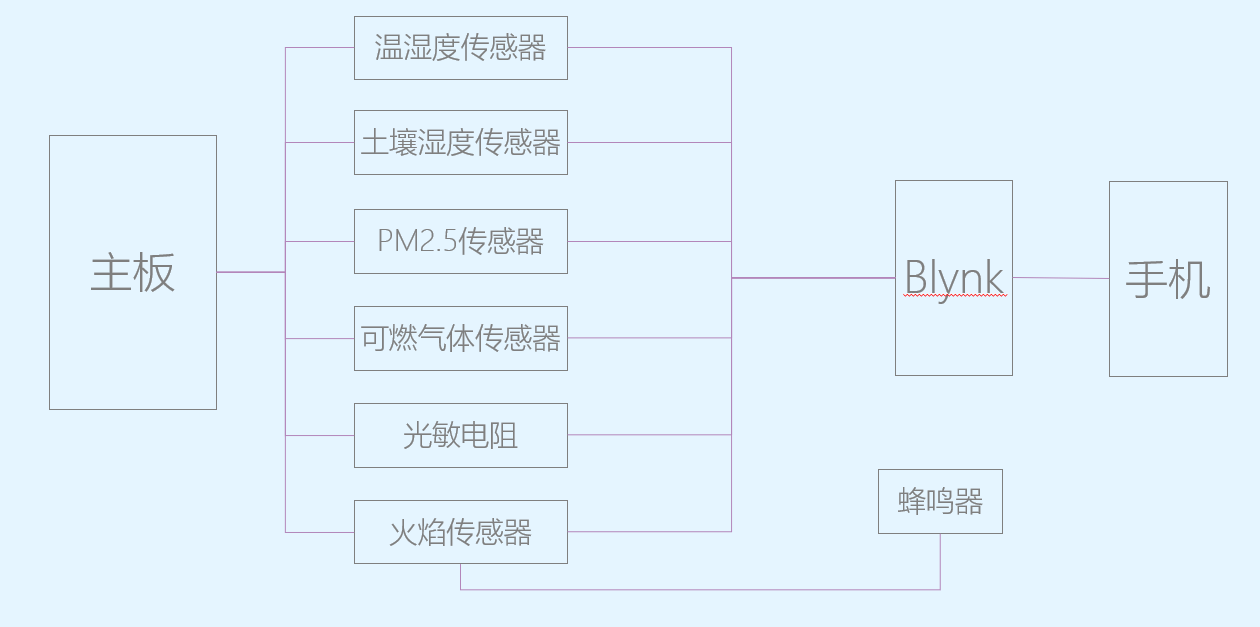
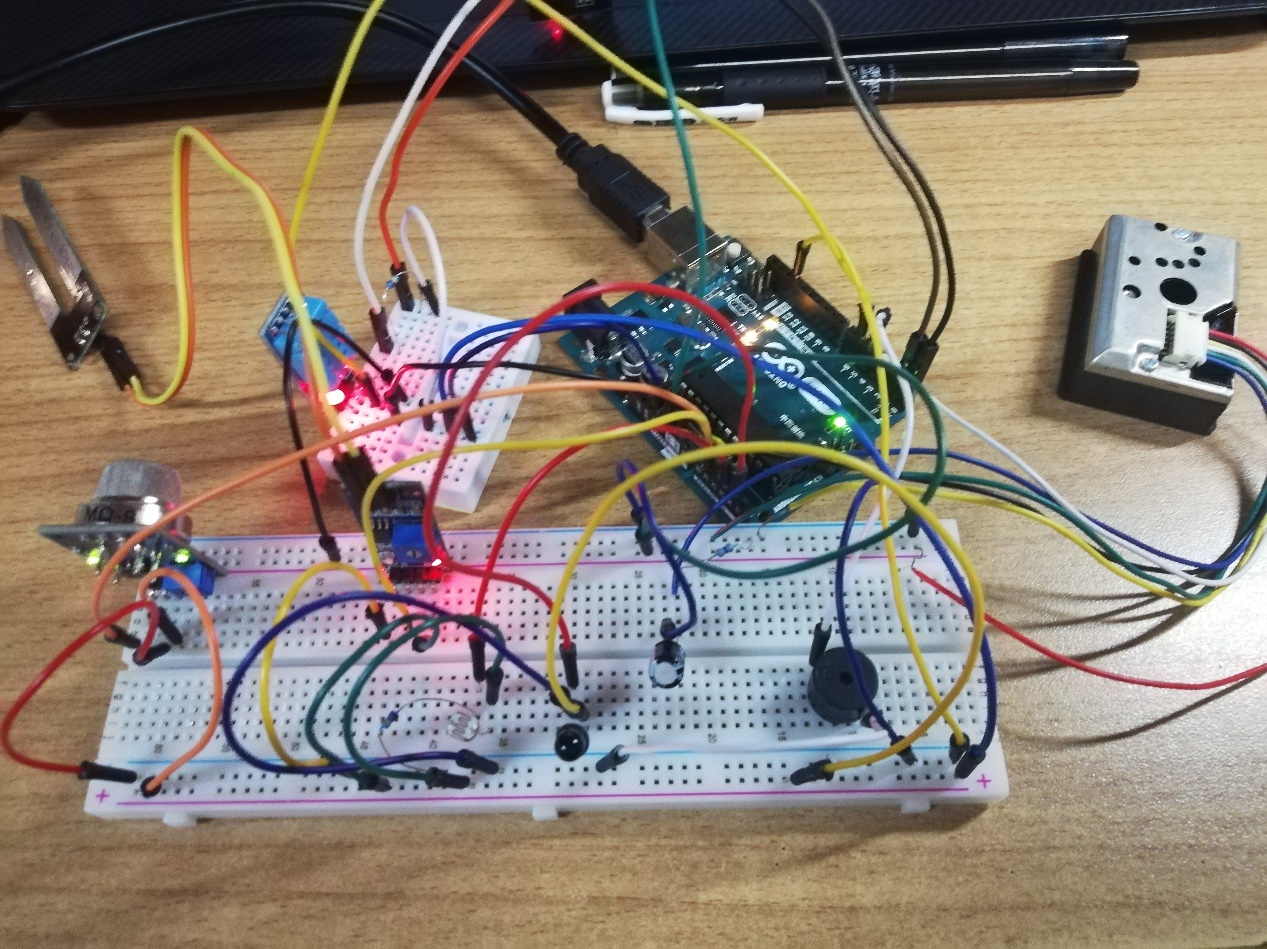
详细设计





