#include <SoftwareSerial.h>

#define temp A0

#define hum A1

#define gas A2

#define buz A3

char str[70];

String gpsString = "";

char \*test = "$GPGGA";

String latitude = "No Range ";

String longitude = "No Range ";

int i;

boolean gps\_status = 0;

#include <LiquidCrystal.h>

const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

SoftwareSerial gps(10, 13); // RX, TX

SoftwareSerial iot(7, 8); // RX, TX

void setup() {

Serial.begin(9600);

iot.begin(9600);

gps.begin(9600);

lcd.begin(16, 2);

Serial.println("CLEAR DATA");

Serial.println("LABEL,Computer Time,TEMP,HUM,GAS");

pinMode(temp, INPUT);

pinMode(hum, INPUT);

pinMode(gas, INPUT);

pinMode(buz, OUTPUT);

lcd.clear();

lcd.setCursor(1, 0);

lcd.print("ENVIRONMENT");

lcd.setCursor(3, 1);

lcd.print("MONITOR");

delay(1000);

lcd.clear();

digitalWrite(buz, LOW);

}

void loop() {

float voltage = analogRead(temp);

voltage = voltage \* 5.0;

voltage / 1024.0;

float temperatureC = (voltage \* 50 / 1024) ;

//sensorValue = (sensorValue \* 500) / 1023;

float volt = temperatureC + 10;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("TEMP:");

lcd.setCursor(6, 0);

lcd.print(volt);

delay(2000);

int humval = analogRead(hum);

float humvalue = humval;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("HUM:");

lcd.setCursor(6, 0);

lcd.print(humvalue);

delay(2000);

int gasval = analogRead(gas);

int gasvalue = gasval;

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("GAS");

lcd.setCursor(6, 0);

lcd.print(gasvalue);

delay(1000);

get\_gps();

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("LATI ");

lcd.print(latitude);

lcd.setCursor(0, 1);

lcd.print("LONG ");

lcd.print(longitude);

delay(1000);

Serial.print("DATA,TIME,");

Serial.print(volt);

Serial.print(",");

Serial.print(humvalue);

Serial.print(",");

Serial.println(gasvalue);

delay(2000);

iot.print("\*");delay(1000);

iot.print("L");delay(1000);

iot.print("T");delay(1000);

iot.print(" ");delay(1000);

iot.print("L");delay(1000);

iot.print("G");delay(1000);

iot.print(latitude);delay(1000);

iot.print("#");delay(1000);

iot.print("\*");delay(1000);

//iot.print("L");delay(1000);

//iot.print("G");delay(1000);

//iot.print("N");delay(1000);

//iot.print(" ");delay(1000);

iot.print(longitude);delay(1000);

iot.print("#");delay(1000);

if (volt > 40)

{

digitalWrite(buz, HIGH);

delay(2000);

digitalWrite(buz, LOW);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("TEMP HIGH"); delay(1000);

iot.print("\*"); delay(1000);

iot.print("T"); delay(1000);

iot.print("E"); delay(1000);

iot.print("M"); delay(1000);

iot.print("P"); delay(1000);

iot.print(" "); delay(1000);

iot.print("H"); delay(1000);

iot.print("I"); delay(1000);

iot.print("G"); delay(1000);

iot.print("H"); delay(1000);

iot.print(" "); delay(1000);

iot.print(volt);delay(1000);

iot.print("#"); delay(1000);

}

else if (volt < 40)

{

digitalWrite(buz, LOW);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("TEMP NORMAL"); delay(1000);

iot.print("\*"); delay(1000);

iot.print("T"); delay(1000);

iot.print("E"); delay(1000);

iot.print("M"); delay(1000);

iot.print("P"); delay(1000);

iot.print(" "); delay(1000);

iot.print("N"); delay(1000);

iot.print("R"); delay(1000);

iot.print("M"); delay(1000);

iot.print("L"); delay(1000);

iot.print(" "); delay(1000);

iot.print(volt);delay(1000);

iot.print("#"); delay(1000);

}

if (humvalue > 200)

{

digitalWrite(buz, HIGH);

delay(2000);

digitalWrite(buz, LOW);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("HUM LOW"); delay(1000);

iot.print("\*"); delay(1000);

iot.print("H"); delay(1000);

iot.print("U"); delay(1000);

iot.print("M"); delay(1000);

iot.print(" "); delay(1000);

iot.print("L"); delay(1000);

iot.print("O"); delay(1000);

iot.print("W"); delay(1000);

iot.print(" "); delay(1000);

iot.print(humvalue); delay(1000);

iot.print("#"); delay(1000);

}

else if (humvalue <200)

{

digitalWrite(buz, LOW);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("HUM NORMAL"); delay(1000);

iot.print("\*"); delay(1000);

iot.print("H"); delay(1000);

iot.print("U"); delay(1000);

iot.print("M"); delay(1000);

iot.print(" "); delay(1000);

iot.print("N"); delay(1000);

iot.print("R"); delay(1000);

iot.print("M"); delay(1000);

iot.print("L"); delay(1000);

iot.print(" "); delay(1000);

iot.print(humvalue); delay(1000);

iot.print("#"); delay(1000);

}

if (gasvalue < 900)

{

digitalWrite(buz, HIGH);

delay(2000);

digitalWrite(buz, LOW);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("GAS DETECTED"); delay(1000);

iot.print("\*"); delay(1000);

iot.print("G"); delay(1000);

iot.print("A"); delay(1000);

iot.print("S"); delay(1000);

iot.print(" "); delay(1000);

iot.print("D"); delay(1000);

iot.print("E"); delay(1000);

iot.print("T"); delay(1000);

iot.print("E"); delay(1000);

iot.print("C"); delay(1000);

iot.print("T"); delay(1000);

iot.print(" "); delay(1000);

iot.print(gasvalue); delay(1000);

iot.print("#"); delay(1000);

}

if (gasvalue >900)

{

digitalWrite(buz, LOW);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("NO GAS"); delay(1000);

iot.print("\*"); delay(1000);

iot.print("N"); delay(1000);

iot.print("O"); delay(1000);

iot.print(""); delay(1000);

iot.print("G"); delay(1000);

iot.print("A"); delay(1000);

iot.print("S"); delay(1000);

iot.print(" "); delay(1000);

iot.print(gasvalue); delay(1000);

iot.print("#"); delay(1000);

}

}

void gpsEvent()

{

gpsString = "";

while (1)

{

while (gps.available() > 0)

{

char inChar = (char)gps.read();

gpsString += inChar;

i++;

if (i < 7)

{

if (gpsString[i - 1] != test[i - 1])

{

i = 0;

gpsString = "";

}

}

if (inChar == '\r')

{

if (i > 65)

{

gps\_status = 1;

break;

}

else

{

i = 0;

}

}

}

if (gps\_status)

break;

}

}

void get\_gps()

{

gps\_status = 0;

int x = 0;

while (gps\_status == 0)

{

gpsEvent();

int str\_lenth = i;

latitude = "";

longitude = "";

int comma = 0;

while (x < str\_lenth)

{

if (gpsString[x] == ',')

comma++;

if (comma == 2)

latitude += gpsString[x + 1];

else if (comma == 4)

longitude += gpsString[x + 1];

x++;

}

int l1 = latitude.length();

latitude[l1 - 1] = ' ';

l1 = longitude.length();

longitude[l1 - 1] = ' ';

i = 0; x = 0;

str\_lenth = 0;

delay(2000);

}

}