BME 301: Biomedical Engineering Design (Spring 2010)

Project #19: Indoor Air Quality Monitor

Final Report

May 5, 2010

Prepared by:
Ben Fleming (Team Leader)
Cole Drifka (BSAC)
Lauren Eichaker (Communicator)
Adam Pala (BWIG)

Client:
Dr. David Van Sickle, Department of Population Health Sciences (University of Wisconsin-Madison)

Advisor:
Peter Klomberg, Department of Biomedical Engineering (University of Wisconsin-Madison)

Table of Contents

Abstract

This semester, the team set out to design a device which could display, in a qualitative manner, the quality of ambient indoor air. Currently, there are no tools on the market that display air quality in a universal manner. The motivation for this project came from Dr. David Van Sickle, who studies pulmonary diseases and strives to educate people in other countries about the effects of poor air quality on the human body. He requested that the team design a device that monitors particulate matter and then portrays its readings in a fashion that requires no use of language. The team had to function within a budget constraint of $1,000. Once the design process had begun, the main design alternatives considered was the type of user interface and which harmful airborne substances to monitor in order to define "air quality." In the end, the team was able to construct a device which is composed of a Shinyei PPD4NS particulate monitor, with a user interface similar in appearance to a modern stoplight; a green light signifies good air quality, a yellow light signifies air of moderate quality and a red light denotes poor air quality. An Arduino® microcontroller is responsible for interpreting the output of the sensor and controls the display.

Problem Statement

Poor indoor air quality is a major issue worldwide and results in millions of deaths each year. Currently, there is no simplistic and inexpensive device which monitors air quality. Our goal is to design a device that will monitor particulate matter levels in the surrounding air, display these levels in a clear manner, log the data which it collects, and provide a means of downloading this data for further research.

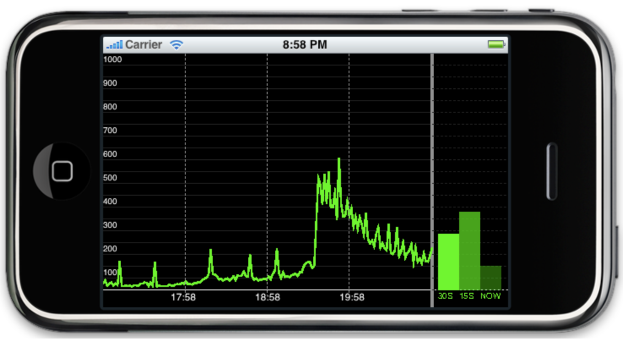
Background Information

A typical person spends 90% of the day indoors eating, sleeping, working, cooking, and spending time with family [1]. However, poor indoor air quality can contribute to the development of chronic respiratory diseases such as asthma, heart disease, and lung cancer [1]. The U.S. Environmental Protection Agency (EPA) designates a standardized air quality level known as the Air Quality Index (AQI). This is a function of mainly the following pollutants: carbon monoxide, nitrogen dioxide, particulate matter, carbon dioxide, and sulfur dioxide [2].

www.treehugger.com/20091203-india-cooking.jpg

In the last decade, a number of quantitative epidemiological studies have been carried out in low and middle-income developing countries that attribute respiratory diseases to indoor tobacco use and burning of biomass fuels for cooking and heating purposes [3]. Indoor tobacco use results in high concentrations of volatile organic compounds such as benzene and carbon monoxide. Indoor burning of biomass fuels, such as liquefied petroleum gas or kerosene, is a major source of particulate matter. Particulate matter is conventionally termed “coarse” if more than 10 microns in diameter and “fine” if less than 10 microns in diameter [4]. Exposure to fine particles, notably $PM_{2.5}$, poses the greatest respiratory risks because they can penetrate deep into the lung alveoli upon inhalation [1]. The EPA standard is that indoor $PM_{2.5}$ mass concentrations should not exceed an average of 35 $\mu g/m^3$ in a 24-hour period [4]. However, a study of Indian households has detected mass concentrations of more than 1000 $\mu g/m^3$ [3]. Poor air quality due to indoor tobacco use and biomass fuel burning is thought to result in 1.5-2 million deaths per year [3].

