

Motion Controlled Colour Changer

PIR Sensor (Passive Infrared Sensor)

A **PIR sensor** detects motion by sensing changes in infrared radiation. It consists of a **pyroelectric sensor** that detects heat emitted by objects, especially humans and animals. When a moving object with a different temperature than the background enters its detection range, the sensor generates an **electrical signal**, which can be processed by an Arduino or microcontroller.

The PIR sensor has **three main pins**:

- **VCC** – Power supply (usually **5V** or **3.3V**)
- **GND** – Ground connection
- **OUT** – Digital signal output (**HIGH** when motion is detected, **LOW** otherwise)

It is widely used in **security systems, automatic lighting, and smart home applications** due to its low power consumption and high efficiency.



Fig17.1: PIR Sensor

RGB LED (Red-Green-Blue LED)

An **RGB LED** is a type of LED that combines **red, green, and blue** light-emitting diodes in a single package. By adjusting the intensity of each color, it can produce a wide range of colors through **additive color mixing**.

Pin Configuration:

An RGB LED has **four pins**:

- **Common Anode (+) or Common Cathode (-)**
- **Red (R) Pin** – Controls the red LED
- **Green (G) Pin** – Controls the green LED
- **Blue (B) Pin** – Controls the blue LED

Fig17.

For a **Common Anode RGB LED**, the common pin connects to **5V**, and the colors are controlled using **PWM (Pulse Width Modulation)** on the other three pins. For a **Common Cathode RGB LED**, the common pin connects to **GND**, and the colors are controlled by supplying voltage to the color pins.

By varying the **PWM duty cycle** of each color, an **Arduino or microcontroller** can create different colors, making RGB LEDs useful for **indicators, decorative lighting, and visual feedback systems**.

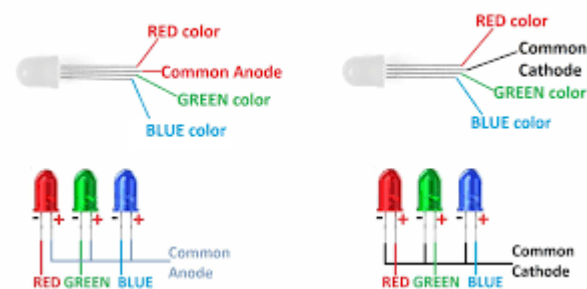


Fig17.2: RGB LED

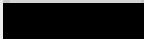













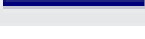
| Color | HTML/CSS NAME | Decimal (R,G,B) |
|---|---------------|-----------------|
|  | Black | (0,0,0) |
|  | White | (255,255,255) |
|  | Red | (255,0,0) |
|  | Lime | (0,255,0) |
|  | Blue | (0,0,255) |
|  | Yellow | (255,255,0) |
|  | Cyan | (0,255,255) |
|  | Magenta | (255,0,255) |
|  | Silver | (192,192,192) |
|  | Grey | (128,128,128) |
|  | Maroon | (128,0,0) |
|  | Olive | (128,128,0) |
|  | Green | (0,128,0) |
|  | Purple | (128,0,128) |
|  | Teal | (0,128,128) |
|  | Navy | (0,0,128) |

Fig 17.3: Color Char list for RGB LED

Buzzer

A **buzzer** is an electronic component that produces sound when an electric current is applied. It can be **active** (self-oscillating) or **passive** (controlled via PWM signals).

The buzzer has **two main pins**:

- **Positive (+) Pin** – Connects to a digital output pin of the microcontroller
- **Negative (-) Pin** – Connects to GND

In Arduino projects, a buzzer is controlled using the `tone()` function to generate different frequencies. It is commonly used in **alarm systems, notifications, and sound-based feedback mechanisms** in electronic circuits.