首先放一张结构图

再来一张电路实体图

主板选用的是Arduino UNO，之后连接了八个模块，其中六个模块为数据读取（温湿度传感器、土壤湿度传感器、可燃气体传感器、PM2.5传感器、光敏电阻、火焰传感器），还有两个模块，一个是用来连接到互联网的ESP8266-12E模块，另一个是用来输出的无源蜂鸣器模块。

下面是代码部分（会有具体注释）

#define BLYNK\_PRINT Serial

#include <ESP8266\_Lib.h> //wifi模块的库文件

#include <BlynkSimpleShieldEsp8266.h> //连接Blynk的库文件

#include <dht11.h> //温湿度传感器的库文件

//定义全局变量，一些引脚

dht11 DHT11;

int gasSensor\_D\_0 = A1;

int soilHumidity\_0 = A2;

int sensorPin = A3;

int sensorfire=A4;

int measurePin =A0;

int ledPower = 4;

float dustDensity = 0;

int length;

int tonepin=5;

int val=0;

//定义连接Blynk的变量

char auth[] = "2c0c629216114fe38fdf96327a7547e5";

char ssid[]="0315";

char pass[]="wt511355803";

//定义连接ESP8266-12E的变量

#define EspSerial Serial

#define ESP8266\_BAUD 115200

ESP8266 wifi(&EspSerial);

//定义Blynk的虚拟引脚

#define PIN\_UPTIMET V5

#define PIN\_UPTIMEH V6

#define PIN\_UPTIMEsh V7

#define PIN\_UPTIMEgs V8

#define PIN\_UPTIMEPM V9

#define PIN\_UPTIMEG V10

#define PIN\_UPTIMEFIRE V11

//定义个变量

BlynkTimer timer;

//发送PM2.5数据的函数

void sendpm()

