ISSN: 2994-2609

Research Article

Plant Biology Soil Health Journal

Particulate Matter Quantification Via DINÉ (Digitally INtegrated Environmental) Arduino UNO R3 Platform for Environmental Quality, Safety, and Health on the Navajo Nation

Ember Bahe[1] and Daniel Winarski[2*]

[1] *University of Arizona, Tucson, Arizona*

[2] *Independent Researcher, Tucson, Arizona*

[*] Corresponding Author
Daniel Winarski, Independent Researcher, Tucson, Arizona, US.

Submitted: 2025, Jan 17; **Accepted:** 2025, Feb 24; **Published:** 2025, Mar 03

Citation: Bahe, E., Winarski, D. (2025). Particulate Matter Quantification Via DINÉ (Digitally INtegrated Environmental) Arduino UNO R3 Platform for Environmental Quality, Safety, and Health on the Navajo Nation. *Plant Biol Soil Health J, 2*(1), 01-16.

Abstract

After the hardware-integration of an Arduino UNO and the PMSA003 particulate matter (PM) sensor, the Arduino UNO was programmed to count six PM diameters: 0.3, 0.5, 1.0, 2.5, 5, and 10 micrometers, for a fixed air volume of 0.1 Liters. Indoor PM data was gathered within 12 different locations at Navajo Preparatory School, Farmington, NM. Outdoor atmospheric PM data was then gathered in Tucson, Arizona, before and after various weather events, such as high winds (PM generating) and rain (PM scrubbing). Additionally, indoor and outdoor data gathered in West Virginia during heavy smoke from the 2023 Canadian forest fires. The final data was gathered throughout the four-corners region of the desert southwest. The output of the Arduino UNO included current, average, maximum, and minimum PM values for each particle size. The Arduino UNO also calculated the least-squares fit a negative-exponential model of the particulate matter as a function of count and particle size, and calculated the correlation "R" between the actual PM count data and the model. Correlations as high as 99.99% were achieved at a confidence of 99.95%. This will help to understand the PM problems on the Navajo Nation, which could include radioactive dust from over 500 abandoned uranium mines.

Keywords: Particulate Matter, Arduino UNO, Negative-Exponential Model, Air Quality, Forest Fire Smoke, Atmospheric Dust

Abbreviations

The following abbreviations are used in this manuscript:

A	Constant in Theoretical Negative Exponential Distribution of Particle Size
DAS	Data Acquisition System
DINÉ	Digitally INtegrated Environmental
DOF	Degrees of Freedom, equal to the sample size N minus 2
IDE	Arduino Integrated Development Environment
N	Sample Size. N=6 particulate matter sizes in all cases in this paper.
PM	Particulate Matter
R	Statistical Correlation
RXD	Receive Data across a Serial Interface
T	Calculation compared to Student-T Distribution
Tau	63.2% percentile particle size in Theoretical Negative Exponential Distribution
TXD	Transmit Data across a Serial Interface

1. Introduction

The assessment of atmospheric particulate matter is important both locally to the Navajo Nation and globally, because unhealthy air is shortening the life span of many people. The American Lung Association and the U.S. Environmental Protection Agency both indicate that particulate matter is a very serious health concern [1-3]. From the Environmental Protection Agency, fine particulate matter (0.1 to 2.5 micrometers in diameter):

- Causes early death (both short-term and long-term exposure)
- Causes cardiovascular harm (e.g., heart attacks, strokes, heart disease, congestive heart failure)
- Likely to cause respiratory harm (e.g., worsened asthma, worsened COPD, inflammation)
- Likely to cause cancer
- Likely to cause harm to the nervous system (e.g., reduced brain volume, cognitive effects)
- May cause reproductive and developmental harm

2. Materials and Methods

The research plan was to instrument an Arduino Uno with a PMS003 Air Monitoring Breakout sensor, Figure 1 [4,5]. The PMSA003, Figure 2, counts particulate matter in six diameters: 0.3, 0.5, 1, 2.5, 5, and 10 µm (micrometers) for a fixed volume of air, namely 0.1 Liters.

In Figure 1, the wires were color coded as follows. Ground was black and +5 volts was red. The digital communications between the Arduino UNO and the PMSA003 particle counter were green for receive (RXD) and blue for transmit (TXD).

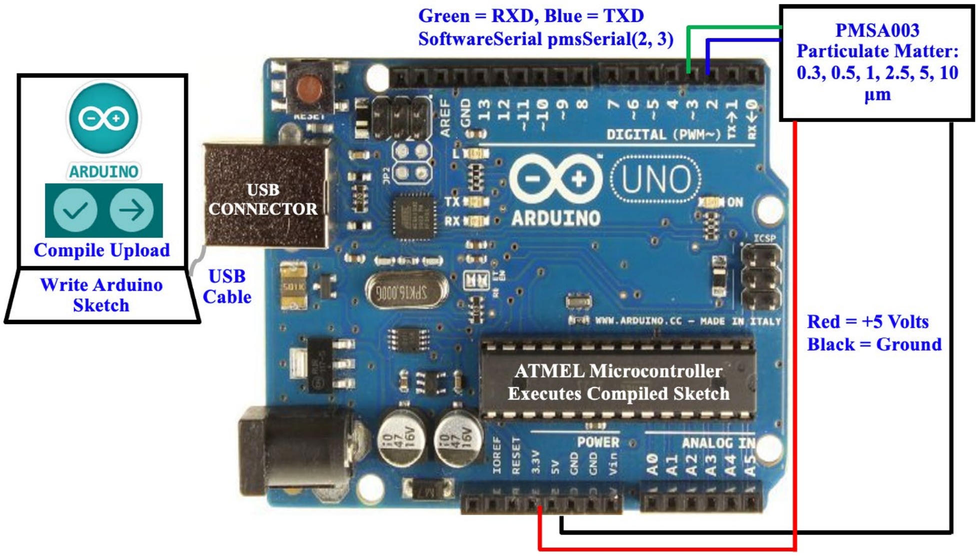


Figure 1: Arduino UNO Tracking 0.3, 0.5, 1, 2.5, 5, 10 µm "PM" Particulate Matter

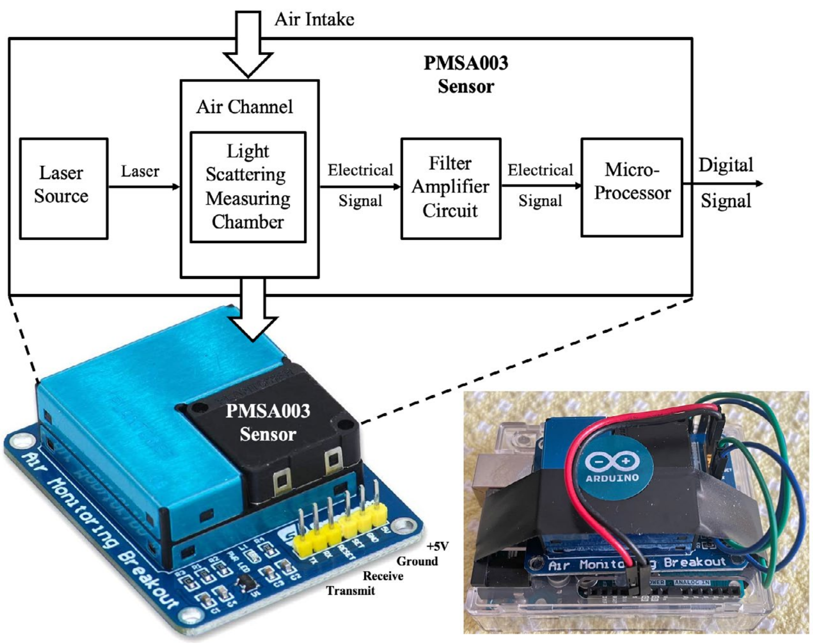


Figure 2: PMSA003 Particulate Matter Sensor Counting 0.3, 0.5, 1, 2.5, 5, 10 µm "PM"

Figures 1 and 2 indicated that three computers were simultaneously linked to gather and process the experimental data, namely an Apple laptop, the Atmel Microcontroller on the Arduino UNO board (Figure 1), and the microprocessor in the PMSA003 Particulate Matter sensor (Figure 2).

