Experiment No.: 07

Experiment Name: Study and Observation of Arduino Uno Based LDR Sensor

Objective:

Theory:

Required Hardwares with Quantity:

Required Software:

Working Procedure:

Hardware Arrangement Diagram:

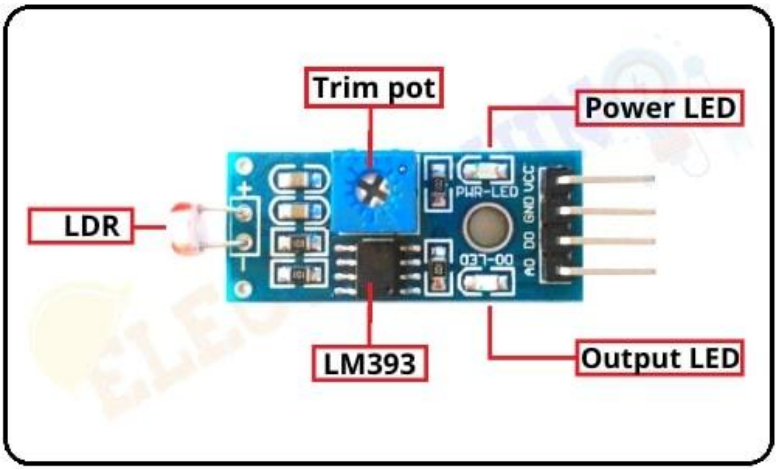
Sketch:

Result and Output:

Conclusion:

Basic Knowledge:

A **Light Dependent Resistor (LDR)**, also known as a photoresistor, is a passive electronic component whose resistance changes based on the intensity of light falling on its surface. LDRs are widely used in circuits that require light detection, such as automatic lighting systems and light-sensitive alarms.



1. LDR or Light Dependent Resistor

LDR or Light Dependent Resistor is one type of variable resistor. It is also known as a photoresistor. The Light Dependent Resistor (LDR) works on the principle of "Photo Conductivity". The LDR resistance is change according to the light intensity falls on the LDR. When light intensity increases on the LDR surface, then the LDR resistance will decrease and the element conductivity will increase. When light intensity decreases on the LDR surface, then the LDR resistance will increase and the element conductivity will decrease.

3. Variable Resistor (Trim pot)
The LDR sensor module has an onboard variable resistor or potentiometer, this variable resistor is a 10k preset. It is used to set the sensitivity of this LDR sensor. Rotate the preset knob to adjust the sensitivity of the light intensity detection. If we will rotate the preset knob in the clockwise direction, the sensitivity of the light intensity detection will increase. If it rotated counterclockwise direction, the sensitivity of the light intensity detection will decrease.

4. Power LED
This onboard LED indicates the LDR sensor module power supply is ON or OFF. When we turn on the sensor power supply this Green LED is also turn on.

5. Output LED
When the LDR sensor detects the light, the green LED is turn on. When the LDR sensor detects the darkness, the green LED is turn off.

Steps to Set the Threshold:

```
const int ldrPin = A0;  // LDR sensor input
void setup() {
  Serial.begin(9600);  // Start serial communication
}
void loop() {
  int ldrValue = analogRead(ldrPin);  // Read LDR value
  Serial.println(ldrValue);  // Print LDR value to serial monitor
  delay(500);  // Delay for readability
}
```

Ex.: If in bright light you see readings around 800, and in darkness you see readings near 100, you might choose a threshold around 500. This would mean that if the LDR value is below 500, it is considered dark, and if it is above 500, it is considered light.

Code 2:

```
// Pin assignments
const int ldrPin = A0;  // Analog pin where the LDR voltage divider is connected
const int ledPin = 13;   // Pin where an LED is connected (built-in LED for example)

// Light level threshold for detecting "dark" condition (adjust as needed)
int threshold = 500;  // Example threshold: adjust based on light conditions

void setup() {
  // Initialize the LDR sensor input pin
  pinMode(ldrPin, INPUT);

  // Initialize the LED output pin
  pinMode(ledPin, OUTPUT);
```

```
  // Start the serial communication for debugging
  Serial.begin(9600);
}

void loop() {
  // Read the analog value from the LDR voltage divider
  int ldrValue = analogRead(ldrPin);

  // Debugging: Print the LDR sensor value
  Serial.print("LDR Value: ");
  Serial.println(ldrValue);

  // Compare the LDR value with the threshold
  if (ldrValue < threshold) {
    // If the light level is below the threshold (dark), turn ON the LED
    digitalWrite(ledPin, HIGH);
    Serial.println("Dark condition detected! LED ON");
  } else {
    // If the light level is above the threshold (light), turn OFF the LED
    digitalWrite(ledPin, LOW);
    Serial.println("Light condition detected! LED OFF");
  }

  // Add a small delay to avoid too fast polling
  delay(500);
}
```