

LAB 5: Smart Color Detection & Control with MIT App

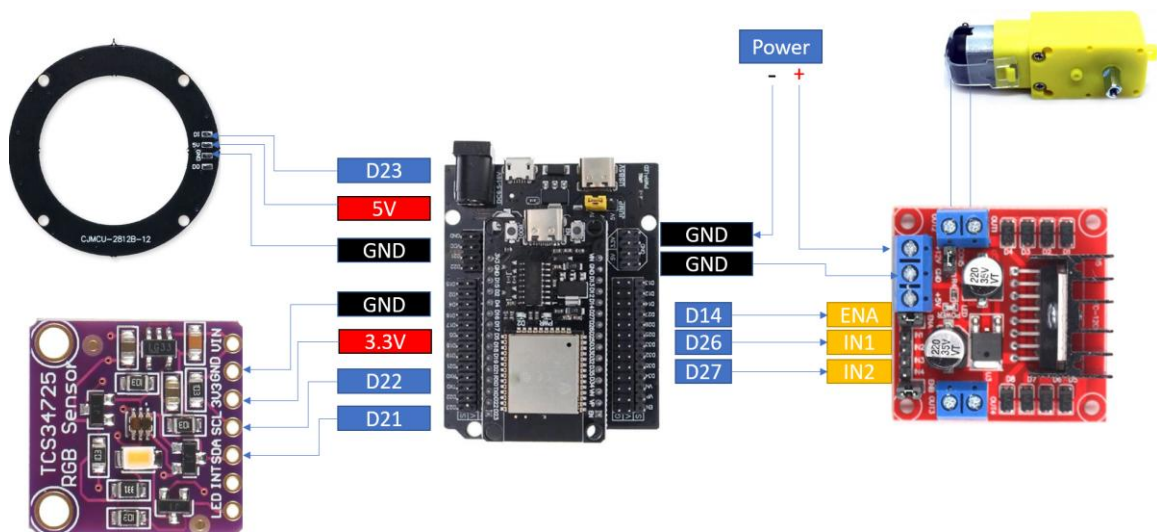
1. Overview

In this lab, students will design and implement a color-based IoT control system using ESP32 and MicroPython (Thonny). The system integrates TCS34725 (color sensor), NeoPixel (RGB LED), and a DC motor. Students must implement edge logic processing to classify colors and control outputs accordingly. The system will also communicate with MIT App Inventor for real-time monitoring and manual control.

2. Learning Outcomes (CLO Alignment)

- Integrate I2C sensor (TCS34725) with ESP32.
- Implement rule-based color classification logic.
- Control NeoPixel LED using RGB values.
- Control DC motor speed using PWM.
- Design a mobile app using MIT App Inventor.
- Implement combined automatic and manual control systems.

3. Connection



4. System Description

The ESP32 continuously reads RGB values from the TCS34725 sensor. Based on the detected color, the system performs the following:

1. Read RGB values from sensor.
2. Classify detected color (RED, GREEN, BLUE).
3. Control NeoPixel color based on classification.
4. Adjust motor speed using PWM.
5. Send detected color to MIT App.
6. Allow manual override from MIT App (motor + NeoPixel).

5. Tasks & Checkpoints

Task 1 - RGB Reading

- Read RGB values from TCS34725.
- Print values to Serial Monitor.

Evidence: Screenshot showing RGB values.

Task 2 - Color Classification

Classification Rules:

- $R > G$ and $R > B \rightarrow$ RED
- $G > R$ and $G > B \rightarrow$ GREEN
- $B > R$ and $B > G \rightarrow$ BLUE

Evidence: Demonstration of correct color detection.

Task 3 - NeoPixel Control

- RED \rightarrow NeoPixel shows Red
- GREEN \rightarrow NeoPixel shows Green
- BLUE \rightarrow NeoPixel shows Blue

Evidence: NeoPixel color change demonstration.

Task 4 - Motor Control (PWM)

- RED → PWM = 700
- GREEN → PWM = 500
- BLUE → PWM = 300

Evidence: Motor speed variation.

Task 5 - MIT App Integration

App Requirements:

- Display detected color (Label).
- Buttons: Forward, Stop, Backward.
- RGB input boxes (R, G, B).
- Button to set NeoPixel color manually.

Evidence: Screenshot of working app.

6. Submission & Academic Integrity

- main.py (MicroPython source code)
- Flowchart (Important)
- MIT App file or screenshots
- Demo video (60–90 seconds)
- README.md explaining system logic

All submitted work must be original. Code sharing is strictly prohibited.