The research plan then included programming the Arduino Uno to gather the particulate matter shown in Table 1, and tally the PM counts, then calculate the average, maximum, and minimum of the PM counts. Then a least-squares fit of a negative exponential-distribution was done to the empirical particulate matter counts. Finally, the correlation "R" between the negative-exponential theoretical-model and the empirically measured PM counts was calculated, followed by the calculation of the confidence in that correlation R using the Student t-Distribution. One goal was to measure particulate matter in various locations in New

Mexico, Arizona, West Virginia, and the Four-Corners Region, then to compare indoor particulate matter versus outdoor "free atmosphere" particulate matter. This study included identifying dust generating events in our atmosphere, such as high winds generated by weather fronts, and dust scrubbing events such as gentle rain.

Independent Variables	Dependent Variables
N = 6 different Particulate Matter sizes: 0.3, 0.5, 1, 2.5, 5, 10 µm	Average, Maximum, and Minimum counts of each Particulate Matter size. Least-Squares fit of "Amount A" and Tau (63.2% "percentile") of negative exponential distribution of particulate matter "PM" count versus size in micrometers): $Count = A * e^{-(PMsize - 0.3)/Tau}$. Correlation "R" between Negative Exponential Distribution and actual measured PM counts. Student t-Distribution Confidence: $T = R\sqrt{N-2}/\sqrt{1-R^2}$. Standardized Test Statistic, Z.

Table 1: Independent and Dependent Variables

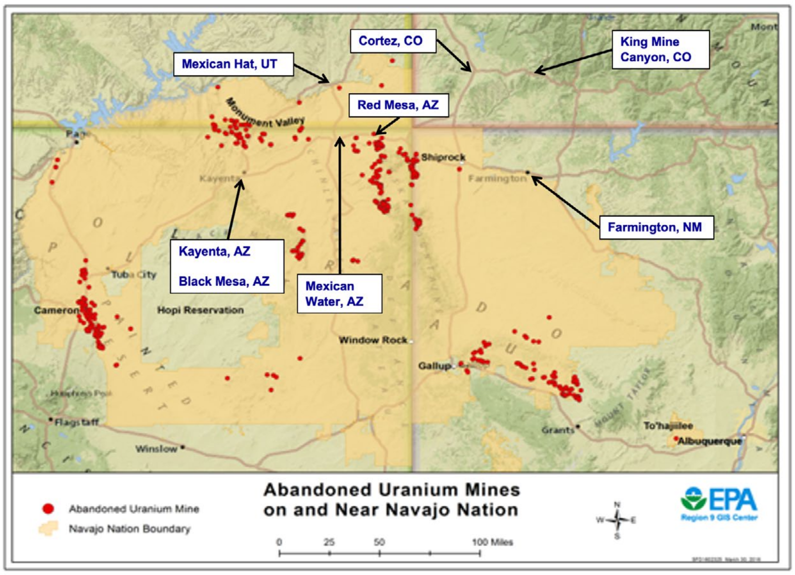


Figure 3: Map of the Navajo Nation [6]

Figure 3 provides a map where particulate matter was measured on the Navajo Nation [6]. The Navajo Nation straddles Arizona, New Mexico, and Utah in the Four-Corners region of the desert southwest.

Figure 3 also details the locations of over 500 Abandoned Uranium Mines (AUMs) on the Navajo Nation. Just east of Gallup, NM was the tragic Church Rock Uranium Spill [7]. The Church Rock Uranium Mill Spill occurred in New Mexico on July 16, 1979, when United Nuclear Corporation's uranium tailings disposal pond at its uranium mill in Church Rock breached its rock dam. The "accident" remains the largest release of radioactive material in U.S. history, having released more radioactivity than the Three Mile Island accident four months earlier. The uranium ores present in the tailings were already made mobile by the same acids that ate through the rock retaining wall and allowed for a historic flood of radioactive material into the Navajo Nation. The released radioactivity combined with the area soil, resulting in mobile radioactive dust. Our reservation has 523 abandoned uranium mines (AUM) that are both remnants of the cold-war era and currently EPA Superfund Cleanup sites [8].

Additionally, wood fired stoves in Hogans can release large amounts of particulate matter, due to incomplete combustion. Thus, it is critical to better understand the particulate matter across the Navajo Nation.

Particulate Matter sizes were collected via the Arduino Uno which executed the Arduino sketch, which is listed in Appendix A. A flowchart of the data acquisition programming is shown in Figure 4, on the next page. The data acquisition program begins with the initialization of key parameters. First of all, the Sample_Number is initialized to 1. Then, for each particulate matter size, the maximum count is set to zero, the minimum count is set to 10000, and the Count_Sum is set to 0.

The next step was the PMSA sensor counting particulate matter of the six sizes 0.3, 0.5, 1, 2.5, 5, and 10 µm. This process continued within the PMSA sensor until a volume of air of 0.1 Liters was reached, then the counts for that 0.1 Liter volume of air were transmitted to the Arduino UNO.

It may seem counterintuitive to initialize the maximum to a low number such as zero, but when samples come in with counts exceeding the current maximum, the maximum is reset to the value of that higher count, by the Arduino code executed within the Atmel microcontroller of the Arduino UNO.

Similarly, it may seem counterintuitive to initialize the minimum to a really high number such as 10000, but when samples come in with counts smaller than the current minimum, the minimum is reset to the value of that lower count, by the Arduino code executed within the Atmel microcontroller of the Arduino UNO. Thus, the Arduino UNO individually tracks the maximums and minimums of the six different particulate matter sizes, and adjusts those maximums and minimums as needed.

The New_Count information is added to the previous Count_Sum to create an updated Count_Sum. When the updated Count_Sum is divided by the Sample_Number, the average count for that particular particulate matter size is calculated.

Then, the Arduino code within the Arduino UNO does a least-squares fit of a theoretical model within the Atmel microcontroller of the Arduino UNO. This theoretical model is a Negative Exponential Distribution: Theoretical Count = $A * e^{-(PMsize - 0.3)/Tau}$.

After the least-squares fit, the Arduino code calculates the Correlation "R" between actual PM Count Distribution "X" and the Theoretical Negative Exponential Distribution "Y" with this equation, [9]:

$$R = [N * \sum XY - (\sum X)(\sum Y)] / \sqrt{[N * \sum X^2 - (\sum X)^2][N * \sum Y^2 - (\sum Y)^2]}$$

Finally, the Arduino code calculates the t-Distribution for "Confidence:"

$$T = R\sqrt{N-2}/\sqrt{1-R^2}$$

The "confidence" is a measure of the "believability" of the correlation R. The value of T is compared by the Arduino code to a tabulated "test" value of 8.61, based on a desired 99.95% confidence and 4 degrees-of-freedom [9]. The 4 degrees-of-freedom is calculated from N=6 particulate matter sizes (0.3, 0.5, 1.2.5, 5, 10) from which 2 is subtracted. The number 2, which was subtracted from N=6 to get the degrees-of-freedom, represents the "Amount A" and Tau "63.2% percentile" constants which were least-squares fit by the Arduino code executed by the Arduino UNO.

