

PM2.5 检测器

各位生活在帝都、魔都的朋友们是否会有这样的感慨：早晨一推开窗，浓重的雾霾天气，让你鼓不起勇气迈步杀进那个仿佛时刻在摧毁你五脏六腑的外面的世界。

各位心灵手巧的 Maker 小伙伴们，是时候动手做一个空气污染监测装置了。利用一个空气监测传感器搭配 Arduino 控制器和扩展板，马上就可以搭建出一个简易空气监测装置。即使没有那些专业的监测设备和砖家的报告，你也能科学客观的了解你时刻呼吸的空气质量！

硬件材料：

- [粉尘传感器](#)
- <http://www.dfrobot.com.cn/goods.php?id=698>
- [DFRduino UNO R3](#)
- <http://www.dfrobot.com.cn/goods.php?id=521>
- [IO 传感器扩展板 V7](#)
- <http://www.dfrobot.com.cn/goods.php?id=791>
- [转接板](#)
- <http://www.dfrobot.com.cn/goods.php?id=827>

硬件连接：

- 1、I/O 扩展板插在 UNO 上
- 2、粉尘传感器与转接板通过杜邦线连接
- 3、转接板上标有分别标有 D 和 A 的两个接口
 - D → 数字口 2
 - A → 模拟口 0

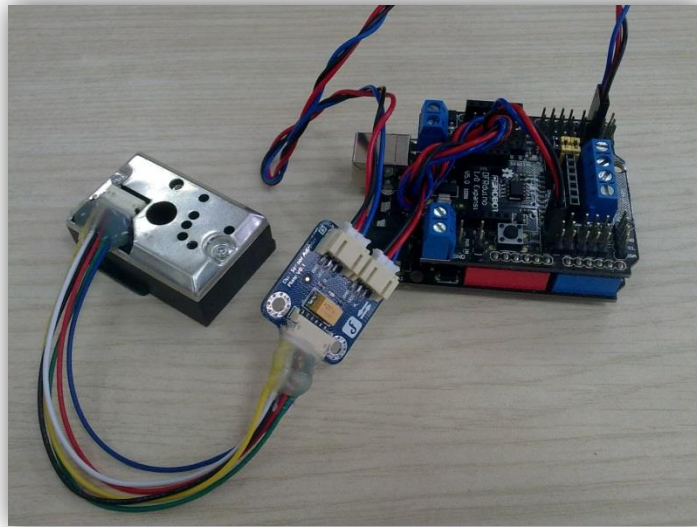


图 1 完整连接图

只要轻松的插上几根线就完成连接，是不是很方便呀？

4、下载程序

```
int measurePin = 0; // 连接模拟口 0  
int ledPower = 2; // 连接数字口 2
```

```
int samplingTime = 280;  
int deltaTime = 40;  
int sleepTime = 9680;
```

```
float voMeasured = 0;  
float calcVoltage = 0;  
float dustDensity = 0;
```

```
void setup(){  
  Serial.begin(9600);  
  pinMode(ledPower,OUTPUT);  
}
```

```
void loop(){  
  digitalWrite(ledPower,LOW); //开启内部 LED  
  delayMicroseconds(samplingTime); // 开启 LED 后的 280us 的等待时间
```

```
  voMeasured = analogRead(measurePin); // 读取模拟值
```

```
  delayMicroseconds(deltaTime); // 40us 等待时间  
  digitalWrite(ledPower,HIGH); // 关闭 LED  
  delayMicroseconds(sleepTime);
```

```

// 0 - 5V mapped to 0 - 1023 integer values
// recover voltage
calcVoltage = voMeasured * (5.0 / 1024.0); //将模拟值转换为电压值

// linear eqaution taken from http://www.howmuchsnow.com/arduino/airquality/
// Chris Nafis (c) 2012
dustDensity = 0.17 * calcVoltage - 0.1; //将电压值转换为粉尘密度输出单位

Serial.print("Raw Signal Value (0-1023): ");
Serial.print(voMeasured);

Serial.print(" - Voltage: ");
Serial.print(calcVoltage);
Serial.print(" - Dust Density: ");
Serial.println(dustDensity); // 输出单位: 毫克/立方米

delay(1000);
}