Developing a reliable method to detect pollutants in the air is important as the initial process of air-quality improvement techniques such as source control, improved ventilation, and air cleaning [2]. Currently, a variety of new commercial products monitor and provide visual feedback on indoor air quality (for examples, please pursue the Ambient EnergyJoule or inAir references); however, they incorporate technology that may not be well-suited for developing countries. The main motivation behind this project is to develop a low-cost indoor air quality monitoring device that can be used as an intervention tool in developing households adhering to indoor tobacco use and biomass fuel burning. It will seek to monitor some indicator of poor air, such as carbon monoxide, $PM_{2.5}$, or a combination of the two, and display the overall air quality in a simple “stoplight” manner so families can easily see current conditions and the improvements due to lifestyle changes seeking to reduce indoor pollution. From a research perspective, the device will also periodically log air quality measurements and provide a means to download the data.



inAir: Measuring and Visualizing Indoor Air Quality, Sunyoung Kim & Eric Paulos

If this project results in a functional prototype, it will be rapidly integrated into two ongoing projects on indoor air quality in India and Rwanda, as well as Project Quit Tobacco, a large US National Institutes of Health-funded tobacco cessation project currently being carried out in India and Indonesia.

Design Motivation

Poor indoor air quality is a major global health problem responsible for a vast epidemic of chronic respiratory disease. In low and lower-middle income countries, tobacco smoke and the burning of biomass fuel sources for heating and cooking result in very high levels of indoor particulates, resulting in millions of deaths per year. The objective of this project was to develop a new kind of indoor air quality monitor designed to provide simple, ongoing feedback to household residents on the quality of their indoor air. The client, Dr. David Van Sickle and his team, intends to implement the final product into two ongoing projects on indoor air quality in India and Rwanda, as well as Project Quit Tobacco [9]. The team thus sought to create a device to educate households about the hazards of poor air quality through the creation of a particulate monitoring machine.

Client Requirements

Our client had a few specific requirements for this device which we used to construct our product design specifications. First, our client indicated that he wanted the product to monitor the indoor air quality specifically related to tobacco smoke and biomass fuel use; biomass fuel use involves burning organic fuel sources such as wood, coal, etc. [5]. Also, our client required our tool to indicate the level of air quality to the user in a simple fashion. This would incorporate a key function of the device: to serve as an in-home educational tool. The user interface needed to span language and cultural barriers, as the device is intended for use in India and Rwanda. It was based on these concerns that the team set out to design the user interface. Another key function of the device was to serve as a research tool. Thus, the client required that the device have some sort of non-volatile memory to record data. The client required that the data be taken at a frequency of approximately once per minute. Also, this data would need to be easily downloaded to a computer for analysis at a later date. Lastly, the client required that the device should function for at least 1 year without user maintenance. The main problem with this concept is that the areas in which our design is intended to be used (small villages in India and Rwanda) commonly experience power outages. The client therefore specifically required that the device have some means of a “back-up” power source. Besides having a consistent power source for one year, other components of the device should not break down within that time frame. In line with this concept, the device should be able to function at various conditions including wide temperature ranges.

Design Specifications

Based on the client's requirements we were able to compile several specifications for our initial prototype. The first set of specifications pertains to operational features. First, after considerable research and debate, we decided to monitor only particulate levels. Particulates are defined as "tiny subdivisions of solid or liquid matter suspended in a gas or liquid" [6]. This decision was based on the fact that particulates are the most diverse and all-inclusive byproduct of both tobacco smoke and biomass fuel burning [7]. Next, based on our client requirements for the user interface, we decided to use LEDs (light emitting diodes) to indicate the level of indoor air quality. We came up with a design that would make use of three separate LEDs in a manner similar to a stoplight. The different LEDs (green, yellow, red) would light up or turn off at different threshold values to indicate the relative level of contamination in the air. In attempting to solve the problem of providing power to our device, we decided to pursue an option based on one already used in other electronic devices: our instrument will utilize mainly a wall outlet (via a plug) power source, but also have a rechargeable battery, which will serve as the "back-up" power source. This design is similar to that used in some carbon monoxide detectors, which plug into the wall, or some clock-radios.

Our next set of design specifications defines physical aspects of our product. We determined that the device should function between 0 and 60 °C. The extreme end limits of this range were determined based on the extreme climates present in India and Rwanda; the thoughts of the client also enforced these limits. Also, the device should not be too small or too large. Thus, we determined that a size of approximately 8cm×10cm×8cm and a mass of approximately 2 kg are appropriate. Also, it was a given that our device would be made up of several different, basic electronic components. Most important among these are the particulate sensor and the programmable Arduino® microcontroller.

Lastly, we accumulated a few miscellaneous specifications pertaining to the prototype. We will only make one device initially. However, if our prototype is shown to work effectively, then several devices will be made and possibly put into use for research being conducted this summer (2010). While we have a larger budget for our initial design ($1,000 USD), it has been determined that the "off-the-shelf" cost of one of these devices should not exceed $100. Lastly, our device must bring something new or novel in its function and operation when compared to other devices, as there are several similar devices on the market. However, we feel that, given our specific design, our device will provide something unique and new compared to what is currently available in particulate detection.