The calculations done by the Arduino UNO for correlation "R" and the Student t-Distribution were confirmed by the use of Excel.

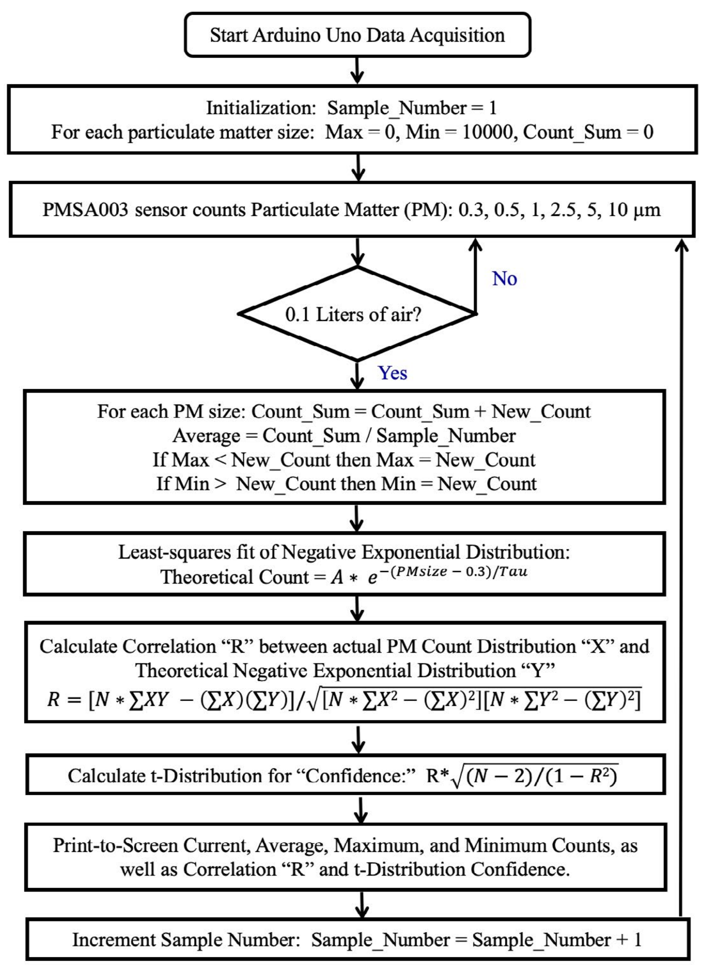


Figure 4: Flow Chart of Data Acquisition and Post-Processing

3. Results

The PMSA003 particulate matter sensor counts particles of the sizes 0.3, 0.5, 1. 2.5, and 10 micrometers for an air volume of 0.1 Liters (100cc). The averages, maximums, and minimums of the above variables are calculated, as well as the least-squares fit of the theoretical Negative Exponential Distribution, the correlation "R" between the theoretical Negative Exponential Distribution, and the confidence in that correlation "R," before all are printed to the screen of the attached laptop. The number of that sample are also printed to the laptop screen, as shown in Figure 5. By clicking on the serial monitor icon in the upper right corner and selecting 115200 baud, the output display was shown.

```
-----------------------------------Sample: 430.00----------------------------------------
Current Particles > 0.3um / 0.1L air: 14307    Ave: 11490.29  Max: 15582.00  Min: 6822.00
Current Particles > 0.5um / 0.1L air: 4659     Ave: 3724.93   Max: 4960.00   Min: 2204.00
Current Particles > 1.0um / 0.1L air: 1641     Ave: 1222.95   Max: 1704.00   Min: 747.00
Current Particles > 2.5um / 0.1L air: 175      Ave: 123.94    Max: 214.00    Min: 66.00
Current Particles > 5.0um / 0.1L air: 32       Ave: 25.41     Max: 68.00     Min: 11.00
Current Particles > 10.0um / 0.1L air: 13      Ave: 8.86      Max: 32.00     Min: 1.00

Least-Squares Average Particulate-Matter Count = A*exp(-(PM_size - 0.3)/tau)
A: 11490.29   tau: 0.18       R: 99.97 % Correlation   Student-T: 79.45      99.95% Confidence
------------------------------------------------------------------------------------------
WARNING: EXCESSIVE PARTICULATE MATTER
------------------------------------------------------------------------------------------
```

Figure 5: Actual Data Displayed by the Arduino UNO (Outdoor Measurement, Tables 6-7, July 29)

After the above is example data was displayed, the Sample Number is incremented by one and the processes cycles back to gather a new set of data from the PMSA003 Particulate Matter Sensor. Each new sample from the PMSA003 Particulate Matter Sensor updates the averages, maximums, minimums, correlation "R," and Student t-Distribution. The lead-author gathered the above "Canadian-Fire Smoke" data in West Virginia, during the summer-2023 Native Youth Climate Adaptation Leadership Conference (NYCALC). The AI (artificial intelligence) warning threshold in the Arduino UNO code was adjustable and could be selectively lowered for people with COPD (Chronic Obstructive Pulmonary Disease), asthma, allergies, etc.

3.1. Indoor Data – Navajo Preparatory School, Farmington, NM

Figure 6 shows an example graph of particle count in 0.1 Liters of air versus particle size in micrometers, gathered inside the Bahe Dorm Room, Table 2, of Navajo Preparatory School. Note the negative-exponential-like distribution of particle count versus particle size. The highest maximum, average, and minimum were for the 0.3 micrometer particles.

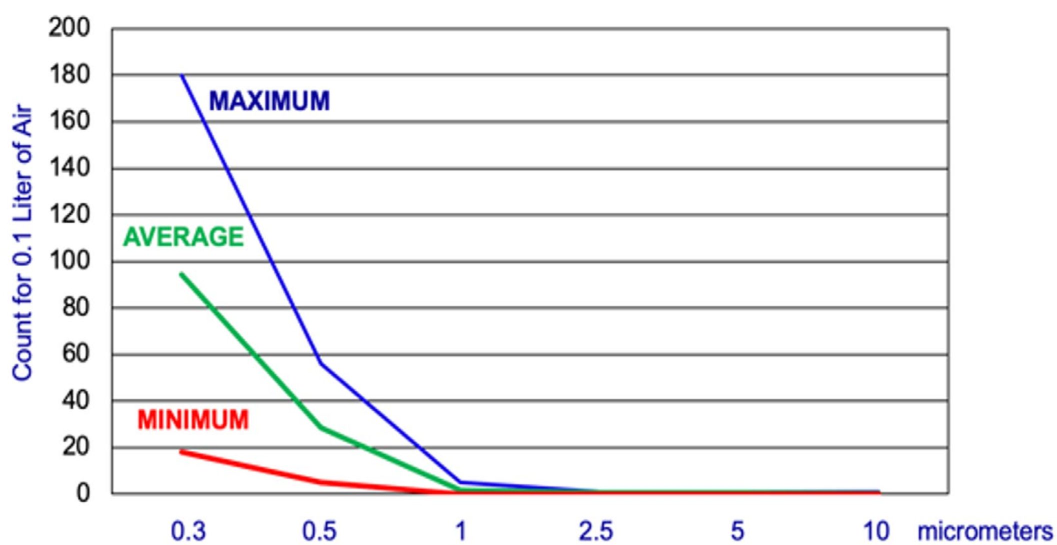


Figure 6: Graph of Particle Count vs Particle Size, Bahe Dorm Room, Navajo Preparatory School