```

5、下载完之后，得到一串的数字是不是还是很模糊呀？我们先简单了解下粉尘传感器，可以对应代码一道看！（如果觉得不需要了解的话，只需直接下载代码即可使用。）

下面这张图 2 是个粉尘传感器的内部结构图以及硬件说明：

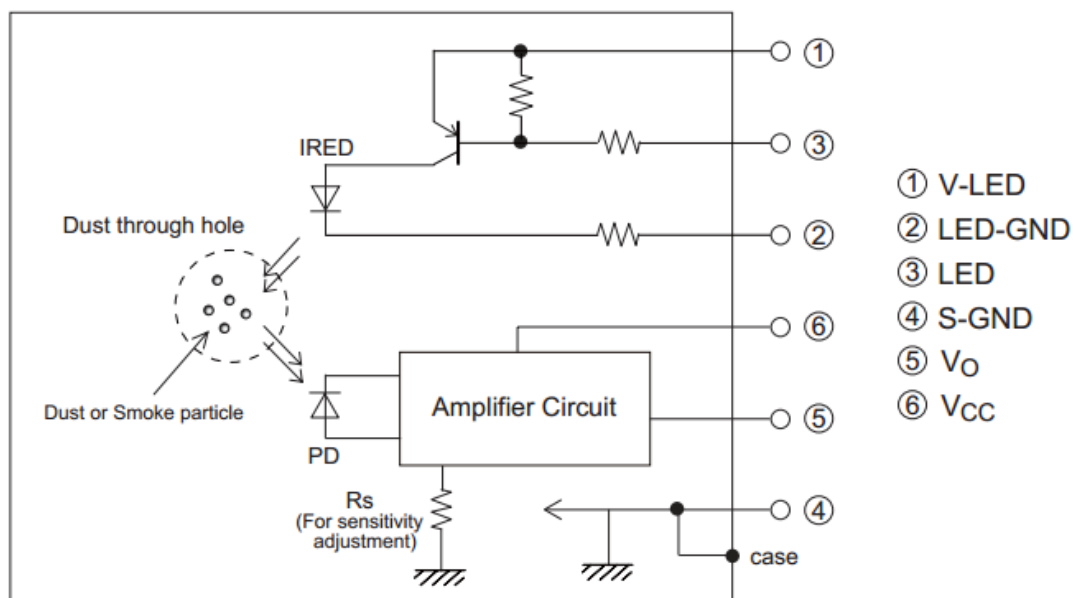


图 2 粉尘传感器内部结构图

可以从图 2 看出，在发射端，我们需要驱动一个红外发射管，也就是引脚 3。

对应程序中定义的 *ledPower*。同样，我们也可以看出输出是引脚 5，输出的是模拟量，对应到程序中的 *measurePin*。

那我们如何得到我们想要的值呢？

下图摘自 datasheet，你可以看出粉尘传感器的输出电压是几乎呈线性的。 **$dustDensity = 0.17 * calcVoltage - 0.1$** ；这个公式曲线近似转换过来的（来自 Chris Nafis）。这里 *dustDensity* 是粉尘密度值，单位 mg/m^3 。*calcVoltage* 是输出电压值。