Human Factors, Ergonomics and Engineering Ethics

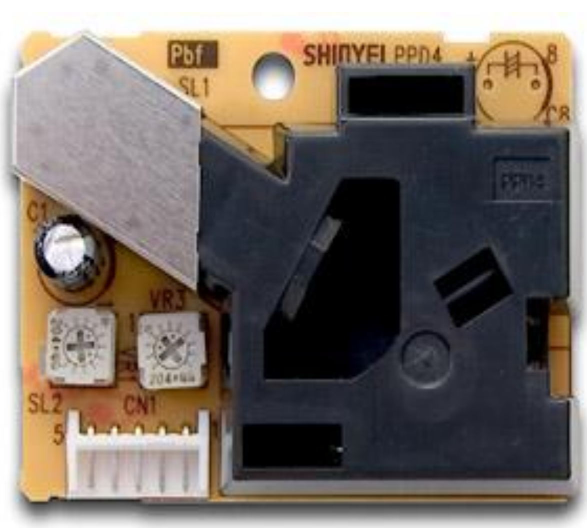
A primary concern in designing the air quality monitor device is ease of use. The first issue here is ease of interpreting the relative level of air quality from the user interface. As mentioned earlier, this device is meant to serve as an educational/intervention tool in different regions of the world. Thus, it is essential that we insure that our user interface effectively communicates the quality of the air in its vicinity. The other main concern regarding human factors is closely related to the other main function of our device. Our device is meant to serve as a research tool, and thus must collect data that is easily downloadable. Thus, if one of the researchers wants to examine the data collected from a specific device, it should be no more difficult than connecting the device via USB to a computer.

As mentioned in the *Background* section, our device will serve as a means to define air quality to households in other countries. When calibrating our device, we will therefore need to correctly correlate the light display (green, yellow and red) with a particulate range that is not detrimental, moderately bad and harmful towards human health, respectively. The educational aspect of our device must be taken into account.

Features of the Final Design

The following prototype components effectively and efficiently work in concert to provide the desired user input-output functions, and they are therefore a logical choice for the purposes of the proposed design. The main features include: the sensor, the microcontroller, the power supply, and the data processing code.

The Optical Particulate Matter Sensor



http://www.shinyei.co.jp/stc/optical/dust_e.html

The model PPD4NS particulate matter sensor unit by Shinyei Corp. (shown to left) detects airborne particles based on a light scattering method. Air is self-aspirated utilizing a built-in heater to generate an air current. The sensor's receptor receives scattered light from a particle as a pulse. Each raw pulse is amplified by an op-amp so that the pulse can be acknowledged clearly. A fixed threshold (1V for output P1) is used to select pulses whose height is over the threshold. The low-pulse-occupancy-time ratio (LPOT) over 30 seconds is used to determine the concentration of particulates in each air sample. This can be used in subsequent processing to indicate air quality (as well as for data logging). The data sheet for this component may be found in the appendix. According to the U.S. Environmental Protection Agency (EPA), which maintains an Air Quality Index (AQI), relatively healthy air quality conditions for 2.5 micron (or less) particulate matter ($PM_{2.5}$) are those which are less than 35.5 µg/m^3; moderately healthy air quality conditions (i.e. healthy for the general public, but potentially hazardous for certain sensitive groups) are those between 35.5 µg/m^3 and 65.4 µg/m^3; and air quality conditions 65.5 µg/m^3 or above are considered generally unhealthy [8].

The Arduino Duemilanove® Microcontroller

The microcontroller is used to process the analog voltage input from the sensor in order to produce a meaningful output for the user. The Arduino Duemilanove® is an open-source, versatile device based on the ATMega328 microcontroller which can be used to implement a variety of input-output functions [10]. The Duemilanove® has six analog input pins (each with 10 bits of resolution) and fourteen digital pins which can be configured either as inputs or outputs (5V, 40 mA maximum operation). The Duemilanove® has a USB port which allows for communication with computers (i.e. to upload code to the microcontroller and to log data). Therefore, the Duemilanove® is an excellent solution to bridge the input from the sensor to the LED output (this process will be described in the sections titled *The Code* and *Final Design*).