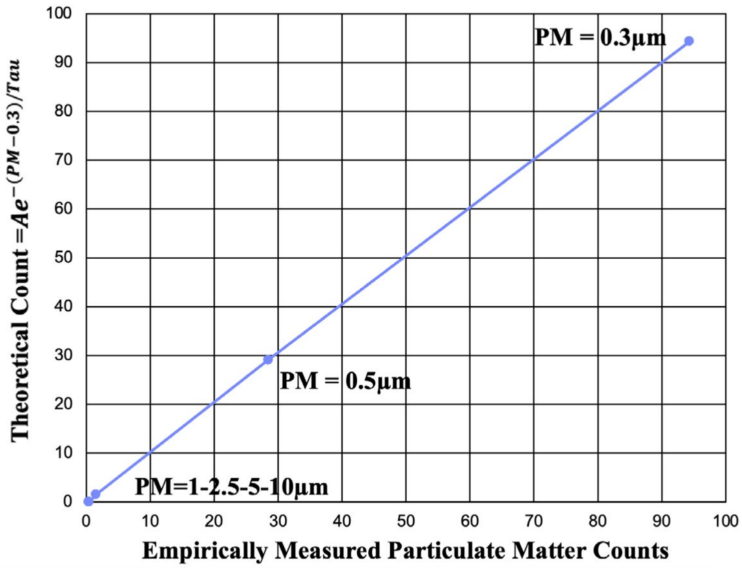


Figure 7: Negative Exponential Curve Fit vs Measured Particle Matter Count: Bahe Dorm Room

The equation used to model the average particle count in Figure 7 was that of a negative exponential curve fit of the form: Theoretical Count = $A * e^{-(PMsize - 0.3)/Tau}$. The 0.3 term in the numerator of the exponential function was the smallest particle size detected. Using the =PEARSON function in EXCEL, a 99.99% correlation (100% is a perfect correlation) was calculated for the least- squares values of A = 94.33 and TAU = 0.17 micrometers This least-squares calculation shows that the 63.2% (TAU) percentile constant is 0.17 micrometers, close to the wavelength of Ultraviolet-C, Figure 8. Note: the argument of the exponential function needs to be unitless, and it is, because micrometers occur both in the numerator and denominator (µm/µm). Graphing the count from the negative exponential curve in Figure 7 shows a nearly straight line (nearly perfect correlation).

Light: Color and Wavelengths	Particle Sizes
Infrared-C 3000 - 10000nm Infrared-B 1400 - 3000nm Infrared-A 740 - 1400nm	5µm and 10µm 2.5µm (2500nm) 1µm (1000nm)
Red 625 - 740nm Orange 590 - 625nm Yellow 565 - 590nm Green 500 - 565nm Blue 450 - 500nm Violet 400 - 450nm	0.5µm (500nm)
Ultraviolet-A 315 - 400nm Ultraviolet-B 280 - 315nm Ultraviolet-C 190 - 280nm	0.3µm (300nm)

Figure 8: Particle Sizes Versus Wavelengths of Light

Table 2, below, lists the average particle count for six different particle sizes for all 12 indoor locations where data was obtained at Navajo Preparatory School. Clearly, the Zah Living Room, with its dust-retaining carpeted floor, had an astoundingly large amount of particulate matter contamination, as compared to all other areas which did not have carpeting.

Indoor Navajo Preparatory School	PM 0.3	PM 0.5	PM 1.0	PM 2.5	PM 5	PM 10
Bahe Dorm Room	94.33	28.46	1.54	0.46	0.46	0.35
Zah Living Room Carpeted	3788.41	1148.08	132.73	15.37	4.43	1.8
Wolfe's Room	121.50	40.17	19.83	7.50	1.00	1.00
Recreation Room	171.45	51.70	2.97	0	0	0
Manliteo Room	472.50	149.71	26.92	12.54	2.67	1.88

Library	906.47	291.02	83.04	20.33	13.22	7.18
Hogan	907.26	291.30	82.76	20.26	13.26	7.22
Cafeteria Heater	805.12	242.09	11.79	1.81	0.77	0.30
Chemistry Room	876.27	262.82	29.82	0	0	0
Dorm Living Room	115.29	30.71	0.76	0.76	0.76	0.76
Bates Living Room	820.25	230.77	2.72	0.40	0.32	0.32
Arthur Hall	145.05	45.70	12.26	6.88	0.86	0.86

Table 2: Average Particulate Count For N=6 Particle Sizes, 12 Indoor Sites

Table 3, below, lists least-squares fit of TAU (63.2% percentile particle size, column 3) and the statistical correlations "R" (column 4) of the fit of the theoretical model of particle count: Theoretical Count = $A * e^{-(PMsize\,-\,0.3)/Tau}$ and the empirical (experimental) data. The statistical correlations" R" were amazingly high, between 99.08% and 99.99%. The confidence (column 5) is based on the T- Distribution for a sample size of N=6 different particle sizes, which gives 4 Degrees-Of-Freedom (DOF = N-2 = 4).

In Table 3, the Degrees-Of-Freedom are 2 less than N=6 because there are two parameters in the theoretical model (Amount "A" and "Tau"). Calculated values of the T-Distribution (column 5) had a confidence of 99.95% confidence (T-test value = 8.61, [9]) for all twelve indoor sites.

Indoor Navajo Preparatory School	Constant "A" $Ae^{-(PM-0.3)/Tau}$	Constant "Tau" $Ae^{-(PM-0.3)/Tau}$	Statistical Correlation "R"	99.95% Confidence N-2= 4 DOF
Bahe Dorm Room	94.33	0.17 µm	99.99%	197.01 > 8.61
Zah Living Room Carpeted	3,788.41	0.16 µm	99.97%	85.25 > 8.61
Wolfe's Room	121.50	0.17 µm	99.08%	14.68 > 8.61
Recreation Room	171.45	0.17 µm	99.99%	278.99 > 8.61
Manliteo Room	472.50	0.17 µm	99.93%	56.93 > 8.61
Library	906.47	0.17 µm	99.81%	32.07 > 8.61
Hogan	907.26	0.17 µm	99.81%	32.28 > 8.61
Cafeteria Heater	805.12	0.17 µm	99.99%	229.80 > 8.61
Chemistry Room	876.27	0.17 µm	99.97%	95.55 > 8.61
Dorm Living Room	115.29	0.15 µm	99.99%	219.41 > 8.61
Bates Living Room	820.25	0.17 µm	99.99%	197.60 > 8.61
Arthur Hall	145.05	0.17 µm	99.99%	30.89 > 8.61

Table 3: Constants A and Tau, Correlation R, Confidence for Navajo Preparatory School

3.2. Outdoor Data – Tucson, Arizona Wind and Rain

Table 4, below, lists the average particle count for six different particle sizes for 9 different dates as measured in Tucson, Arizona. Table 5 then lists the least-squares fit of Amount "A," Tau, correlation "R," and the confidence in that correlation. Note how Tau (63.2% percentile) value in Tables 3 and 5 are nearly equal.