Fig. 3 Output Voltage vs. Dust Density

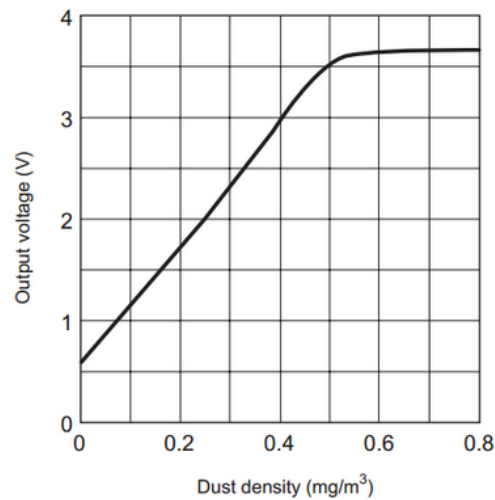


图 3

我们都知道输出 0~5V 电压对应 UNO 模拟口读出的是 0~1023。 **$calcVoltage = voMeasured * (5.0 / 1024.0)$** ；这个公式就是模拟值转为电压值。*voMeasured* 模拟口输出量。

采用时间计算

根据 datasheet，也就是图 5 可看出，我们需要开启内部的 LED 并等待 280us（微秒），之后再读取输出值。图 4 看出，整个脉冲持续时间为 320us。因此，我们还需再等待 40us，才能将 LED 关闭。

Pulse-driven wave form

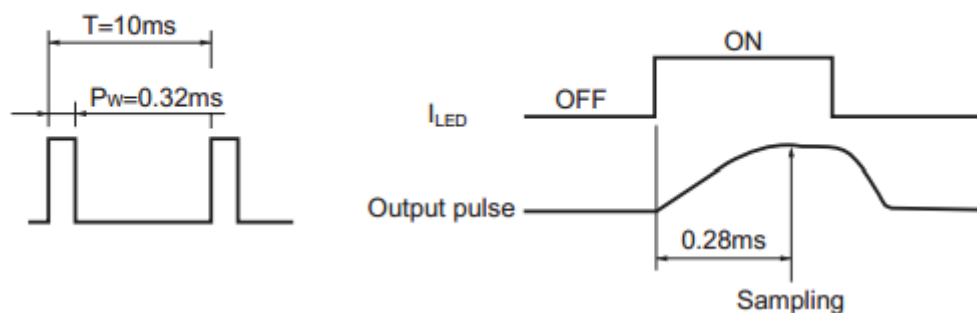


图 4

图 5

以上就是对程序以及，粉尘传感器工作原理的简单说明。我不知道是否讲明白了？

还有个问题，有人说如果没有转接板怎么办，那就接着看下面这张图。

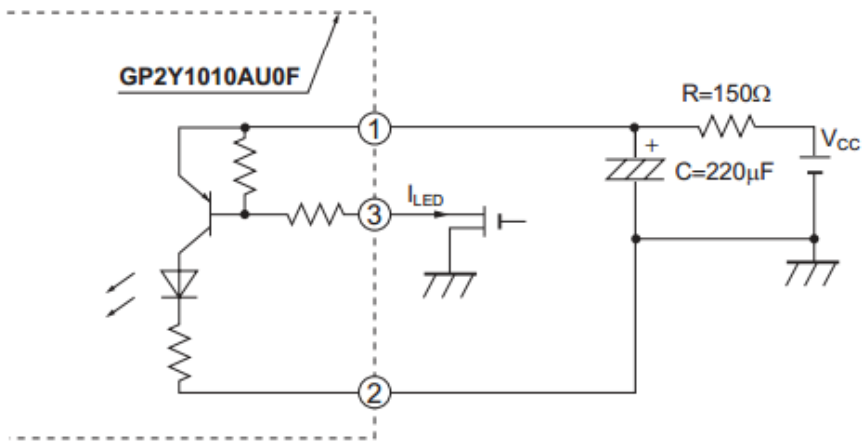


图 3 外接图

看出来了吗？我们需要在引脚 1 外接一个 150 欧的电阻，并且并联一个 220uF 的电容。这也就是我们使用转接板的原因，把这个部分在转接板上完成了，大大方便了连线，也增加了稳定性。