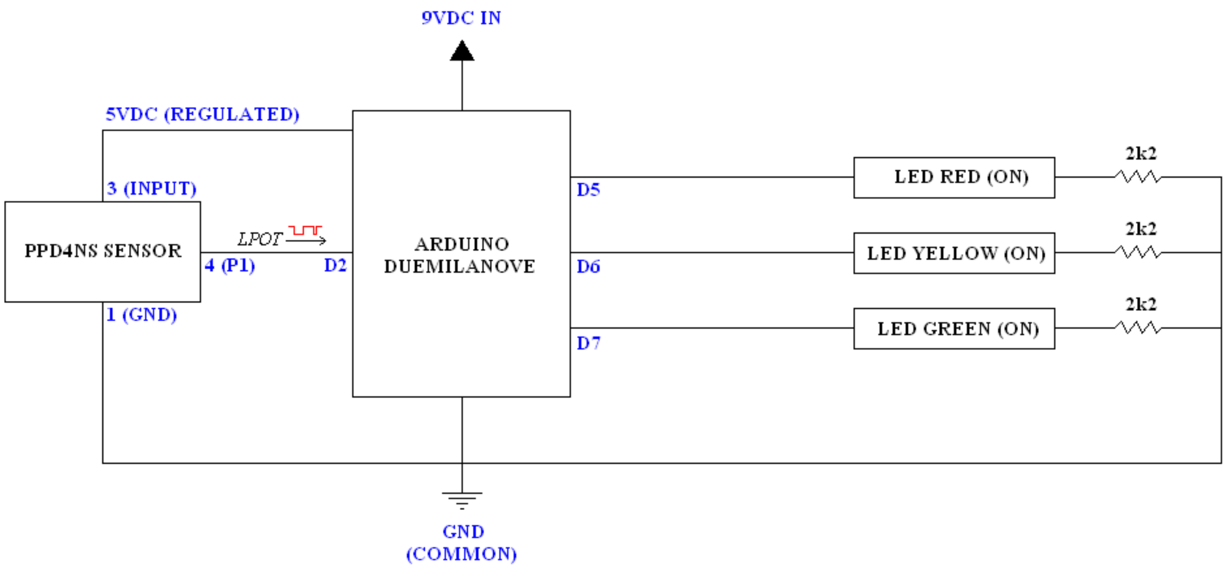
http://www.inmotion.pt/store/product_info.php?cPath=10&products_id=56

The Power Source

Given that the device (e.g. the Arduino Duemilanove®) requires power to run the peripheral components, a power source must be included accordingly. The Duemilanove® includes a power-supply input socket which can receive inputs from 7 to 12 VDC and includes an on-board voltage regulator to provide 3.3 VDC (50 mA maximum current) or 5 VDC for the analog and digital pins [10]. The Duemilanove can also be powered via the USB input (in addition to the aforementioned programming capabilities via USB). Per the client’s instruction, it would also be desirable to provide an alternative rechargeable-battery power supply to run the device when wall power is unavailable. Other design considerations include variations in wall power supplies (i.e. 120 V at 60 Hz versus 230 V at 50 Hz), which will necessitate the selection of a power supply with power transformation capabilities, as well as a switching mechanism from wall power to the rechargeable batteries (this ideally may be automated using the Duemilanove®).

The Data Processing Code

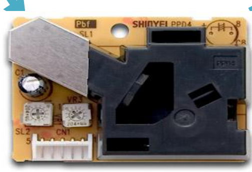
In order to instruct the microcontroller how to handle the output of the particulate sensor, we needed develop original code for the Arduino®. First, variables were declared in order to specify which digital pins serve as outputs, which serve as inputs, where we would store data such as the LPOT, and so on. The respective digital pins were set as outputs or inputs by using the "pinMode" function. The code was then set to loop continuously for the duration of time that the device is being powered. Using a do/while loop, the LPOT is calculated for the given time interval and if/else statements are used to determine which LEDs to illuminate. For example, if the LPOT is between 4% and 8%, a yellow LED display appears. At the end of the given time interval, the program begins again and a new LPOT is calculated for the next data collection period. See the appendix for the actual code. The block diagram below summarizes this sequence.