Tucson, Arizona	PM 0.3	PM 0.5	PM 1.0	PM 2.5	PM 5	PM 10
May 8	218.82	67.76	4.57	0.57	0.57	0.57
May 9	199.98	63.10	19.06	6.61	0.75	0.75
May 10 red flag day	395.90	126.22	20.35	3.38	2.92	1.52
May 11	378.97	121.16	8.94	0	0	0
May 13	494.51	159.57	16.22	0.96	0.96	0.96
May 14	355.18	109.51	9.20	0.63	0.43	0.43
May 16 wind	620.44	196.17	20.44	0.44	0	0
May 19 5mm rain	174.3	54.10	3.8	0	0	0
May 20	292.36	107.0	8.27	0	0	0

Table 4: Average Particulate Count For N=6 Particle Sizes, 9 Outdoor 2023 Dates

Outdoor Tucson, Arizona 2023	Constant “A” $Ae^{-(PM-0.3)/Tau}$	Constant “Tau” $Ae^{-(PM-0.3)/Tau}$	Statistical Correlation “R”	99.95% Confidence N-2 = 4 DOF
May 8	218.82	0.17 µm	99.99%	130.44 > 8.61
May 9	199.98	0.17 µm	99.93%	53.18 > 8.61
May 10 red flag day	395.90	0.17 µm	99.98%	94.55 > 8.61
May 11	378.97	0.17 µm	99.98%	94.15 > 8.61
May 13	494.51	0.18 µm	99.98%	101.87 > 8.61
May 14	355.18	0.17 µm	99.98%	127.16 > 8.61
May 16 wind	620.44	0.17 µm	99.98%	104.32 > 8.61
May 19 5mm rain	174.3	0.17 µm	99.99%	121.31 > 8.61
May 20	292.36	0.17 µm	99.99%	125.50 > 8.61

Table 5: Constants A and Tau, Correlation R, Confidence for Outdoor Tucson, Arizona

3.3. West Virginia Smoky Air

While the lead author attended the 2023 Native Youth Climate Adaptation Leadership Congress (NYCALC, through the Bureau of Indian Affairs) in West Virginia, she gathered the particulate matter data shown in Table 6, during a particularly smoky day. This pervasive smoke was from the Canadian wildfires. The data in Table 6 dwarfed the New Mexico data in Table 3 and the Arizona data in Table 5.

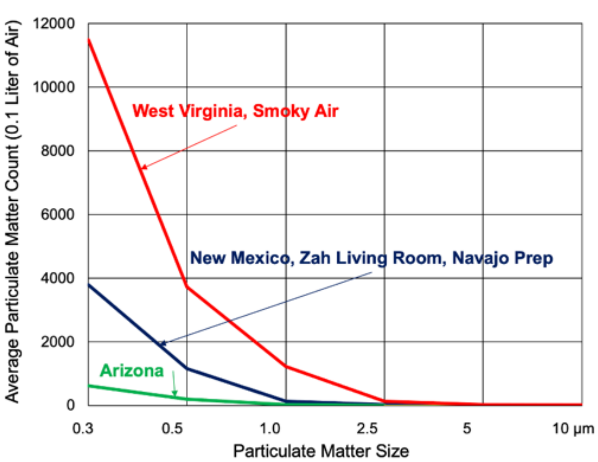
Table 7 then lists the least-squares fit of Amount “A,” Tau, correlation “R,” and the confidence in that correlation. Interestingly enough, Tables 3, 5, and 7 all had Tau (63.2% percentile) values between 0.15 and 0.18. The outdoor measurements were repeated 24 hour later, showing more than a 3X drop in particulate matter. Figure 9 shows how much worse West Virginia was versus New Mexico and Arizona, as well as how difficult it was to see the sun during daytime.

West Virginia 29 June 2023	PM 0.3	PM 0.5	PM 1.0	PM 2.5	PM 5	PM 10
Smoky indoor	10672.2	3459.9	1120.0	108.1	21.5	7.0
Smoky outdoor	11490.3	3724.9	1222.9	123.9	25.4	8.9
Outdoor+24 hours	3581.8	1157.6	334.8	25.6	5.3	1.9

Table 6: Average Particulate Count For N=6 Particle Sizes, Smoky Air Data, West Virginia

West Virginia 29 June 2023	Constant “A” $Ae^{-(PM-0.3)/Tau}$	Constant “Tau” $Ae^{-(PM-0.3)/Tau}$	Statistical Correlation “R”	99.95% Confidence N-2 = 4 DOF
Smoky Indoor	10672.2	0.18 µm	99.97%	80.62 > 8.61
Smoky Outdoor	11490.3	0.18 µm	99.97%	79.45 > 8.61
Outdoor+24 hours	3581.8	0.18 µm	99.97%	86.28 > 8.61

Table 7: Constants A and Tau, Correlation R, Confidence for Smoky Air Data, West Virginia



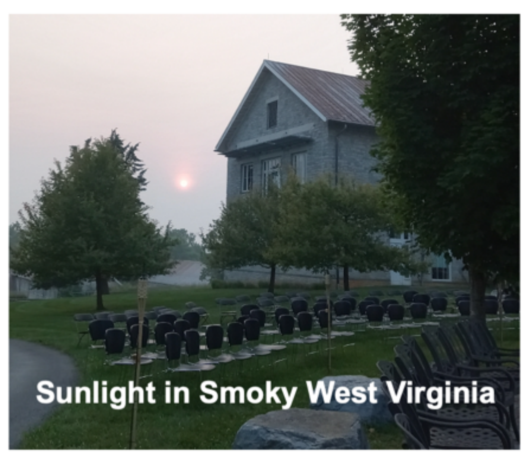


Figure 9: Smoky Air in West Virginia Worse Than New Mexico And Arizona

3.4. Four-Corners Region

Table 8 lists the average particle count for six different particle sizes for eight different sites in the Four-Corners Region. The Four-Corners Region is formed by the borders of the states of Colorado (CO), Utah (UT), Arizona (AZ), and New Mexico (NM). Table 9 then lists the least-squares fit of Amount "A," Tau, correlation "R," and the confidence in that correlation. This data was taken in 2024.

Four-Corners Region	PM 0.3	PM 0.5	PM 1.0	PM 2.5	PM 5	PM 10
Cortez, CO	146.14	40.37	1.02	0.48	0.23	0.13
King Mine Canyon, CO	142.41	43.86	2.63	0.32	0.20	0.10
Mexican Hat, UT	105.48	31.14	0.46	0.19	0.03	0.01
Black Mesa, AZ	277.46	88.24	14.40	3.64	2.35	1.39
Kayenta, AZ	2160.95	684.58	84.53	4.67	2.79	1.47
Mexican Water, AZ	25.25	6.99	0.25	0.17	0.15	0.12
Red Mesa, AZ	145.90	46.35	11.98	1.16	0.47	0.19
Hogan, Farmington, NM	521.10	163.60	14.88	1.47	1.10	0.52

Table 8: Average Particulate Count for N=6 Particle Sizes, Four-Corners Region

Four-Corners Region	Constant "A" $Ae^{-(PM-0.3)/Tau}$	Constant "Tau" $Ae^{-(PM-0.3)/Tau}$	Statistical Correlation "R"	99.95% Confidence N-2 = 4 DOF
Cortez, CO	146.14	0.15 µm	99.99%	118.40 > 8.61
King Mine Canyon, CO	142.41	0.17 µm	99.99%	121.68 > 8.61
Mexican Hat, UT	105.48	0.16 µm	99.99%	119.45 > 8.61
Black Mesa, AZ	277.46	0.17 µm	99.98%	89.99 > 8.61
Kayenta, AZ	2160.95	0.17 µm	99.98%	102.82 > 8.61
Mexican Water, AZ	25.25	0.16 µm	99.99%	119.00 > 8.61
Red Mesa, AZ	145.90	0.17 µm	99.98%	92.40 > 8.61
Hogan, Farmington, NM	521.10	0.17 µm	99.98%	110.38 > 8.61

Table 9: Constants A and Tau, Correlation R, Confidence for Four-Corners Region

4. Discussion

It is interesting to compare "Tau," the 63.2% percentile particle size, between the New Mexico measurements of Table 3, the Arizona measurements of Table 5, the West Virginia measurements of Table 7, and the Four-Corners Region measurements of Table 9. Tau was never empirically measured. Instead, Tau is mathematically calculated by least-squares fit in Tables 3, 5, 7, and 9. The units of Tau are µm (particle size) and Tau is smaller than the smallest measured particle of 0.3µm, in all cases. This can be explained via Figure 10, which shows that the percentage of 0.3µm is 69% for the Navajo Hogan, which exceeds the definition of Tau as the 63.2% percentile. Thus, Tau is understandably smaller than 0.3µm.

