Experiment No.: 02

Experiment Name: Experimental study on controlling three different LEDs based on the range of the flame.

Objective:

Theory:

Required Hardwares with Quantity:

Required Software:

Working Procedure:

Hardware Arrangement Diagram:

Sketch:

Result and Output:

Conclusion:

Basic Knowledge:

Flame Sensor: A flame-sensor is one kind of detector which is mainly designed for detecting as well as responding to the occurrence of a fire or flame. This sensor detects flame otherwise wavelength within the range of 760 nm – 1100 nm from the light source. This sensor can be easily damaged to high temperature. So, this sensor can be placed at a certain distance from the flame. The flame detection can be done from a 100cm distance and the detection angle will be 60º. The output of this sensor is an analog signal or digital signal. These sensors are used in fire fighting robots like as a flame alarm.

The wavelength range from 760 nanometers (nm) to 1100 nanometers (nm) includes a portion of the near-infrared (NIR) spectrum. When it comes to flame detection, this wavelength range is often used because flames emit significant amounts of infrared radiation, including in the NIR range. Here's how it relates to flame detection:

IR Flame Detection: Flames emit infrared radiation, and this emission can vary depending on factors like the type of fuel and the temperature of the flame. The wavelength range you mentioned (760 nm to 1100 nm) is well-suited for detecting the presence of flames, particularly in applications where you want to monitor for the presence or absence of a flame.

Flame Sensors: Specialized flame sensors, such as IR flame sensors, are designed to detect the infrared radiation emitted by flames in this wavelength range. These sensors can be used for flame monitoring, safety systems, and industrial processes where flame detection is critical.

Flame Sensor Module The pin configuration of this sensor is shown below. It includes four pins which include the following. When this module works with a microcontroller unit then the pins are

• Pin1 (VCC pin): Voltage supply rages from 3.3V to 5.3V

• Pin2 (GND): This is a ground pin

• Pin3 (AOUT): This is an analog output pin

• Pin4 (DOUT): This is a digital output pin

Flame Sensor Internal Elements:

YG1006 Phototransistor In flame sensor, YG1006 Phototransistor is an NPN type Transistor that is coated with black epoxy to make it responsive to IR radiation. Although, this transistor from the outside looks like a 2- terminal block LED. The main function of this transistor is to detect light between 760 nm to 1100 nm wavelength range. This transistor includes two terminals where a longer terminal works like an emitter and a shorter terminal works like the collector and it doesn't include a base terminal similar to the normal transistor. Once it detects flame then current flows in between emitter & collector terminals.

Operational Amplifier (Op-Amp) The current generated by the IR photodiode or phototransistor is usually very small, so an operational amplifier is used to boost the signal to a usable level for further processing.

Trim pot The trimpot is also called a variable resistor with a preset knob and this knob is rotated to control the sensitivity of fire detection. If this knob is turned in a clockwise direction, then the sensor's sensitivity will be enhanced. Similarly, if it is turned in anticlockwise then the Flame sensor's sensitivity will be reduced. The sensitivity adjustment allows you to control how sensitive the flame sensor is to detecting flames or specific wavelengths of light, such as infrared (IR) or ultraviolet (UV) radiation emitted by flames. By turning the trimpot, you can increase or decrease the sensitivity of the flame sensor. Increasing sensitivity means the sensor will trigger at lower levels of detected radiation, while decreasing sensitivity makes it less sensitive and requires a stronger flame signal to trigger.

Power LED The function of the Power LED in this sensor is to indicate the sensor's power supply is turned ON/OFF. Once we connect the power supply to the sensor, the LED will be activated.

Output LED Once the flame sensor detects the flame then the red color LED will be turned ON. Once it does not notice any flame then this LED will be deactivated.

Code for Capturing Flame Sensor Readings:

```
// Pin definitions const
int flameSensorPin = A0;

// Analog pin connected to the flame sensor
void setup() {
// Initialize serial communication at 9600 baud rate
Serial.begin(9600);
}
void loop() {
// Read the flame sensor's analog value
int sensorValue = analogRead(flameSensorPin);

// Output the sensor value to the serial monitor
Serial.print("Flame sensor value: ");
Serial.println(sensorValue);

// Delay for a short period to avoid flooding the serial monitor with data
delay(500); // Adjust this delay if you want faster or slower readings }
```

Main Code:

```
// Pin definitions
const int flameSensorPin = A0;  // Analog pin connected to the flame sensor
const int greenLEDPin = 9;      // Green LED for far flame
const int yellowLEDPin = 10;    // Yellow LED for medium flame
const int redLEDPin = 11;       // Red LED for close flame
```

```
// Thresholds for flame detection
int lowThreshold = 200;   // Far flame, turn on green LED
int mediumThreshold = 500; // Medium flame, turn on yellow LED
int highThreshold = 800;   // Close flame, turn on red LED

void setup() {
  // Initialize serial communication for debugging
  Serial.begin(9600);

  // Set up LED pins as OUTPUT
  pinMode(greenLEDPin, OUTPUT);
  pinMode(yellowLEDPin, OUTPUT);
  pinMode(redLEDPin, OUTPUT);

  // Turn off all LEDs initially
  digitalWrite(greenLEDPin, LOW);
  digitalWrite(yellowLEDPin, LOW);
  digitalWrite(redLEDPin, LOW);
}

void loop() {
  // Read the flame sensor's analog value
  int sensorValue = analogRead(flameSensorPin);

  // Output sensor value for debugging
  Serial.print("Flame sensor value: ");
  Serial.println(sensorValue);

  // Control LEDs based on flame range
```

```
  if (sensorValue < lowThreshold) {
    // Flame is far (low intensity)
    digitalWrite(greenLEDPin, HIGH);  // Turn on Green LED
    digitalWrite(yellowLEDPin, LOW);  // Turn off Yellow LED
    digitalWrite(redLEDPin, LOW);     // Turn off Red LED
  }
  else if (sensorValue < mediumThreshold) {
    // Flame is at medium range (moderate intensity)
    digitalWrite(greenLEDPin, LOW);   // Turn off Green LED
    digitalWrite(yellowLEDPin, HIGH); // Turn on Yellow LED
    digitalWrite(redLEDPin, LOW);     // Turn off Red LED
  }
  else if (sensorValue < highThreshold) {
    // Flame is close (high intensity)
    digitalWrite(greenLEDPin, LOW);   // Turn off Green LED
    digitalWrite(yellowLEDPin, LOW);  // Turn off Yellow LED
    digitalWrite(redLEDPin, HIGH);    // Turn on Red LED
  }
  else {
    // Flame is extremely close or sensor value is very high
    digitalWrite(greenLEDPin, LOW);
    digitalWrite(yellowLEDPin, LOW);
    digitalWrite(redLEDPin, HIGH);    // Keep Red LED on as a warning
  }

  // Short delay to avoid overwhelming the serial monitor
  delay(100);
}
```