#include <ESP8266WiFi.h>

#include <ESP8266WebServer.h>

// Motor driver pins

#define M1\_IN1 D2

#define M1\_IN2 D3

#define M1\_IN3 D4

#define M1\_IN4 D5

#define CUTTING\_MOTOR D8

// Wi-Fi credentials (Access Point Mode)

const char\* ssid = "ESP8266\_Grass\_Car";

const char\* password = "12345678";

// Create server on port 80

ESP8266WebServer server(80);

// Function to stop all motors

void stopMotors() {

digitalWrite(M1\_IN1, LOW);

digitalWrite(M1\_IN2, LOW);

digitalWrite(M1\_IN3, LOW);

digitalWrite(M1\_IN4, LOW);

}

// Function to move forward

void moveForward() {

digitalWrite(M1\_IN1, HIGH);

digitalWrite(M1\_IN2, LOW);

digitalWrite(M1\_IN3, HIGH);

digitalWrite(M1\_IN4, LOW);

}

// Function to move backward

void moveBackward() {

digitalWrite(M1\_IN1, LOW);

digitalWrite(M1\_IN2, HIGH);

digitalWrite(M1\_IN3, LOW);

digitalWrite(M1\_IN4, HIGH);

}

// Function to turn left

void turnLeft() {

digitalWrite(M1\_IN1, LOW);

digitalWrite(M1\_IN2, HIGH);

digitalWrite(M1\_IN3, HIGH);

digitalWrite(M1\_IN4, LOW);

}

// Function to turn right

void turnRight() {

digitalWrite(M1\_IN1, HIGH);

digitalWrite(M1\_IN2, LOW);

digitalWrite(M1\_IN3, LOW);

digitalWrite(M1\_IN4, HIGH);

}

// Function to start the cutting motor

void startCutting() {

digitalWrite(CUTTING\_MOTOR, HIGH);

}

// Function to stop the cutting motor

void stopCutting() {

digitalWrite(CUTTING\_MOTOR, LOW);

}

// Handle commands from the app

void handleCommands() {

String command = server.uri(); // Get the command sent via the app

if (command == "/F") moveForward(); // Forward

else if (command == "/B") moveBackward(); // Backward

else if (command == "/L") turnLeft(); // Turn Left

else if (command == "/R") turnRight(); // Turn Right

else if (command == "/S") stopMotors(); // Stop

else if (command == "/W") startCutting(); // Cutting Motor ON (Light ON)

else if (command == "/w") stopCutting(); // Cutting Motor OFF (Light OFF)

server.send(200, "text/plain", "OK"); // Send response to the app

}

void setup() {

// Set motor pins as output

pinMode(M1\_IN1, OUTPUT);

pinMode(M1\_IN2, OUTPUT);

pinMode(M1\_IN3, OUTPUT);

pinMode(M1\_IN4, OUTPUT);

pinMode(CUTTING\_MOTOR, OUTPUT);

// Stop all motors initially

stopMotors();

stopCutting();

// Start Wi-Fi in Access Point mode

WiFi.softAP(ssid, password);

// Start the server

server.onNotFound(handleCommands); // Handle all incoming requests

server.begin();

}

void loop() {

server.handleClient(); // Handle incoming client requests

}