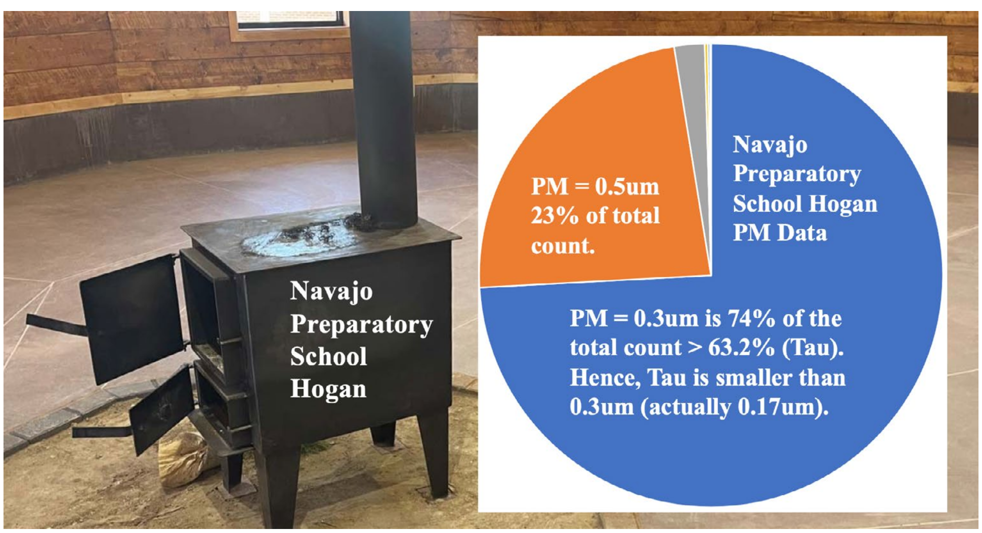


Figure 10: Pie Chart of Particle Count vs Size (Table 2), for Navajo Preparatory School Hogan [10]

Assuming Gaussian Distributions for "Tau," the average of "indoor Tau" in Table 3 is 0.1675µm and the average of "outdoor Tau" in Table 5 was 0.1711µm, such as calculated by the example Excel function =AVERAGE (B2:B10). The standard deviation of "indoor Tau" in Table 3 is 0.0062µm and the standard deviation of "outdoor Tau" in Table 5 was 0.0033µm, such as calculated by the example Excel function =STDEV.S(B2:B10). These parameters can be applied to the Standardized Normal Variate Test Statistic" Z." Assuming there was no difference between the average indoor Tau and the outdoor Tau, the following equation was used, [11]:

Test Value $Z = [TAU2 - TAU1]/\sqrt[2]{Denominator}$
Denominator = $SD1^2/M1 + SD2^2/M2$
Where: TAU1 = average indoor Tau = 0.1675µm, TAU2 = average outdoor Tau = 0.1711µm
SD1 = standard deviation of indoor Tau = 0.0062µm,
SD2 = standard deviation of outdoor Tau = 0.0033µm
M1 = 12 indoor samples from Table 3, and M2 = 9 outdoor samples from Table 5

The units of Test Value Z are µm/µm, which means that Z is unitless, as desired. The value of Z from the above information is Z = 1.711, which equates to α/2 = 5% [12]. Given this is a two-tailed test, this gives α = 10%, or a 90% confidence that the indoor and outdoor values of Tau are statistically equal.

The next item which is of interest is that the "Amount A" can be much larger for Table 3 (indoor measurements) than Table 5 (outdoor measurements). The explanation is that indoor rooms can act as "dust accumulators." This indicates that indoor rooms need to be kept clean for respiratory health. An example is the Bahe Dorm Room, with the lowest "Amount A" of all samples, which shows the importance of cleanliness and neatness.

The final item of interest is shown in Figure 11. The left graph shows how high velocity winds during a "red flag warning" clearly lift a lot of dust into the air. However, the right graph shows how a gentle rain cleanses particulate matter out of the outdoor air. The time axis in each graph in Figure 11 is arranged to show the lower PM counts in the forefront, to keep the higher PM counts from obscuring the lower PM counts.

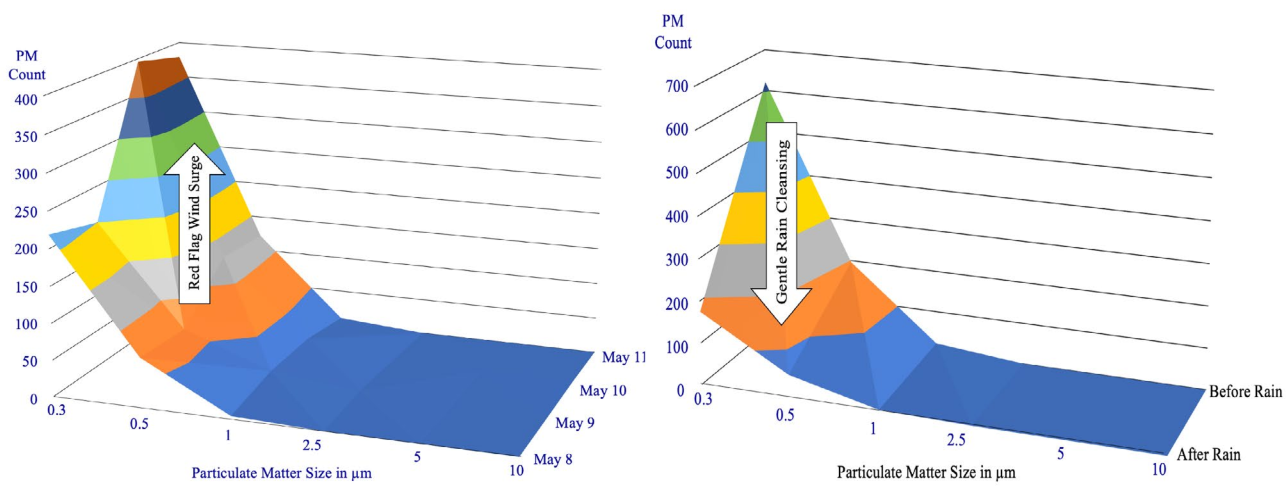


Figure 11: Red Flag Winds Cause PM Surge, then Gentle Rains Cleanse the Air

5. Conclusions

The Arduino DAS (Data Acquisition System) shown in Figure 1 only cost $77 (Arduino Uno $27, PMSA003 Particulate Matter sensor $52). Thus, a personal goal of a DAS for under $100 was met.