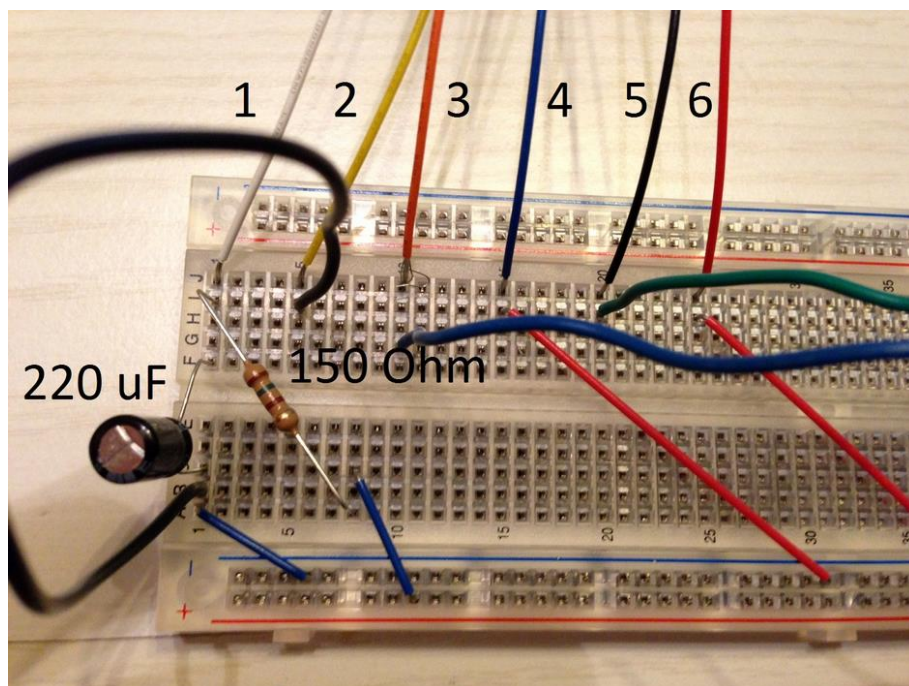
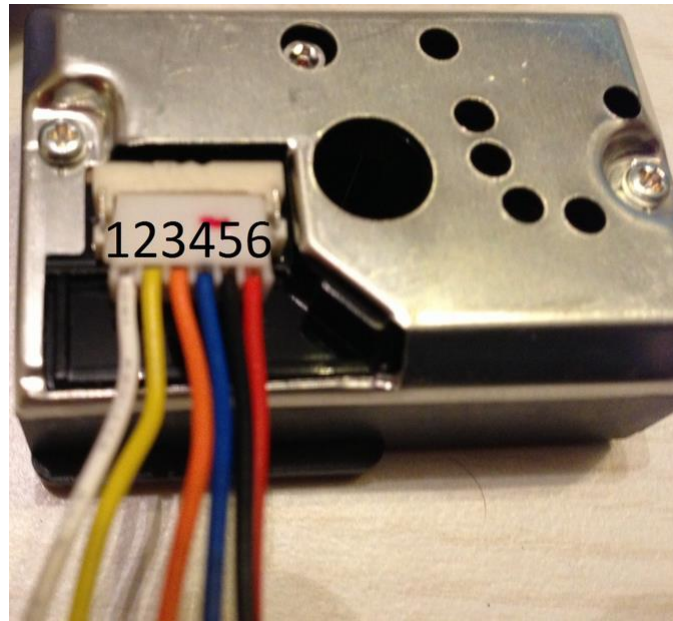
下面来看下实物图上的引脚位置，如果是需要自己外接电阻、电容的朋友们请继续往下看。



图 4 引脚位置图

粉尘传感器对应的 Arduino 引脚:

引脚号	传感器引脚	Arduino 引脚
1	V-LED	5V(串联一个 150 欧的电阻)
2	LED-GND	GND
3	LED	数字口 2
4	S-GND	GND
5	Vo	模拟口 0
6	Vcc	5V



照着连就可以了，连接完成之后，同样下载上面的样例代码。