

# AlCoders Light Lab

## Arduino Physics Kit Manual

### Introduction

Welcome to the **AlCoders Light Lab**! This kit helps you explore two exciting projects using Arduino:

1. **Laser Security System** – Detects intrusions using a laser and an LDR.
2. **RGB Color Mixer** – Controls an RGB LED using potentiometers.

Each project demonstrates fundamental physics concepts related to **light, optics, and electronics**.

---

## 1. Laser Security System

### Project Overview:

The **Laser Security System** detects when an object interrupts a laser beam. When the laser is blocked, the LDR detects a drop in light intensity, triggering an alarm. This concept is used in security alarms, automated door systems, and light-based sensors.

### Components Required

- 1 x Arduino Uno
- 1 x Laser module (5V)
- 1 x LDR (Light Dependent Resistor)
- 1 x 10k $\Omega$  Resistor
- 1 x Buzzer
- Jumper wires
- Breadboard

### Physics Concept

- **Light Detection:** LDRs vary their resistance based on light intensity. More light means lower resistance, while less light increases resistance.
- **Security Systems:** Used in real-world alarm systems to detect unauthorized entry.
- **Ohm's Law ( $V = IR$ ):** Resistance change in the LDR alters voltage, which the Arduino detects.

### Circuit Connections:

1. Connect one end of the **LDR** to **5V**.
2. Connect the other end of the **LDR** to **A0** and also to **GND** via a **10k $\Omega$  resistor**.

3. Connect the **Laser module** to **digital pin 9** and **GND**.
4. Connect the **Buzzer** to **digital pin 8**.

### Arduino Code:

```
const int LDR = A0;
const int buzzer = 8;
const int laser = 9;

void setup() {
  pinMode(buzzer, OUTPUT);
  pinMode(laser, OUTPUT);
  pinMode(LDR, INPUT);
  Serial.begin(9600);
  digitalWrite(laser, HIGH); // Turn on the laser
}

void loop() {
  int lightValue = analogRead(LDR);
  Serial.println(lightValue);

  if (lightValue < 300) { // Adjust threshold based on ambient light
    digitalWrite(buzzer, HIGH);
  } else {
    digitalWrite(buzzer, LOW);
  }
  delay(500);
}
```

### How to Test:

1. Upload the code to Arduino.
2. Ensure the laser points directly at the LDR.
3. Block the laser beam with an object.
4. If the buzzer sounds, the system is working correctly.

### Possible Improvements:

- Use a relay to trigger a **stronger alarm** or **LED indicator**.
  - Implement a **wireless notification system** using **Bluetooth** or **Wi-Fi**.
- 

## 2. RGB Color Mixer

### Project Overview:

The **RGB Color Mixer** allows users to mix different colors using an **RGB LED** and three potentiometers, each controlling the intensity of Red, Green, and Blue light.

## Components Required:

- 1 x Arduino Uno
- 1 x RGB LED (Common Cathode)
- 3 x Potentiometers (10k $\Omega$  each)
- Jumper wires
- Breadboard

## Physics Concept:

- **Color Mixing:** Red, Green, and Blue light combine to form different colors.
- **Additive Color Model:** Used in TVs and displays.
- **Voltage Control:** Potentiometers adjust PWM values, which control LED brightness.

## Circuit Connections:

1. Connect the **RGB LED**:
  - Red pin to **D9**
  - Green pin to **D10**
  - Blue pin to **D11**
  - Common Cathode to **GND**
2. Connect the **Potentiometers**:
  - Middle pins to **A0, A1, A2**
  - One side to **5V**, other to **GND**

## Arduino Code:

```
const int redPin = 9;
const int greenPin = 10;
const int bluePin = 11;
const int potRed = A0;
const int potGreen = A1;
const int potBlue = A2;

void setup() {
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  int redVal = analogRead(potRed) / 4;
  int greenVal = analogRead(potGreen) / 4;
  int blueVal = analogRead(potBlue) / 4;

  analogWrite(redPin, redVal);
  analogWrite(greenPin, greenVal);
  analogWrite(bluePin, blueVal);

  Serial.print("Red: "); Serial.print(redVal);
```

```
Serial.print(" Green: "); Serial.print(greenVal);  
Serial.print(" Blue: "); Serial.println(blueVal);  
delay(500);  
}
```

## How to Test:

1. Upload the code to Arduino.
2. Turn the potentiometers to change Red, Green, and Blue intensity.
3. Observe the **color change** on the LED.

## Possible Improvements:

- Add **predefined colors** using buttons.
  - Control colors using a **Bluetooth app**.
- 

## Conclusion

- **Laser Security System** teaches light detection and real-world security applications.
- **RGB Color Mixer** demonstrates color theory and light intensity control.

These projects provide hands-on learning about physics, electronics, and programming! ?

---

## Next Steps

Would you like additional **experiments, troubleshooting tips, or documentation on sensors**? Let us know!