Code with web send

#include <Arduino.h>

#include <Wire.h>

#include <Adafruit\_Sensor.h>

#include <Adafruit\_BMP280.h>

#define ledPin 4 // LED pin

String incomingNum = “”; // Variable to store the sender’s number

Adafruit\_BMP280 bmp; // I2C

Void setup() {

pinMode(ledPin, OUTPUT);

Serial2.begin(9600, SERIAL\_8N1, 16, 17); // Using Serial2 for GSM module on GPIO 16 (RX) and 17 (TX)

Serial.begin(115200); // USB Serial Monitor

Wire.begin(21, 22); // SDA on GPIO 21, SCL on GPIO 22

If (!bmp.begin(0x76)) { // Address of the BMP280 sensor

Serial.println(“Could not find a valid BMP280 sensor, check wiring!”);

While (1);

}

// Read and print initial sensor values

printSensorValues();

delay(1000);

Serial2.println(“AT+CMGF=1”);

Delay(1000);

Serial2.println(“AT+CNMI=1,2,0,0,0”); // Configure module to send SMS data directly to the serial

Delay(1000);

}

Void loop() {

If (Serial2.available()) {

String smsText = Serial2.readString();

Serial.println(“Received SMS: “ + smsText); // Debug: print the whole SMS data

Int indexNumStart = smsText.indexOf(“+CMT: \””) + 7;

Int indexNumEnd = smsText.indexOf(‘\”’, indexNumStart);

incomingNum = smsText.substring(indexNumStart, indexNumEnd);

Serial.println(“Sender Number: “ + incomingNum); // Debug: print the sender’s number

If (smsText.indexOf(“#1”) != -1) {

String sensorValues = readSensorValues();

sendSMS(sensorValues, incomingNum);

} else if (smsText.indexOf(“#4”) != -1) {

toggleLED();

sendSMS(“LED toggled.”, incomingNum);

} else if (smsText.indexOf(“#5”) != -1) {

pingAndRespond();

} else if (smsText.indexOf(“#6”) != -1) {

sendDataToWebsite();

sendSMS(“Data sent to website.”, incomingNum);

}

}

}

Void sendSMS(String message, String num) {

Serial2.print(“AT+CMGS=\””);

Serial2.print(num);

Serial2.println(“\””);

Delay(1000);

Serial2.println(message);

Serial2.write(26); // ASCII code for Ctrl-Z

Delay(5000);

}

Void toggleLED() {

digitalWrite(ledPin, HIGH);

delay(5000);

digitalWrite(ledPin, LOW);

}

String readSensorValues() {

Float temperature = bmp.readTemperature();

Float pressure = bmp.readPressure();

Float altitude = bmp.readAltitude(1013.25); // Adjust the sea level pressure as necessary

String message = “Temp: “ + String(temperature) + “ C, “;

Message += “Pressure: “ + String(pressure) + “ Pa, “;

Message += “Altitude: “ + String(altitude) + “ m”;

Return message;

}

Void printSensorValues() {

Float temperature = bmp.readTemperature();

Float pressure = bmp.readPressure();

Float altitude = bmp.readAltitude(1013.25); // Adjust the sea level pressure as necessary

Serial.print(“Initial Sensor Readings -> “);

Serial.print(“Temperature: “);

Serial.print(temperature);

Serial.print(“ °C, Pressure: “);

Serial.print(pressure);

Serial.print(“ Pa, Altitude: “);

Serial.print(altitude);

Serial.println(“ m”);

}

Void setupGPRS() {

// Set bearer profile for GPRS

Serial2.println(“AT+SAPBR=3,1,\”Contype\”,\”GPRS\””);

Delay(2000);

printSerial2Response();

// Set APN

Serial2.println(“AT+SAPBR=3,1,\”APN\”,\”live.vodafone.com\””);

Delay(2000);

printSerial2Response();

// Activate bearer profile

Serial2.println(“AT+SAPBR=1,1”);

Delay(5000);

printSerial2Response();

// Query bearer profile IP address

Serial2.println(“AT+SAPBR=2,1”);

Delay(2000);

printSerial2Response();

// Start Task and Set APN, USER NAME, PASSWORD

Serial2.println(“AT+CSTT=\”live.vodafone.com\”,\”\”,\”\””);

Delay(2000);

printSerial2Response();

// Bring up wireless connection with GPRS or CSD

Serial2.println(“AT+CIICR”);

Delay(5000);

printSerial2Response();

// Get local IP address

Serial2.println(“AT+CIFSR”);

Delay(3000);

printSerial2Response();

}

Void printSerial2Response() {

// Read and print the response from Serial2

While (Serial2.available()) {

String response = Serial2.readString();

Serial.println(response);

}

}

Void pingAndRespond() {

setupGPRS(); // Setup GPRS before pinging

String pingResult = pingIP(“1.1.1.1”);

sendSMS(pingResult, incomingNum);

}

String pingIP(const char\* ip) {

Serial2.println(“AT+CIPPING=\”1.1.1.1\””); // Ping command

Delay(5000); // Wait for ping response

String response = “”;

While (Serial2.available()) {

Response += Serial2.readStringUntil(‘\n’);

}

If (response.indexOf(“+CIPPING: 1”) != -1) {

Return “Successful ping to “ + String(ip);

} else {

Return “Failed ping attempt to “ + String(ip);

}

}

Void sendDataToWebsite() {

setupGPRS(); // Setup GPRS before sending data

// Prepare the data to send

String sensorData = readSensorValues();

// Create the HTTP POST request

String httpsRequest = “AT+HTTPPARA=\”URL\”,\<http://iotgsm-dm.com/Data.html\>”;

Serial2.println(httpsRequest);

Delay(2000);

printSerial2Response();

// Initialize HTTP service

Serial2.println(“AT+HTTPINIT”);

Delay(2000);

printSerial2Response();

// Set the content type

Serial2.println(“AT+HTTPPARA=\”CONTENT\”,\”application/x-www-form-urlencoded\””);

Delay(2000);

printSerial2Response();

// Set the data to send

String httpData = “data=” + sensorData;

Serial2.println(“AT+HTTPDATA=” + String(httpData.length()) + “,10000”);

Delay(2000);

Serial2.println(httpData);

Delay(2000);

printSerial2Response();

// Execute the POST request

Serial2.println(“AT+HTTPACTION=1”);

Delay(5000);

printSerial2Response();

// Read the HTTP response

Serial2.println(“AT+HTTPREAD”);

Delay(2000);

printSerial2Response();

// Terminate the HTTP service

Serial2.println(“AT+HTTPTERM”);

Delay(2000);

printSerial2Response();

}