

```

////////////////////////////////////
// RemoteXY include library      //
////////////////////////////////////

// you can enable debug logging to Serial at 115200
// #define REMOTEXY__DEBUGLOG

// RemoteXY select connection mode and include library
#define REMOTEXY_MODE__ESP8266_HARDSERIAL_CLOUD

// RemoteXY connection settings
#define REMOTEXY_SERIAL Serial
#define REMOTEXY_SERIAL_SPEED 115200
#define REMOTEXY_WIFI_SSID "project"
#define REMOTEXY_WIFI_PASSWORD "12345678"
#define REMOTEXY_CLOUD_SERVER "cloud.remotexy.com"
#define REMOTEXY_CLOUD_PORT 6376
#define REMOTEXY_CLOUD_TOKEN
"b4ff66f8abbb3089194be9cc731fab6d"

#include <RemoteXY.h>

// RemoteXY GUI configuration
#pragma pack(push, 1)
uint8_t RemoteXY_CONF[] = // 86 bytes
{ 255,3,0,13,0,79,0,17,0,0,0,31,1,126,200,1,1,5,0,10,
  91,14,28,28,48,4,26,31,79,78,0,31,79,70,70,0,69,84,68,20,
  20,0,1,67,16,74,40,10,4,2,26,11,10,16,15,26,26,48,4,26,
  31,79,78,0,31,79,70,70,0,10,54,15,26,26,48,4,26,31,79,78,
  0,31,79,70,70,0 };

// this structure defines all the variables and events of your control interface
struct {

    // input variables
    uint8_t pushSwitch_1; // =1 if state is ON, else =0
    uint8_t pushSwitch_01; // =1 if state is ON, else =0
    uint8_t pushSwitch_02; // =1 if state is ON, else =0

    // output variables
    int16_t sound_1; // =0 no sound, else ID of sound, =1001 for example, look
    sound list in app
    char text_1[11]; // string UTF8 end zero

```

```

    // other variable
    uint8_t connect_flag; // =1 if wire connected, else =0

} RemoteXY;
#pragma pack(pop)

#include <Servo.h>

Servo myservo; // create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0; // variable to store the servo position

////////////////////////////////////
//      END RemoteXY include      //
////////////////////////////////////

#include <LiquidCrystal.h>
LiquidCrystal lcd(7,8,9,10,11,12);

#include <stdlib.h>
#include <dht.h>
#define dht_dpin A0

int i, j, k;
dht DHT;

void setup()
{
    lcd.begin(16,2);
    lcd.setCursor(0,0);
    lcd.print("DHT Gas Sensor");
    lcd.setCursor(0,1);
    lcd.print(" ");
    delay(2000);
    lcd.clear();
    RemoteXY_Init ();
    myservo.attach(A3);
    myservo.write(pos);
    pinMode(2, OUTPUT);
    digitalWrite(2, HIGH);
    pinMode(3, OUTPUT);
    digitalWrite(3, HIGH);
    pinMode(4, OUTPUT);
    digitalWrite(4, HIGH);
    pinMode(A2, INPUT);

```

```

    pinMode(A1, INPUT);
    // TODO you setup code

}

void loop()
{
    RemoteXY_Handler ();
    DHT.read11(dht_dpin);

    i=DHT.humidity;
    j =DHT.temperature;
    k= analogRead(A1);
    lcd.setCursor(0 ,0);
    lcd.print("T: ");
    lcd.print(j);
    // Serial.println(j);
    lcd.print(" C ");
    lcd.setCursor(8 ,0);
    lcd.print("H: ");
    lcd.print(i);
    lcd.print(" % ");
    lcd.setCursor(0 ,1);
    lcd.print("G: ");
    lcd.print(k);

    lcd.print(" ");
    if(k>600)
    {
        myservo.write(90);
        delay(3000);
        myservo.write(0);
    }
    if(j >40 || i>70 || k>600)
    {

        strcpy (RemoteXY.text_1, "Alert!! ");

    }
    else
    {
        strcpy (RemoteXY.text_1, " ");
    }
    if (digitalRead(A2) == HIGH) {
        RemoteXY.sound_1 = 1001;
    }
}

```

```
else {  
  RemoteXY.sound_1 = 0;  
}  
  if (RemoteXY.pushSwitch_01!=0) {  
    /* button pressed */  
    digitalWrite(2, LOW);  
  }  
  else {  
    /* button not pressed */  
    digitalWrite(2, HIGH);  
  }  
  if (RemoteXY.pushSwitch_02!=0) {  
    /* button pressed */  
    digitalWrite(3, LOW);  
  }  
  else {  
    /* button not pressed */  
    digitalWrite(3, HIGH);  
  }  
}
```