#include <ESP8266WiFi.h>

#include <LoRa.h>

// Wi-Fi Credentials

const char\* ssid = "Your\_SSID";

const char\* password = "Your\_PASSWORD";

// LoRa Parameters

#define SS 5 // GPIO5 - LoRa NSS

#define RST 14 // GPIO14 - LoRa RESET

#define DIO0 2 // GPIO2 - LoRa IRQ

void setup() {

// Start Serial Communication

Serial.begin(9600);

// Initialize Wi-Fi

WiFi.begin(ssid, password);

Serial.print("Connecting to Wi-Fi");

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

Serial.print(".");

}

Serial.println("\nConnected to Wi-Fi");

// Initialize LoRa

LoRa.setPins(SS, RST, DIO0);

if (!LoRa.begin(868E6)) {

Serial.println("Starting LoRa failed!");

while (1);

}

Serial.println("LoRa initialized.");

}

void loop() {

// Check if data is available from MSP430

if (Serial.available()) {

String data = Serial.readStringUntil('\n');

// Debugging on serial monitor

Serial.println("Received from MSP430: " + data);

// Transmit via LoRa

LoRa.beginPacket();

LoRa.print(data);

LoRa.endPacket();

// Send data to a cloud platform (e.g., via HTTP POST)

if (WiFi.status() == WL\_CONNECTED) {

WiFiClient client;

if (client.connect("api.yourcloudservice.com", 80)) {

String postRequest = "POST /data HTTP/1.1\r\n";

postRequest += "Host: api.yourcloudservice.com\r\n";

postRequest += "Content-Type: application/json\r\n";

postRequest += "Content-Length: " + String(data.length()) + "\r\n\r\n";

postRequest += data;

client.print(postRequest);

Serial.println("Data sent to cloud: " + data);

client.stop();

}

}

delay(2000); // Wait for next transmission

}

}