**Documentation for ESP8266 Code:**

cppCopy code

#include <WiFi.h> #include <Adafruit\_MQTT.h> #include <Adafruit\_MQTT\_Client.h> #include <HardwareSerial.h>

**Libraries Used:**

1. **WiFi.h**: Enables the ESP8266 to connect to a Wi-Fi network.
2. **Adafruit\_MQTT.h**: Provides MQTT (Message Queuing Telemetry Transport) functionality for communicating with the Adafruit IO platform.
3. **Adafruit\_MQTT\_Client.h**: Implements the MQTT client for connecting to the Adafruit IO MQTT server.
4. **HardwareSerial.h**: Allows communication with other devices using hardware serial communication.

**Wi-Fi and Adafruit IO Credentials:**

cppCopy code

#define WLAN\_SSID "csmlair" #define WLAN\_PASS "12345678" #define AIO\_SERVER "io.adafruit.com" #define AIO\_SERVERPORT 1883 #define AIO\_USERNAME "CSML" #define AIO\_KEY "aio\_MxMS92chbYpYEcn3tdEbKQoMKlfp"

**Hardware Serial Configuration:**

cppCopy code

HardwareSerial espSerial(2); // Connect ESP32's RX2 to pin 16, and TX2 to pin 17

**Wi-Fi Client and MQTT Client Initialization:**

cppCopy code

WiFiClient client; Adafruit\_MQTT\_Client mqtt(&client, AIO\_SERVER, AIO\_SERVERPORT, AIO\_USERNAME, AIO\_KEY);

**Adafruit IO Feed Initialization:**

cppCopy code

Adafruit\_MQTT\_Publish feed = Adafruit\_MQTT\_Publish(&mqtt, AIO\_USERNAME "/feeds/dust");

**Setup Function:**

cppCopy code

void setup() { Serial.begin(9600); espSerial.begin(9600, SERIAL\_8N1, 16, 17); // Begin HardwareSerial communication with Arduino connect(); }

**Connect Function:**

cppCopy code

void connect() { // Connect to Wi-Fi // Connect to Adafruit IO MQTT server }

**Loop Function:**

cppCopy code

void loop() { // Check for data from the dust sensor // Publish the data to Adafruit IO // Add a delay to prevent rapid publishing }

**Documentation for Arduino Code:**

cppCopy code

#include <SD\_ZH03B.h> #include <SoftwareSerial.h>

**Libraries Used:**

1. **SD\_ZH03B.h**: Library for interfacing with the ZH03B dust sensor. Link: [SD\_ZH03B Library](https://github.com/ShaggyDog18/SD_ZH03B)
2. **SoftwareSerial.h**: Allows communication with other devices using software serial communication.

**Software Serial Configuration:**

cppCopy code

SoftwareSerial ZHSerial(4, 5); // RX, TX SoftwareSerial espSerial(6, 7); // Connect ESP8266's TX to pin 6, and RX to pin 7

**ZH03B Dust Sensor Initialization:**

cppCopy code

SD\_ZH03B ZH03B(ZHSerial, SD\_ZH03B::SENSOR\_ZH03B);

**Analog Pin for NO2 Sensor:**

cppCopy code

const int no2SensorPin = A0;

**Setup Function:**

cppCopy code

void setup() { Serial.begin(9600); espSerial.begin(9600); // Begin SoftwareSerial communication with ESP8266 delay(1000); ZHSerial.begin(9600); delay(100); ZH03B.setInitiativeMode(); // IU mode delay(200); }

**Sending Sensor Data to ESP Function:**

cppCopy code

void sendSensorDataToESP(int pm1, int pm2\_5, int pm10, int no2Value) { // Convert sensor data to a string and send it to the ESP8266 }

**Reading Sensor Data Function:**

cppCopy code

void readSensorData() { // Read data from the ZH03B dust sensor // Read data from the NO2 sensor // Send sensor data to the ESP8266 }

**Loop Function:**

cppCopy code

void loop() { // Repeat reading and sending sensor data for 10 cycles // Add a delay between cycles }

**Note:** Ensure that the connections between the ESP8266 and Arduino are made as described in the code comments. Connect ESP8266's TX to Arduino's RX, and ESP8266's RX to Arduino's TX. Additionally, connect the respective power and ground pins between the two devices. The specific pins used for communication are specified in the code.

Top of Form