**CODE BEHIND THE SMART TRAFFIC SYSTEM**

// Lane 1 Traffic Lights

#define RED1 13

#define YELLOW1 12

#define GREEN1 8

// Lane 1 Ultrasonic Sensor

#define TRIG1 5

#define ECHO1 6

// Lane 2 Traffic Lights

#define RED2 7

#define YELLOW2 4

#define GREEN2 2

// Lane 2 Ultrasonic Sensor

#define TRIG2 9

#define ECHO2 10

void setup() {

// Set up pins for LEDs

pinMode(RED1, OUTPUT);

pinMode(YELLOW1, OUTPUT);

pinMode(GREEN1, OUTPUT);

pinMode(RED2, OUTPUT);

pinMode(YELLOW2, OUTPUT);

pinMode(GREEN2, OUTPUT);

// Set up pins for ultrasonic sensors

pinMode(TRIG1, OUTPUT);

pinMode(ECHO1, INPUT);

pinMode(TRIG2, OUTPUT);

pinMode(ECHO2, INPUT);

Serial.begin(9600);

}

// Function to measure distance using ultrasonic sensor

long getDistance(int trigPin, int echoPin) {

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

long duration = pulseIn(echoPin, HIGH);

long distance = duration \* 0.034 / 2; // Convert to cm

return distance;

}

// Function to handle lane switching with correct delays

void switchLane(int red1, int yellow1, int green1, int red2, int yellow2, int green2) {

// Step 1: Red Light ON for 7 sec before switching lanes

digitalWrite(red1, HIGH);

digitalWrite(green1, LOW);

digitalWrite(yellow1, LOW);

digitalWrite(red2, HIGH);

digitalWrite(green2, LOW);

digitalWrite(yellow2, LOW);

delay(7000); // Red delay

// Step 2: Yellow Light for 2 sec

digitalWrite(red1, LOW);

digitalWrite(yellow1, HIGH);

delay(2000); // Yellow delay

// Step 3: Green Light for 8 sec

digitalWrite(yellow1, LOW);

digitalWrite(green1, HIGH);

delay(8000); // Green delay

// Step 4: Turn off Green, Prepare to switch

digitalWrite(green1, LOW);

digitalWrite(yellow1, HIGH);

delay(2000); // Yellow before switching

// Switch to next lane

digitalWrite(yellow1, LOW);

digitalWrite(red1, HIGH);

digitalWrite(red2, LOW);

digitalWrite(yellow2, HIGH);

delay(2000); // Yellow transition

digitalWrite(yellow2, LOW);

digitalWrite(green2, HIGH);

delay(8000); // Green delay for second lane

digitalWrite(green2, LOW);

digitalWrite(yellow2, HIGH);

delay(2000); // Yellow before switching back

digitalWrite(yellow2, LOW);

digitalWrite(red2, HIGH);

}

// Main loop to check traffic and control lights

void loop() {

long distance1 = getDistance(TRIG1, ECHO1);

long distance2 = getDistance(TRIG2, ECHO2);

Serial.print("Lane 1 Distance: ");

Serial.print(distance1);

Serial.print(" cm | Lane 2 Distance: ");

Serial.println(distance2);

if (distance1 < 20 && distance2 >= 20) {

// Vehicles detected in Lane 1, but not in Lane 2

switchLane(RED1, YELLOW1, GREEN1, RED2, YELLOW2, GREEN2);

} else if (distance2 < 20 && distance1 >= 20) {

// Vehicles detected in Lane 2, but not in Lane 1

switchLane(RED2, YELLOW2, GREEN2, RED1, YELLOW1, GREEN1);

} else if (distance1 < 20 && distance2 < 20) {

// Vehicles detected in both lanes, follow normal switching

switchLane(RED1, YELLOW1, GREEN1, RED2, YELLOW2, GREEN2);

} else {

// No vehicles detected, default cycle

switchLane(RED1, YELLOW1, GREEN1, RED2, YELLOW2, GREEN2);

}

delay(500); // Small delay before next detection