive

int mq8Pin = A0;         // MQ-8 sensor connected to analog pin A0

int sensorValue = 0;     // Variable to store sensor value

int threshold = 350;     // Define a threshold for gas detection

void setup() {

  pinMode(ledPin\_1, OUTPUT);

  pinMode(ledPin\_2, OUTPUT);

  pinMode(ledPin\_3, OUTPUT);

  pinMode(ledPin\_4, OUTPUT);

  pinMode(ledPin\_5, OUTPUT);

  pinMode(ledPin\_6, OUTPUT); // Set LED pin as output

         // Set up the LCD (16 columns and 2 rows)

  Serial.begin(9600);         // Initialize serial communication

}

void loop() {

  sensorValue = analogRead(mq8Pin);   // Read the analog value from the MQ-8 sensor

  Serial.print("Gas Sensor Value: ");

  Serial.println(sensorValue);  // Print the sensor value in Serial Monitor

  // Check if the sensor value exceeds the threshold

  if (sensorValue > threshold) {

    digitalWrite(ledPin\_2, LOW);  // Turn on the LED

    digitalWrite(ledPin\_1, HIGH);

    digitalWrite(ledPin\_5, LOW);  // Turn on the LED

    digitalWrite(ledPin\_6, HIGH);

    digitalWrite(ledPin\_4, LOW);

  } else {

  delay(1000);  // Wait for a second before the next reading

    digitalWrite(ledPin\_1, LOW);

    digitalWrite(ledPin\_4, LOW);

    digitalWrite(ledPin\_6, LOW);  // Turn on the LED

    digitalWrite(ledPin\_4, HIGH);

}}

Ultrasonic

const int trigPin = 2;

const int echoPin = 3;

const int buzzerPin = 8;  // Pin for the piezo buzzer

float length = 0;

const float thresholdDistance = 30.0;  // Set the threshold distance in cm

void setup() {

  Serial.begin(9600);

  pinMode(trigPin, OUTPUT);

  pinMode(echoPin, INPUT);

  pinMode(buzzerPin, OUTPUT);  // Set the buzzer pin as output

}

void loop() {

  // Send a short pulse to trigger the ultrasonic sensor

  digitalWrite(trigPin, LOW);

  delayMicroseconds(2);

  digitalWrite(trigPin, HIGH);

  delayMicroseconds(10);

  digitalWrite(trigPin, LOW);

  // Measure the duration of the echo pulse

  long duration = pulseIn(echoPin, HIGH);

  // Calculate the distance in cm

  length = duration \* 0.0343 / 2;

  // Display the distance

  Serial.print("Distance: ");

  Serial.print(length);

  Serial.println(" cm");

  // Check if the distance is below the threshold

  if (length < thresholdDistance) {

    // Emit a sound if the object is close

    tone(buzzerPin, 1000);  // Play a tone at 1000 Hz

    delay(200);              // Sound duration

    noTone(buzzerPin);       // Stop the tone

  }

  delay(500);  // Delay between measurements

}

Water Sensor

int sensorPin = A3;

int sensorValue = 0;

int value;

void setup() {

// put your setup code here, to run once:

  Serial.begin(9600);

  pinMode(sensorPin, INPUT);

}

void loop() {

  // put your main code here, to run repeatedly:

  //sensorValue = analogRead(sensorPin);

  value = analogRead(sensorPin);

 if (value<=480){

    Serial.println("Water level: 0mm - Empty!");

  }

else if (value>480 && value<=530){

    Serial.println("Water level: 0mm to 5mm");

  }

  else if (value>530 && value<=615){

    Serial.println("Water level: 5mm to 10mm");

  }

  else if (value>615 && value<=660){

    Serial.println("Water level: 10mm to 15mm");

  }

else if (value>660 && value<=680){

    Serial.println("Water level: 15mm to 20mm");

  }

else if (value>680 && value<=690){

    Serial.println("Water level: 20mm to 25mm");

  }

else if (value>690 && value<=700){

    Serial.println("Water level: 25mm to 30mm");

  }

else if (value>700 && value<=705){

    Serial.println("Water level: 30mm to 35mm");

  }

else if (value>705){

    Serial.println("Water level: 35mm to 40mm");

  }

  delay(2000);

}

LED Fad in Fade Out

int led = 9;           // the PWM pin the LED is attached to

int brightness = 0;    // how bright the LED is

int fadeAmount = 5;    // how many points to fade the LED by

// the setup routine runs once when you press reset:

void setup() {

  // declare pin 9 to be an output:

  pinMode(led, OUTPUT);

}

// the loop routine runs over and over again forever:

void loop() {

  // set the brightness of pin 9:

  analogWrite(led, brightness);

  // change the brightness for next time through the loop:

  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:

  if (brightness <= 0 || brightness >= 255) {

    fadeAmount = -fadeAmount;

  }

  // wait for 30 milliseconds to see the dimming effect

  delay(30);

}