Final Design Overview

The design components (listed) describe the components of the final design. With its simple design, low distribution of LEDs, and ease of construction, the stoplight model was chosen to portray the quality of air sensed by the final product. The final design thus utilizes the stoplight LED display to portray the output of the PPD4NS sensor, utilizing an Arduino Duemilanove® microcontroller to integrate the various inputs and outputs. Various circuit components, such as: the power source, the case, resistors, a toggle switch, a breadboard, and 22 gauge wire finish the device.

http://tuberose.com/Cigarette_Smoking.html



Prototype Testing

In order to prove that our device modulates its LED display based on the LPOT output of the particulate sensor, we obstructed the sensor in a manner to activate the green, yellow, and red LEDs at separate time points. Using ELVIS®, we logged the respective voltage pulse data. The data was then exported to Matlab to provide graphical portrayals of the different modes. From this, we calculate the LPOT over the duration of each trial. A calculated LPOT of 0% resulted in the illumination of the green LEDs (the activation range was arbitrarily set at 0%<LPOT<4%). When a LPOT was calculated to be 6.3%, the yellow LEDs were activated (activation range set at 4%<LPOT<8%). Lastly, when a LPOT was calculated to be 55.1%, the red LEDs were illuminated (activation range set at LPOT>8%). These results allowed us to conclude that our device is indeed capable of qualitatively displaying data based on LPOTs due to sensor obstruction and set activation thresholds.

Future Work

Currently, two of the main features desired by the client are missing from the prototype. The battery backup and memory for the purpose of data logging were not incorporated into the design. Although the client noted that these features were optional, the team realizes that they would enhance the effectiveness of the device. In India, where power outages are frequent, the battery backup would allow the device to function 24 hours a day; for the team researching the quality of air in other countries, the data recorded utilizing a memory function, would allow for the tracking of particulate counts.

Testing of our device needs extension; optimally, the team would be able to acquire a particulate sensor with known cut-offs for its output, so that we could calibrate our device. This would allow for the accurate portrayal of air quality, as mentioned in the section titled *Human Factors, Ergonomics and Engineering Ethics.* Once the battery backup system is implemented, we would need to test this feature using simulated power outages. Extensive switching between LED colors would test the circuitry of our device when used for extensive periods of time, with fluctuating particulate concentrations.

Other areas of future work may include sensing other air pollutants in addition to particulate matter such as CO or NO, integrating the array of sensor outputs, and providing a weighted proxy of the real-time air quality. Also, a means to upload data wirelessly by researchers would be advantageous.

Finally, the team could take this device to a classroom, to test its efficacy as a teaching tool on young children. We could also take the device to English as a second language classes for this purpose. This qualitative test would allow us to determine how easy it will be for the researchers using this machine in foreign countries, without the extensive use of language.

References

1. Kim, S, Paulos E. *inAir*: Measuring and Visualizing Indoor Air Quality. *UbiComp*. 2009.

2. Choi S, Kim N, Cha H, Ha R. Micro sensor node for air pollutant monitoring: hardware and software issues. *Sensors*. 2009. 9, 7970-7987.

3. Monkkonen P, Pai P, Maynard A, Lehtinen KEJ, Hameri K, Rechkemmer P, Ramachandran G, Prasad B, Kulmala M. Fine particle number and mass concentration measurements in urban Indian households. *Science of the Total Environment*. 2005. 347, 131-147.

4. US Environmental Protection Agency. 2009. < http://www.epa.gov/air/>.

5. Renewable Biomass. (2009). U.S. Energy Information Administration. http://tonto.eia.doe.gov/kids/energy.cfm?page=biomass_home-basics

6. Particulate. (2010). Wikipedia. http://en.wikipedia.org/wiki/Particulate

7. Exposure to indoor biomass fuel and tobacco smoke and risk of adverse reproductive outcomes, mortality, respiratory morbidity and growth among newborn infants in south India. (2009). Tielsch, J.M., Katz, J., Thulasiraj, R.D., Coles, C.L., Sheeladevi, S., Yanik, E.L., Rahmathullah, L. International Journal of Epidemiology. http://ije.oxfordjournals.org/cgi/content/full/38/5/1351.

8. Air Quality Index (AQI). (2009). U.S. Environmental Protection Agency. http://www.airnow.gov/index.cfm?action=aqibasics.aqi

9. BME Homepage http://ecow2.engr.wisc.edu/courses/file.php/681/Project_proposals_and_lists-S10/Project_List-S10-final.htm

10. Particle Sensor Unit (PPD4NS). (2010). Shinyei. http://www.shinyei.co.jp/stc/optical/dust_e.html

11. Arduino Duemilanove. (2010). Arduino. http://arduino.cc/en/Main/ArduinoBoardDuemilanove

Appendix

Product Design Specifications (PDS)

Project: Indoor Air Quality Monitor

I. Function

This design project is aimed at designing a device which can monitor indoor air quality. The device will be used to improve indoor air quality in low-income areas of countries such as India and Rwanda. Our client requires that the device indicate the level of air quality in a simplistic manner. Also, the device should specifically detect pollutants and particulates in the air which are the direct result of tobacco smoke and biomass fuel burning. The device should have some means of storing the data which it collects. The stored data should be easily downloaded for further research.

