**#include <Wire.h> // Include the Wire library for I2C**

**#include <Adafruit\_TCS34725.h> // Include the Adafruit TCS34725 color sensor library**

**#include <LiquidCrystal\_I2C.h> // Include the LiquidCrystal\_I2C library**

**#include <ESP32Servo.h> // Include the ESP32Servo library**

**#include <WiFi.h> // Include the WiFi library**

**#include <BlynkSimpleEsp32.h> // Include the Blynk library**

**#define BLYNK\_PRINT Serial**

**#define BLYNK\_TEMPLATE\_ID "TMPL6KqL-9GWS"**

**#define BLYNK\_TEMPLATE\_NAME "Final Project Actual Simulation"**

**#define BLYNK\_AUTH\_TOKEN "1s3u3-orayOwc-FxqCOdAdGsm0UOb4\_1"**

**char auth[] = "1s3u3-orayOwc-FxqCOdAdGsm0UOb4\_1";**

**char ssid[] = "PLDTHOMEFIBR233f8";**

**char pass[] = "PLDTWIFIroxasreenfam";**

**Adafruit\_TCS34725 tcs = Adafruit\_TCS34725(TCS34725\_INTEGRATIONTIME\_50MS, TCS34725\_GAIN\_4X);**

**LiquidCrystal\_I2C lcd(0x27, 16, 2); // Change the LCD address and dimensions if needed**

**#define SERVO\_PIN1 16 // Pin number for the base servo**

**#define SERVO\_PIN2 17 // Pin number for the left servo**

**#define SERVO\_PIN3 18 // Pin number for the right servo**

**#define SERVO\_PIN4 19 // Pin number for the end effector servo**

**Servo servo1; // Base servo with angles 90, 180, 0**

**Servo servo2; // Left servo with angles 5, 0**

**Servo servo3; // Right servo with angles 50, 0**

**Servo servo4; // End effector servo with angles 90, 0**

**bool programEnabled = false; // To enable/disable the program**

**void setup() {**

**Serial.begin(115200);**

**tcs.begin();**

**lcd.begin(16, 2); // Initialize the LCD**

**servo1.attach(SERVO\_PIN1); // Base servo**

**servo2.attach(SERVO\_PIN2); // Left servo**

**servo3.attach(SERVO\_PIN3); // Right servo**

**servo4.attach(SERVO\_PIN4); // End effector servo**

**// Turn on the LCD backlight**

**lcd.setBacklight(HIGH); // Change to HIGH for some modules**

**// Connect to WiFi**

**WiFi.begin(ssid, pass);**

**Blynk.begin(auth, ssid, pass);**

**}**

**void loop() {**

**if (programEnabled) {**

**float red, green, blue;**

**tcs.getRGB(&red, &green, &blue);**

**int R = int(red);**

**int G = int(green);**

**int B = int(blue);**

**String color = "";**

**if ((G - B > 10) && (G - R > 30)) { // Color Green**

**color = "UNRIPE";**

**lcd.clear();**

**lcd.setCursor(0, 0);**

**lcd.print("RIPENESS: ");**

**lcd.print(color);**

**// First Movement Sequence**

**servo2.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50); // Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**// Second Movement Sequence**

**servo1.write(90); // Rotate the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo2.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50); // Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo4.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo1.write(0); // Rotate the servo to its initial position**

**delay(1000); // Wait for one second**

**Serial.print("R: "); Serial.print(int(red));**

**Serial.print(" G: "); Serial.print(int(green));**

**Serial.print(" B: "); Serial.print(int(blue));**

**Serial.print(" Color: "); Serial.println(color);**

**}**

**else if ((R - B > 40) && (G - B > 40)) { // Color Yellow**

**color = "RIPE";**

**lcd.clear(); // Clear the LCD display**

**lcd.setCursor(0, 0); // cursor to the first column of the first row**

**lcd.print("RIPENESS: "); // Print "RIPENESS: " on the LCD**

**lcd.print(color); // Print the detected color on the LCD**

**// Third Movement Sequence**

**servo2.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50); // Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**// Fourth Movement Sequence**

**servo1.write(180); // Rotate the servo to 180 degrees**

**delay(1000); // Wait for one second**

**servo2.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50);// Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo4.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo2.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo1.write(0); // Rotate the servo to its initial position**

**delay(1000); // Wait for one second**

**delay(1000); // Wait for one second**

**lcd.clear(); // Clear the LCD display**

**}**

**Blynk.virtualWrite(V1, color); // Displays what the sensor detects.**

**delay(1000); // Wait for one second**

**}**

**Blynk.run();**

**}**

**BLYNK\_WRITE(V2) { // ON/OFF switch for the program.**

**int switchValue = param.asInt();**

**if (switchValue == 1) {**

**programEnabled = true;**

**lcd.clear(); // Clear the LCD display**

**lcd.setCursor(0, 0); // Set the cursor to the first column of the first row**

**lcd.print("System is ON "); // Print "System is ON" on the LCD**

**} else {**

**programEnabled = false;**

**lcd.clear(); // Clear the LCD display**

**lcd.setCursor(0, 0); // Set the cursor to the first column of the first row**

**lcd.print("System is OFF"); // Print "System is Off" on the LCD**

**}**

**}**

**BLYNK\_WRITE(V3) { // Button to control unripe movement**

**if (programEnabled) {**

**int buttonState = param.asInt();**

**if (buttonState == HIGH) {**

**// Perform unripe movement**

**servo2.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50); // Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo1.write(90); // Rotate the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo2.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50); // Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo4.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo1.write(0); // Rotate the servo to its initial position 0 degrees**

**delay(1000); // Wait for one second**

**lcd.clear(); // Clear the LCD display**

**}**

**}**

**}**

**BLYNK\_WRITE(V4) { // Button to control ripe movement**

**if (programEnabled) {**

**int buttonState = param.asInt();**

**if (buttonState == HIGH) {**

**// Perform ripe movement**

**servo2.write(5);// Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50); // Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo1.write(180); // Rotate the servo to 180 degrees**

**delay(1000); // Wait for one second**

**servo2.write(5); // Move the servo to 5 degrees**

**delay(1000); // Wait for one second**

**servo3.write(50); // Move the servo to 50 degrees**

**delay(1000); // Wait for one second**

**servo4.write(90); // Move the servo to 90 degrees**

**delay(1000); // Wait for one second**

**servo4.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo3.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo2.write(0); // Move the servo to 0 degrees**

**delay(1000); // Wait for one second**

**servo1.write(0); // Rotate the servo to its initial position 0 degrees**

**delay(1000); // Wait for one second**

**lcd.clear(); // Clear the LCD display**

**}**

**}**

**}**