Fig17.4: Buzzer

Project

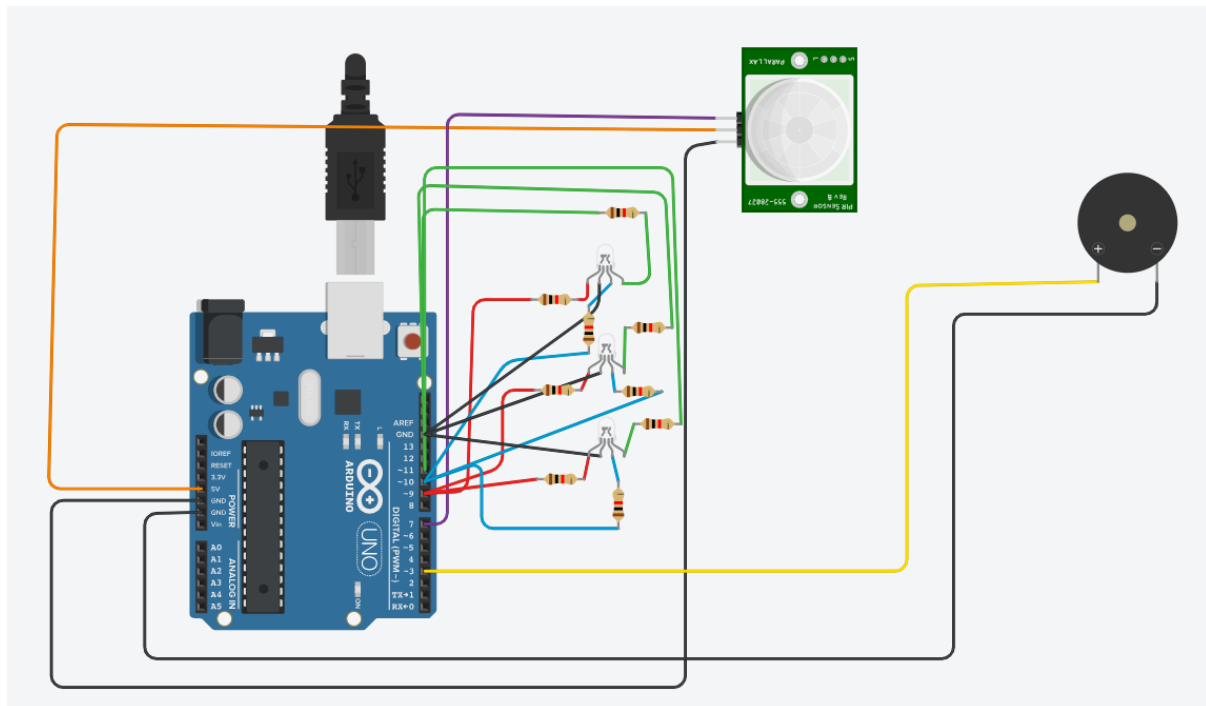


Fig17.4: Motion controlled colour changer using
PIR sensor

Explanation:

Motion-Sensing Color Changer Using Arduino

This project is a motion-sensing color changer that detects movement using a PIR sensor and changes the color of an RGB LED accordingly. Additionally, a buzzer provides an audible alert when motion is detected. The system is controlled by an Arduino Uno, which processes the PIR sensor's signal and adjusts the LED's color output.

Principle of Operation

The PIR sensor detects infrared radiation changes caused by moving objects. When motion is detected, the PIR sensor outputs a HIGH signal to the Arduino, triggering the RGB LED to change color and the buzzer to produce sound. The LED colors are controlled using PWM (Pulse Width Modulation) on the red, green, and blue pins. The system continuously monitors motion and updates the LED colors dynamically.

Pin Connections

- PIR Sensor
 - VCC → 5V (Power supply)
 - GND → GND (Ground)
 - OUT → Digital Pin (e.g., D7) (Motion detection signal to Arduino)

- RGB LED (Common Cathode or Common Anode)
 - Red Pin → Digital PWM Pin (e.g., D9)
 - Green Pin → Digital PWM Pin (e.g., D10)
 - Blue Pin → Digital PWM Pin (e.g., D11)
 - Common Cathode → GND (For common anode, connect to 5V)
- Buzzer
 - Positive (+) → Digital Pin (e.g., D6)
 - Negative (-) → GND

Signal Flow & Working

1. The PIR sensor continuously monitors for motion. When it detects movement, it sends a HIGH signal to the Arduino.
2. The Arduino processes the signal and triggers an LED color change by adjusting PWM values on the red, green, and blue pins.
3. Simultaneously, the buzzer is activated to provide an alert when motion is detected.
4. If no motion is detected, the system remains in an idle state, with the RGB LED turned off or displaying a default color.

This project is useful for motion-activated lighting, security systems, and interactive installations, providing a visual and auditory response to detected movement.