II. Client Requirements

- The device must monitor indoor air quality specifically related to tobacco smoke and biomass fuel burning
- The device must indicate the current air quality in an extremely obvious and clear manner
- The device should have some internal memory to which it can record data
- This data should be easily downloaded to a computer for further studying and analysis
- The device should be primarily powered through the wall outlet
- The device should have a rechargeable battery which will serve as a “back-up” power source
- The device should be sufficiently small and lightweight
- The device should be able to function for a significant amount of time without breaking down

III. Design Requirements

1. **Physical and Operational Requirements**

a) **Monitoring Capabilities:** The device must monitor current levels of air pollutants, specifically volatile organic compounds (benzene, carbon tetrachloride, chloroform, and dichloromethane), carbon monoxide, and particulates.

b) **User Interface:** The device will indicate the current air quality to the user in a simplistic manner. The device will use different colored light emitting diodes (LEDs) to tell whether the air quality is poor (red), fair (yellow), or good (green) in a manner analogous to a stoplight. The various LEDs will light up based on danger threshold levels for hazardous air pollutants.

c) **Data Storage:** The device must be able to frequently store the information which it is monitoring to an internal hard-drive. The data should be easily downloaded via a USB port.
d) **Safety:** The device should be safe and pose no immediate risk to the user (s).
e) **Power Supply:** The device will be powered through a common electrical wall outlet. However, the device will have an internal rechargeable battery which will charge during normal use and supply power during an outage.
f) **Operating Environment:** The device should be able to function correctly at various temperature humidity levels. Specifically, the device should work properly at all humidity levels and between 10 and 40 °C.
g) **Size/Weight:** The device should be approximately 8x10x8 cm and have a mass no greater than 2 kg.
h) **Materials:** The device will consist of sensors for the various compounds being monitored, various circuit components, a project box, LEDS, a programmable microcontroller such as that in Arduino ®, a plug for going into a wall outlet, and a rechargeable battery.

2. Production Requirements

i) **Quantity:** Initially, only one functional device will be constructed. However, it is expected that several more of these devices may be manufactured and put to work as early as this summer.
j) **Budget:** The initial prototype should not take more than $1000 to construct.

3. Miscellaneous

k) **Standards and Specifications:** The device should be able to detect levels of particulates and pollutants at certain threshold levels. The device should be tested by seeing whether or not the LED lights change at given particulate or pollutant levels indicated by more exact instruments.
l) **Customers:** Our customers will be those conducting studies on indoor air quality in India and Rwanda, as well as those running Project Quit Tobacco, a U.S NIH funded project.
m) **User-Related Concerns:** The main concern here is that the device is accurate enough to indicate to the user a potentially dangerous air quality situation. Also, the device should pose no significant electrical shock risk to the users and researchers.
n) **Competition**: There are several similar devices on the market. However, none of these seem to monitor exactly the same pollutants as our device will. Also, most of these devices have a somewhat more complicated user interface.

Sensor Data Sheet

DOCUMENT NO. 1

PPD4NS How it works. (How to read pulse output from P1)

Our PPD4NS has pulse output which works like this.

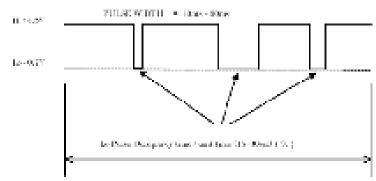
[1] Receptor receives scattered light from the particle, as a pulse.

[2] Each raw pulse is amplified by an op-amp so that pulse can be achnowledged clearly.

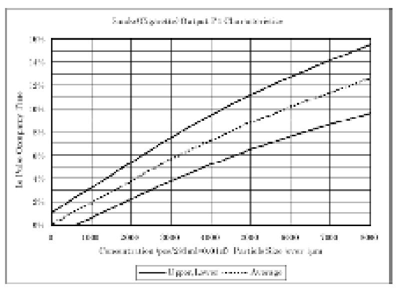
[3] We set a fixed threshold (voltage = 1V for output P1) to select pulse whose height is over the threshold. Selected pulses in this case represent (approx) 1 micron or larger.

With PPD4NS you can read selected pulse, selected with a threshold detection voltage 1V which was converted to Lo Pulse directly at the same time.

When you use our PPD4NS to check unidentified particles in the room, you check the Lo Pulse occupancy time (ratio) over a certain unit time. 30 seconds or longer is recommended as a unit time.



