/\* Arduino+esp8266 thingSpeak example

\* Example name = "Write duration and distance to Thingspeak channel"

\*

\* Download latest Virtuino android app from the link:https://play.google.com/store/apps/details?id=com.virtuino\_automations.virtuino&hl

\* Video tutorial link: https://youtu.be/4XEe0HY0j6k

\*/

// Code to use SoftwareSerial

#include <SoftwareSerial.h>

SoftwareSerial espSerial = SoftwareSerial(2,3); // arduino RX pin=2 arduino TX pin=3 connect the arduino RX pin to esp8266 module TX pin - connect the arduino TX pin to esp8266 module RX pin

const int trigPin = 4;

const int echoPin = 5;

// defines variables

long duration;

int distance;

String apiKey = "DK59X5DLS04TVFG5"; // replace with your channel's thingspeak WRITE API key

String ssid="WIFI NAME"; // Wifi network SSID

String password ="WIFI PASSWORD"; // Wifi network password

boolean DEBUG=true;

//======================================================================== showResponce

void showResponse(int waitTime){

long t=millis();

char c;

while (t+waitTime>millis()){

if (espSerial.available()){

c=espSerial.read();

if (DEBUG) Serial.print(c);

}

}

}

//========================================================================

boolean thingSpeakWrite(long value1, int value2){

String cmd = "AT+CIPSTART=\"TCP\",\""; // TCP connection

cmd += "184.106.153.149"; // api.thingspeak.com

cmd += "\",80";

espSerial.println(cmd);

if (DEBUG) Serial.println(cmd);

if(espSerial.find("Error")){

if (DEBUG) Serial.println("AT+CIPSTART error");

return false;

}

String getStr = "GET /update?api\_key="; // prepare GET string

getStr += apiKey;

getStr +="&field1=";

getStr += String(value1);

getStr +="&field2=";

getStr += String(value2);

// getStr +="&field3=";

// getStr += String(value3);

// ...

getStr += "\r\n\r\n";

// send data length

cmd = "AT+CIPSEND=";

cmd += String(getStr.length());

espSerial.println(cmd);

if (DEBUG) Serial.println(cmd);

delay(100);

if(espSerial.find(">")){

espSerial.print(getStr);

if (DEBUG) Serial.print(getStr);

}

else{

espSerial.println("AT+CIPCLOSE");

// alert user

if (DEBUG) Serial.println("AT+CIPCLOSE");

return false;

}

return true;

}

//================================================================================ setup

void setup() {

DEBUG=true; // enable debug serial

Serial.begin(9600);

pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

espSerial.begin(115200); // enable software serial

// Your esp8266 module's speed is probably at 115200.

// For this reason the first time set the speed to 115200 or to your esp8266 configured speed

// and upload. Then change to 9600 and upload again

//espSerial.println("AT+RST"); // Enable this line to reset the module;

//showResponse(1000);

//espSerial.println("AT+UART\_CUR=9600,8,1,0,0"); // Enable this line to set esp8266 serial speed to 9600 bps

//showResponse(1000);

espSerial.println("AT+CWMODE=1"); // set esp8266 as client

showResponse(1000);

espSerial.println("AT+CWJAP=\""+ssid+"\",\""+password+"\""); // set your home router SSID and password

showResponse(5000);

if (DEBUG) Serial.println("Setup completed");

}

// ====================================================================== loop

void loop() {

// Read sensor values

digitalWrite(trigPin, LOW);

delay(1000);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delay(1000);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculating the distance

distance= duration\*0.034/2; //s=d/t

// Prints the distance on the Serial Monitor

Serial.print("Distance(cm): ");

Serial.println(distance);

thingSpeakWrite(duration,distance); // Write values to thingspeak

// thingspeak needs 15 sec delay between updates,

delay(20000);

}