#include <FreqMeasure.h>

float spd ;

#include <SoftwareSerial.h>

#define RX 2

#define TX 3

int a,b,c,d;

String AP = "vihanga";       // AP NAME

String PASS = "12345678"; // AP PASSWORD

String API = "MQZ7Q0ARXWXDE4KJ";   // Write API KEY

String HOST = "[api.thingspeak.com](http://api.thingspeak.com/)";

String PORT = "80";

int countTrueCommand;

int countTimeCommand;

boolean found = false;

int valSensor = 1;

  SoftwareSerial esp8266(RX,TX);

void setup() {

  Serial.begin(57600);

  FreqMeasure.begin();

  pinMode(13,OUTPUT);

  digitalWrite(13,0);

   pinMode(12,INPUT);

  digitalWrite(12,1);

  esp8266.begin(115200);

  sendCommand("AT",5,"OK");

  sendCommand("AT+CWMODE=1",5,"OK");

  sendCommand("AT+CWJAP=\""+ AP +"\",\""+ PASS +"\"",20,"OK");

pinMode(A0,INPUT);

}

double sum=0;

int count=0;

void loop() {

    String getData = "GET /update?api\_key="+ API +"&field1="+noiseValue();

 sendCommand("AT+CIPMUX=1",5,"OK");

 sendCommand("AT+CIPSTART=0,\"TCP\",\""+ HOST +"\","+ PORT,15,"OK");

 sendCommand("AT+CIPSEND=0," +String(getData.length()+4),4,">");

 esp8266.println(getData);delay(1500);countTrueCommand++;

 sendCommand("AT+CIPCLOSE=0",5,"OK");

  if (FreqMeasure.available()) {

    sum = sum + FreqMeasure.read();

    count = count + 1;

    if (count > 60) {

      float frequency = FreqMeasure.countToFrequency(sum / count);

       spd = frequency / 19.49;

      delay(50);

     // Serial.println("frequency");

      //Serial.println(frequency);

      //Serial.println("SPEED");

      //Serial.println(spd);

       sum = 0;

      count = 0;

       }

}

Serial.println(spd);

       if(spd >= 5.68)

  {

    digitalWrite(13,1);

    digitalWrite(12,0);

    Serial.print("a");

  }

  if(spd <= 5.68)

  {

    digitalWrite(13,0);

       digitalWrite(12,1);

     Serial.print("b");

  }

    }

String noiseValue(){

  a=analogRead(A0);

  c= map(a, 0,1023,0,100);

//Serial.println(c);

    return String(c);

}

void sendCommand(String command, int maxTime, char readReplay[]) {

  Serial.print(countTrueCommand);

  Serial.print(". at command => ");

  Serial.print(command);

  Serial.print(" ");

  while(countTimeCommand < (maxTime\*1))

  {

    esp8266.println(command);//at+cipsend

    if(esp8266.find(readReplay))//ok

    {

      found = true;

      break;

    }

    countTimeCommand++;

  }

   if(found == true)

  {

    Serial.println("OYI");

    countTrueCommand++;

    countTimeCommand = 0;

  }

  if(found == false)

  {

    Serial.println("Fail");

    countTrueCommand = 0;

    countTimeCommand = 0;

  }

  found = false;

 }