Your microprocessor is to be set to check the pulse condition, say, Hi? Lo? In every 1msec and accumulate Lo pulse occupancy ratio to unit time (30 sec or longer you set). By Lo Pulse Occupancy Ratio you can judge the dust concentration level.



For normal application for Air Purifier, we recommend you to use P1 output signal only for simple designing.

*As for output from P2, this represents large sized particles. This is just for user who would like to do "particle sized descrimination". For further information about "size disrimination" please contact SHINYEI.

Setting example of "table" in between sensor output signal (LO pulse occupnacy ratio) and fan speed control.

In case of 4 fan speeds (Example just for your reference)

	Example A (Standard)	Example B	Example C (Sensitive)
SUPER HI	≧ 6.9	≧ 6.3	≧ 5.9
HI	≧ 5.7 ＊ > 6.9	≧ 5.1 ＊ > 6.3	≧ 4.6 ＊ > 5.9
MID	≧ 4.2 ＊ > 5.7	≧ 3.6 ＊ > 5.1	≧ 3.2 ＊ > 4.6
LOW	≧ 2.7 ＊ > 4.2	≧ 2.3 ＊ > 3.6	≧ 1.9 ＊ > 3.2
STOP	≧ 0.0 ＊ > 2.7	≧ 0.0 ＊ > 2.3	≧ 0.0 ＊ > 1.9

Microcontroller Code

```
int x = 1; //integer value that will be incremented after every 30 seconds
float duration = 0; //the amount of time the program has been running
float pulseLow = 0; //will store the amount of time the pulse from the sensor has been at LOW during the current interval
int digitalInput = 2; //digital pin 2 will serve as the input pin from the sensor
int ledPinRed = 5; //digital pin 5 will serve as output to red led
int ledPinRed1 = 4;
int ledPinYellow = 6; //digital pin 6 will serve as output to yellow led
int ledPinYellow1 = 3;
int ledPinGreen = 7; //digital pin 7 will serve as output to green led
int ledPinGreen1 = 8;
float percentage = 0; //the low pulse occupancy time

void setup()
{
  pinMode(digitalInput, INPUT); //pin 2 will be an input
  pinMode(ledPinRed, OUTPUT); //pin 5 will be an output
  pinMode(ledPinYellow, OUTPUT);//pin 6 will be an output
  pinMode(ledPinGreen, OUTPUT);// pin 7 will be an output
  pinMode(ledPinRed1, OUTPUT); //pin 4 will be an output
  pinMode(ledPinYellow1, OUTPUT);//pin 3 will be an output
  pinMode(ledPinGreen1, OUTPUT);// pin 8 will be an output

   digitalWrite(ledPinGreen, HIGH);
  digitalWrite(ledPinGreen1, HIGH);
  digitalWrite(ledPinYellow, HIGH);
  digitalWrite(ledPinYellow1, HIGH);
  digitalWrite(ledPinRed, HIGH);
  digitalWrite(ledPinRed1, HIGH);
  delay(200);
  digitalWrite(ledPinGreen, LOW);
  digitalWrite(ledPinGreen1, LOW);
  digitalWrite(ledPinYellow, LOW);
  digitalWrite(ledPinYellow1, LOW);
  digitalWrite(ledPinRed, LOW);
  digitalWrite(ledPinRed1, LOW);
  delay(200);
  digitalWrite(ledPinGreen, HIGH);
  digitalWrite(ledPinGreen1, HIGH);
  digitalWrite(ledPinYellow, HIGH);
```

```
  digitalWrite(ledPinYellow1, HIGH);
  digitalWrite(ledPinRed, HIGH);
  digitalWrite(ledPinRed1, HIGH);
  delay(200);
  digitalWrite(ledPinGreen, LOW);
  digitalWrite(ledPinGreen1, LOW);
  digitalWrite(ledPinYellow, LOW);
  digitalWrite(ledPinYellow1, LOW);
  digitalWrite(ledPinRed, LOW);
  digitalWrite(ledPinRed1, LOW);
  delay(200);
  digitalWrite(ledPinGreen, HIGH);
  digitalWrite(ledPinGreen1, HIGH);
  digitalWrite(ledPinYellow, HIGH);
  digitalWrite(ledPinYellow1, HIGH);
  digitalWrite(ledPinRed, HIGH);
  digitalWrite(ledPinRed1, HIGH);
}


void loop() //loop program continuously
{

   do //do the following
   {
     pulseLow += pulseIn(digitalInput, LOW);//pulselow wills store the total amount
of time are input is at low during the current interval
     duration = micros(); //duration should roughly store the current time the
program has been running
   } while (duration / 30e6 < x); //program completes the above statements if, and
only if, we are not at the end of a 30 s interval

     x++;//increment x by 1, this will allow the above loop to work for the next 30 s
interval
     percentage = (pulseLow/30e6)*100;//calculates the low pulse occupancy time
     if (percentage >= 0 && percentage < 4) //if low pulse occupancy time is between
0 (inclusive) and 4 (exclusive) %, green LED is on
     {
       digitalWrite(ledPinGreen, HIGH); //turn the green led on
       digitalWrite(ledPinGreen1, HIGH);
       digitalWrite(ledPinYellow, LOW); //turn the yellow led off
       digitalWrite(ledPinYellow1, LOW);
       digitalWrite(ledPinRed, LOW); //turn the red led off
```

```
      digitalWrite(ledPinRed1, LOW);
      delay(200); //stop program for .2 s, initiate  one blink
      digitalWrite(ledPinGreen, LOW);
      digitalWrite(ledPinGreen1, LOW);
      delay(200);
      digitalWrite(ledPinGreen, HIGH); //green led is on
      digitalWrite(ledPinGreen1, HIGH);
    }
   else if (percentage >= 4 && percentage < 8) //if low pulse occupancy time is
between 4 (inclusive) and 8 (exclusive) %, yellow LED is on
    {
      digitalWrite(ledPinGreen, LOW); //turn the green led on
      digitalWrite(ledPinGreen1, LOW);
      digitalWrite(ledPinYellow, HIGH); //turn the yellow led off
      digitalWrite(ledPinYellow1, HIGH);
      digitalWrite(ledPinRed, LOW); //turn the red led off
      digitalWrite(ledPinRed1, LOW);
      delay(200); //stop program for .2 s, initiate  one blink
      digitalWrite(ledPinYellow, LOW);
      digitalWrite(ledPinYellow1, LOW);
      delay(200);
      digitalWrite(ledPinYellow, HIGH); //green led is on
      digitalWrite(ledPinYellow1, HIGH);
    }
    else { //if low pulse occupancy time is greater than 8 (inclusive) %, red LED is on
      digitalWrite(ledPinGreen, LOW); //turn the green led on
      digitalWrite(ledPinGreen1, LOW);
      digitalWrite(ledPinYellow, LOW); //turn the yellow led off
      digitalWrite(ledPinYellow1, LOW);
      digitalWrite(ledPinRed, HIGH); //turn the red led off
      digitalWrite(ledPinRed1, HIGH);
      delay(200); //stop program for .2 s, initiate  one blink
      digitalWrite(ledPinRed, LOW);
      digitalWrite(ledPinRed1, LOW);
      delay(200);
      digitalWrite(ledPinRed, HIGH); //green led is on
      digitalWrite(ledPinRed1, HIGH);
   }
   pulseLow = 0;//resets the low pulse time to 0 for the next 30 second interval
 }
```