The average particle counts shown in Tables 2, 4, 6, and 8 were individually modeled by a negative exponential distribution, and this theoretical model had a correlation "R" of 99.08% - 99.99% with the experimental data, Tables 3, 5, 7, and 9. This proved that it was possible to model the distribution of particulate matter sizes. Both the theoretical model and the experimental data showed that the 0.3 particle size was dominant, which makes sense. Assuming that the 10 and 0.3 micrometer particle sizes have the same mass density, the ratio of the masses of the two particles is a function of radius or diameter to the third power (the volume of a sphere). This mass ratio is $(10/0.3)^3$ which shows that the 10-micrometer particle has 37,000 times the mass of the 0.3-micrometer particle. Hence, the 10-micrometer "coarse" particles would tend to settle out more readily and the 0.3-micrometer "fine" particles would tend to stay airborne.

For all locations tested, that the "63.2% percentile" Tau varied between 0.15 and 0.18 micrometers, and that the indoors Tau was statistically equal to the outdoor Tau. Figure 10 showed that the percentage of 0.3µm particulate matter 69% exceeded 63.2% (Tau) for the Hogan, which explains why Tau was smaller than 0.3µm in Tables 2, 4, 6, and 8.

Based on the above, our hypotheses were accepted, namely that we could create a data acquisition system for under $100, that we could program this DAS and use it to gather actual data, and that we could curve fit a theoretical model to the actual particulate matter data to an extremely high statistical correlation.

The importance of this research is that the “fine” particulate matter is highly mobile. Fine (0.3 um) radioactive dust, dust from farmers plowing their fields, and particulate matter from forest fires could remain in the air for a long time, thus causing the harmful effects documented by the U.S. Environmental Protection Agency. This is highly relevant to the Navajo Nation.

Follow-on research could include putting heat exchangers in the exhaust pipes of wood-burning stoves in Hogans, so that less wood needs to be burned and hence less particulate matter and less carbon monoxide produced.

Our recommendations for dust control include the use of soil stabilizers on agricultural land, and dirt roads on the Navajo Nation. Dust-launching leaf blowers would be taken off the market, and people would go back to using brooms. Within structures, such as the classrooms and dorm rooms at Navajo Preparatory School, frequent changes of the air filters in the air handling systems are recommended, as well as the use of Honeywell HPA300 HEPA (High-Efficiency Particulate Air) Air Purifiers in classrooms [13]. These HEPA filters remove 99.97% of particulate matter which are 0.3 micrometers or larger in size. As our global climate warms, harmful particulate matter will grow as a problem, which means we must act now to reverse climate warming. In 2021, life expectancy for Native Americans was 65 years; or black Americans, 71; for white Americans, 76; for Hispanic Americans, 78; and 84 for Asian Americans [14]. It is our hope that this study will contribute to lengthening the life expectancy of members of the Navajo Nation.

This study used an Arduino UNO R3. In the future, the newly available Arduino UNO R4 WiFi could be used to send data to the cloud [15]. Once in the cloud, the data could be blockchained with a date-time stamp, as discussed [16].

References

1. American Lung Association, *Particle Pollution*.
2. U.S. Environmental Protection Agency. *Integrated Science Assessment (ISA) for Particulate* Matter (Final Report, December, 2019).
3. U.S. Environmental Protection Agency. *Supplement to the 2019 Integrated Science Assessment for Particulate Matter* (*Final Report, 2022*).
4. Arduino® UNO R3
5. Zhendong, Zhao, et-al., *Product Specification, Particulate Matter Sensor, PMSA003-A*. (2019).
6. *Abandoned Uranium Mines on or near the Navajo Nation*. United States Environmental Protection Agency.
7. Heywood, Sam. *Tragedy at Church Rock: Superfunds and the Marginalization of Navajos.*
8. *Navajo Nation: Cleaning Up Abandoned Uranium Mines*. United States Environmental Protection Agency. Last Updated February 14, 2025.
9. CRC Standard Mathematical Tables, 20th edition, published by the Chemical Rubber Company.
10. Hogan Grand Opening. Navajo Preparatory School. Used by permission.
11. Devore, J. (1995). Probability and Statistics for Engineering and the Sciences, Fourth. *New York, NY: Brooks/Cole*.
12. ibid., Standard Normal Curve Areas, pp. 705.
13. Honeywell HPA300 HEPA Air Purifier.
14. Race and Health in the United States.
15. Arduino® UNO R4 WiFi
16. Winarski, T. (2023). TSUNAMI OF BLOCKCHAIN TECHNOLOGY AND PATENTS: A STRATEGIC DATA-ANALYTIC OVERVIEW. *Jurimetrics: The Journal of Law, Science & Technology, 64*(1).