{

//定义变量

unsigned int samplingTime = 280;

unsigned int deltaTime = 40;

unsigned int sleepTime = 9680;

float voMeasured = 0;

float calcVoltage = 0;

//读取值

digitalWrite(ledPower,LOW);

delayMicroseconds(samplingTime);

voMeasured = analogRead(measurePin);

delayMicroseconds(deltaTime);

digitalWrite(ledPower,HIGH);

delayMicroseconds(sleepTime);

//计算PM2.5

//calcVoltage = voMeasured\*(5.0/1024);

//dustDensity = 0.17\*calcVoltage-0.1;

calcVoltage= voMeasured/1024-0.0356;

dustDensity=calcVoltage\*42;

//if ( dustDensity < 0)

//{

// dustDensity = 0.00;

//}

Blynk.virtualWrite(PIN\_UPTIMEPM,dustDensity);

}

//发送各种数据的函数

void sendsensor()

{

//读取值与计算

float sh=analogRead(soilHumidity\_0);

float gs=analogRead(gasSensor\_D\_0);

int chk=DHT11.read(6);

float temp = DHT11.temperature;

float humi = DHT11.humidity;

humi=humi/1024\*100;

float g=analogRead(sensorPin);

sh=sh/1024\*100;

g=g/1024\*100;

//发送值

Blynk.virtualWrite(PIN\_UPTIMET,temp);

Blynk.virtualWrite(PIN\_UPTIMEH,humi);

Blynk.virtualWrite(PIN\_UPTIMEsh,sh);

Blynk.virtualWrite(PIN\_UPTIMEgs,gs);

Blynk.virtualWrite(PIN\_UPTIMEG,g);

}

//发送火焰信息以及警报器的函数

void fireornot()

{

//蜂鸣器

#define F -1

#define F1 350

#define F2 393

#define F2S 415

#define F3 441

#define F4 495

#define F5 556

#define F6 624

#define F7 661

#define FL1 175

#define FL2 196

#define FL3 221

#define FL4 234

#define FL4S 247

#define FL5 262

#define FL6 294

#define FL7 330

#define FH1 700

#define FH2 786

#define FH2S 830

#define FH3 882

#define FH4 935

#define FH4S 988

#define FH5 1049

#define FH6 1178

#define FH7 1322

#define FHH2 1568

int tune[]=

{

FL6,FL6,F6,F3,F2S,F,F2,F1,FL6,F1,F2,

FL5,FL5,F6,F3,F2S,F,F2,F1,FL6,F1,F2,

FL4S,FL4,F6,F3,F2S,F,F2,F1,FL6,F1,F2,

FL4,FL4,F6,F3,F2S,F,F2,F1,FL6,F1,F2

};

float durt[]=

{

0.25,0.25,0.5,0.75,0.25,0.25,0.5,0.5,0.25,0.25,0.25,

0.25,0.25,0.5,0.75,0.25,0.25,0.5,0.5,0.25,0.25,0.25,

0.25,0.25,0.5,0.75,0.25,0.25,0.5,0.5,0.25,0.25,0.25,

0.25,0.25,0.5,0.75,0.25,0.25,0.5,0.5,0.25,0.25,0.25

};

length=sizeof(tune)/sizeof(tune[0]);

//判断火情

if(analogRead(sensorfire)>=20)

{

String fireyes="有火";

Blynk.virtualWrite(PIN\_UPTIMEFIRE,fireyes);

for(int x=0;x<length;x++)

{

tone(tonepin,tune[x]);

delay(500\*durt[x]);

noTone(tonepin);

}

}

else

{

String fireno="无火";

Blynk.virtualWrite(PIN\_UPTIMEFIRE,fireno);

}

}

void setup()

{

Serial.begin(9600);

delay(10);

EspSerial.begin(ESP8266\_BAUD);

delay(10);

Blynk.begin(auth, wifi, ssid, pass);

pinMode(6,OUTPUT);

pinMode(ledPower,OUTPUT);

pinMode(tonepin,OUTPUT);

timer.setInterval(1000L,sendpm);

timer.setInterval(1000L,sendsensor);

timer.setInterval(1000L,fireornot);

}

void loop()

{

Blynk.run();

timer.run();

}