Budget Analysis

Part (quantity)	Manufacturer/Distributer	Part #	Price
Real Time Clock (1)	Digikey	568-1084-5-ND	$3.64
Memory (1)	Digikey	AT24C1024B-PU25-ND	$4.30
Arduino Duemilanove (2)	www.makershed.com	MKSP4 (ATmega328)	$69.98
Electronic Battery 3.6V 1200 mAh 1/2AA (2)	Eagle-Picher from mouser.com	521-PT-2150	$12.18
Air Quality Gas Sensor (1)	www.futurlec.com	MQ135	$6.90
Gas Sensor Socket (3)	www.futurlec.com	GAS_SOCKET	$1.35
Carbon Monoxide Gas Sensor (2)	www.futurlec.com	COSENSOR	$17.80
Half-size breadboard (1)	www.adafruit.com	na	$5.00
Tiny breadboard (1)	www.adafruit.com	na	$4.00
Adafruit Proto Shield for Arduino Kit (v.5)	www.adafruit.com	na	$12.50
Particulate Sensor (1)	Shinyei Corporation of America	PPD42NS	$58.00
Wall Transformer (1)	www.allelectronics.com	DCTX-937	$2.85
LED-red (1)	www.allelectronics.com	LED-1	$1.00
LED-green (1)	www.allelectronics.com	LED-2	$1.50
LED-yellow (1)	www.allelectronics.com	LED-3	$1.50
Project Box (5X2.5X2)	Radio Shack	na	$3.69
Project Box (6X3X2)	Radio Shack	na	$3.79
TOTAL			$209.98