Appendix A: Arduino Sketch

Our Arduino Uno sketch had 211 lines-of-code. It was compiled on an Apple laptop in the Arduino IDE and then uploaded it to the Arduino UNO for execution, as shown in Figure 1. The following code is in a font compatible with the Arduino IDE compiler.

```
/* DINE — Digitally INtegrated Environmental Arduino Platform 2.0  */
#include <SoftwareSerial.h>
//define pin data
SoftwareSerial pmsSerial(2, 3);
   float sample = 1.0;  //Tally number of samples in order to calculate averages
   float sum_03um = 0.0; float sum_05um = 0.0; float sum_10um = 0.0;
   float sum_25um = 0.0; float sum_50um = 0.0; float sum_100um = 0.0;
   float average_03um = 0.0; float average_05um = 0.0; float average_10um = 0.0;
   float average_25um = 0.0; float average_50um = 0.0; float average_100um = 0.0;
   float max_03um = 0.0; float max_05um = 0.0; float max_10um = 0.0;
   float max_25um = 0.0; float max_50um = 0.0; float max_100um = 0.0;
   float min_03um = 10000.0; float min_05um = 10000.0; float min_10um = 10000.0;
   float min_25um = 10000.0; float min_50um = 10000.0; float min_100um = 10000.0;
   float sse = 0.0; float sse_min = 1000000.0;
   float R = 0.0; float R2 = 0.0;  // Variables used for correlation "R"
   float tau = 0.3; float tau_save = 0.0; float A = 0.0; float T;
   float A_save=0.0;
   float X1 = 0.0; float X2 = 0.0; float X3 = 0.0; float X4 = 0.0;
   float X5 = 0.0; float X6 = 0.0;
   float Y1 = 0.0; float Y2 = 0.0; float Y3 = 0.0; float Y4 = 0.0;
   float Y5 = 0.0; float Y6 = 0.0;
   float sumx = 0.0; float sumy = 0.0; float sumxy = 0.0;
   float sumx2 = 0.0; float sumy2 = 0.0;
void setup() {
  Serial.begin(115200);     // output
  pmsSerial.begin(9600);    // sensor baud rate is 9600
}
struct pms5003data {
  uint16_t framelen;
  uint16_t pm10_standard, pm25_standard, pm100_standard;
  uint16_t pm10_env, pm25_env, pm100_env;
  uint16_t particles_03um, particles_05um, particles_10um, particles_25um,
particles_50um, particles_100um;
  uint16_t unused;
  uint16_t checksum;
};
struct pms5003data data;

void loop() {
/* Output */
  if (readPMSdata(&pmsSerial)) {
    // reading data was successful!
    sum_03um = sum_03um + data.particles_03um; sum_05um = sum_05um +
data.particles_05um;
    sum_10um = sum_10um + data.particles_10um; sum_25um = sum_25um +
data.particles_25um;
```

```
    sum_50um = sum_50um + data.particles_50um; sum_100um = sum_100um +
data.particles_100um;
    average_03um = sum_03um/sample; average_05um = sum_05um/sample;
    average_10um = sum_10um/sample; average_25um = sum_25um/sample;
    average_50um = sum_50um/sample; average_100um = sum_100um/sample;
    if (max_03um < data.particles_03um) {
    max_03um = data.particles_03um;
    }
    if (max_05um < data.particles_05um) {
    max_05um = data.particles_05um;
    }
    if (max_10um < data.particles_10um) {
    max_10um = data.particles_10um;
    }
    if (max_25um < data.particles_25um) {
    max_25um = data.particles_25um;
    }
    if (max_50um < data.particles_50um) {
    max_50um = data.particles_50um;
    }
    if (max_100um < data.particles_100um) {
    max_100um = data.particles_100um;
    }
    if (min_03um > data.particles_03um) {
    min_03um = data.particles_03um;
    }
    if (min_05um > data.particles_05um) {
    min_05um = data.particles_05um;
    }
    if (min_10um > data.particles_10um) {
    min_10um = data.particles_10um;
    }
    if (min_25um > data.particles_25um) {
    min_25um = data.particles_25um;
    }
    if (min_50um > data.particles_50um) {
    min_50um = data.particles_50um;
    }
    if (min_100um > data.particles_100um) {
    min_100um = data.particles_100um;
    }

    Serial.println();
    Serial.print("------------------------------------Sample: ");
Serial.print(sample);
    Serial.println("------------------------------------------");
    Serial.print("Current Particles > 0.3um / 0.1L air:   ");
Serial.print(data.particles_03um);
    Serial.print("\t  Ave: "); Serial.print(average_03um);
    Serial.print("\t  Max: "); Serial.print(max_03um);
```

```
    Serial.print("\t  Min: "); Serial.println(min_03um);
    Serial.print("Current Particles > 0.5um / 0.1L air:   ");
Serial.print(data.particles_05um);
    Serial.print("\t  Ave: "); Serial.print(average_05um);
    Serial.print("\t  Max: "); Serial.print(max_05um);
    Serial.print("\t  Min: "); Serial.println(min_05um);
    Serial.print("Current Particles > 1.0um / 0.1L air:   ");
Serial.print(data.particles_10um);
    Serial.print("\t  Ave: "); Serial.print(average_10um);
    Serial.print("\t  Max: "); Serial.print(max_10um);
    Serial.print("\t  Min: "); Serial.println(min_10um);
    Serial.print("Current Particles > 2.5um / 0.1L air:   ");
Serial.print(data.particles_25um);
    Serial.print("\t  Ave: "); Serial.print(average_25um);
    Serial.print("\t  Max: "); Serial.print(max_25um);
    Serial.print("\t  Min: "); Serial.println(min_25um);
    Serial.print("Current Particles > 5.0um / 0.1L air:   ");
Serial.print(data.particles_50um);
    Serial.print("\t  Ave: "); Serial.print(average_50um);
    Serial.print("\t  Max: "); Serial.print(max_50um);
    Serial.print("\t  Min: "); Serial.println(min_50um);
    Serial.print("Current Particles > 10.0um / 0.1L air:  ");
Serial.print(data.particles_100um);
    Serial.print("\t  Ave: "); Serial.print(average_100um);
    Serial.print("\t  Max: "); Serial.print(max_100um);
    Serial.print("\t  Min: "); Serial.println(min_100um);
    Serial.println("  ");
    Serial.print("Least-Squares Average Particulate-Matter Count = A*exp(-(PM_size -
0.3)/tau)");
    Serial.println("  ");
    /* Calculate the minimum of the Sum of Squares of Errors SSE here.
    Counter ii:  from 0.1 microns to 10 microns in increments of 0.01 microns
    Divide ii by 100 to get the trial "tau"
    Assumes a negative-exponential distribution of particulate matter.   */
    sse_min = 100000.0;
    for (int ii = 10; ii <= 1000; ii++) {
      tau = ii/100.0;
      A = average_03um;
      X1 = average_03um; X2 = average_05um; X3 = average_10um;
      X4 = average_25um; X5 = average_50um; X3 = average_100um;
      Y1 = A; Y2 = A*exp(-0.2/tau); Y3 = A*exp(-0.7/tau);
      Y4 = A*exp(-2.2/tau); Y5 = A*exp(-4.7/tau); Y6 = A*exp(-9.7/tau);
      sse = sq(Y1-X1) +sq(Y2-X2) +sq(Y3-X3) +sq(Y4-X4) +sq(Y5-X5) +sq(Y6-X6);
      if (sse_min > sse) {
        sse_min = sse;
        tau_save = tau;
        A_save = A;
        /* Calculate Pearson Correlation Coefficient R */
        sumx = X1 + X2 + X3 + X4 + X5 + X6;
        sumy = Y1 + Y2 + Y3 + Y4 + Y5 + Y6;
```

```
        sumxy = X1*Y1 + X2*Y2 + X3*Y3 + X4*Y4 + X5*Y5 +X6*Y6;
        sumx2 = X1*X1 + X2*X2 + X3*X3 + X4*X4 + X5*X5 +X6*X6;
        sumy2 = Y1*Y1 + Y2*Y2 + Y3*Y3 + Y4*Y4 + Y5*Y5 +Y6*Y6;
        R2 = (sq(6.0*sumxy - sumx*sumy))/((6.0*sumx2 - sumx*sumx)*(6.0*sumy2 -
sumy*sumy));
        R = pow(R2,0.5);
        T = 2.0*R/pow((1-R2),0.5);
      }
    }
    Serial.print("A: "); Serial.print(A);
    Serial.print("\t tau: "); Serial.print(tau_save);
    Serial.print("\t R: "); Serial.print(100*R); Serial.print(" % Correlation");
    Serial.print("\t Student-T: "); Serial.print(T);
    if (8.61 <= T) {
    Serial.println("\t  99.95% Confidence ");
    }
    else if ((8.61 > T) && (4.604 < T))
    {
    Serial.println("\t  99.5% Confidence ");
    }
    else if ((4.604 > T) && (3.747 < T))
    {
    Serial.println("\t  99% Confidence ");
    }
    else if ((3.747 > T) && (2.776 < T))
    {
    Serial.println("\t  97.5% Confidence ");
    }
    else if ((2.776 > T) && (2.132 < T))
    {
    Serial.println("\t  95% Confidence ");
    }
    Serial.println
    ("-----------------------------------------------------------------------------
-----------");
    if (A > 1000) {
    Serial.println("WARNING: EXCESSIVE PARTICULATE MATTER");
    Serial.println
    ("--------------------------------------------------------------------------------
-----------");
    }
    sample = sample + 1.0;    //increment sample number by one
  }
}
/* Gather Particulate Matter "PM" Data */
boolean readPMSdata(Stream *s) {
  if (! s->available()) {
    return false;
  }
```

```
  // Read a byte at a time until we get to the special '0x42' start-byte
  if (s->peek() != 0x42) {
    s->read();
    return false;
  }
  // Now read all 32 bytes
  if (s->available() < 32) {
    return false;
  }
  uint8_t buffer[32];
  uint16_t sum = 0;
  s->readBytes(buffer, 32);
  // get checksum ready
  for (uint8_t i=0; i<30; i++) {
    sum += buffer[i];
  }
  // The data comes in endian'd, this solves it so it works on all platforms
  uint16_t buffer_u16[15];
  for (uint8_t i=0; i<15; i++) {
    buffer_u16[i] = buffer[2 + i*2 + 1];
    buffer_u16[i] += (buffer[2 + i*2] << 8);
  }
  memcpy((void *)&data, (void *)buffer_u16, 30);     // put it into a nice struct :)
  if (sum != data.checksum) {
    Serial.println("Checksum failure");
    return false;
  }
  return true;    // success!
}
```