# Module 2: Water Filtration

**Engineering Disciplines:** Environmental Engineering, Electrical Engineering

For the Module 2, Labs 1 & 2 Report, each team will submit:
- 1 .m file containing code from Labs 1 & 2
- 1 PDF file containing typed answers to the discussion questions for Labs 1 & 2
- 1 PDF file containing the Project Planning and Management Task List

Prepare and submit these files in accordance with the ENGR 130 Style Guide and Assignment Submission Guide.

# Lab 1: Reading Turbidity Signals

## Overview

Turbidity is one metric used to describe the quality of water. Turbidity sensors measure suspended solids (particles floating in water). Suspended solids can include harmful microbes, soil, and metal particles. They can make you sick, clog pumps, and make water aesthetically displeasing.

In this lab, you will start by building a circuit to learn to use MATLAB and the Arduino to control an external element. Then you will construct the sensor circuit and take some test measurements.

## Materials

- Arduino Uno
- USB C to B or USB A to B Cable
- Turbidity sensor (DFRobot PN:SEN0189)
- Computer running MATLAB with Arduino add on
- Breadboard
- 330-Ω resistor
- LED
- Jumper wires

## Procedures

### *1. Project Planning and Management*

a. Complete the Module 2 Project Planning and Management Task List, following the instructions in the Module 2 Project Planning and Management document. You will continue to edit your task list this week and submit your task list, current as of the submission date, as part of your Part 1 (Labs 1 & 2) Lab Report.

## 2. Connect your Arduino

a. Connect your Arduino. In your script file, clear the workspace and command window, then initialize your Arduino using `a = arduino();`. Note: `arduinosetup`, as run last week, downloaded the necessary MATLAB setup files and these are still saved on the Arduino. No need to do this setup again.

## 3. LED circuit

a. Assemble the circuit diagrammed in Figure 1. Pay attention to the position of the components. Remember how a breadboard conducts electricity and how this affects component placement. If you are unsure, ask a TA.

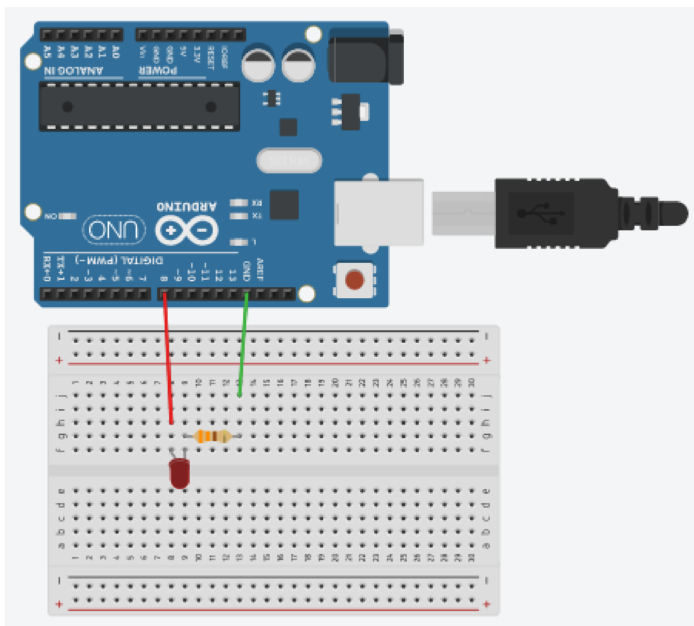

**Figure 1:** LED circuit diagram

b. Write code using the `writeDigitalPin` function to make the LED blink four times (alternate between on `time_period` seconds and off `time_period` seconds a total of four times). Prompt the user for the value of `time_period`. In the circuit we have asked you to build, the LED is connected to pin `'D8'`.

## 4. Testing your turbidity sensor

a. Connect the sensor to the Arduino (Figure 2). The blue wire should be connected to Analog pin A0. Make sure your sensor adapter (the separate circuit board) is set to analog mode by setting the switch to “A”. The cable that attaches to the sensor has a clip that fits into the sensor base.

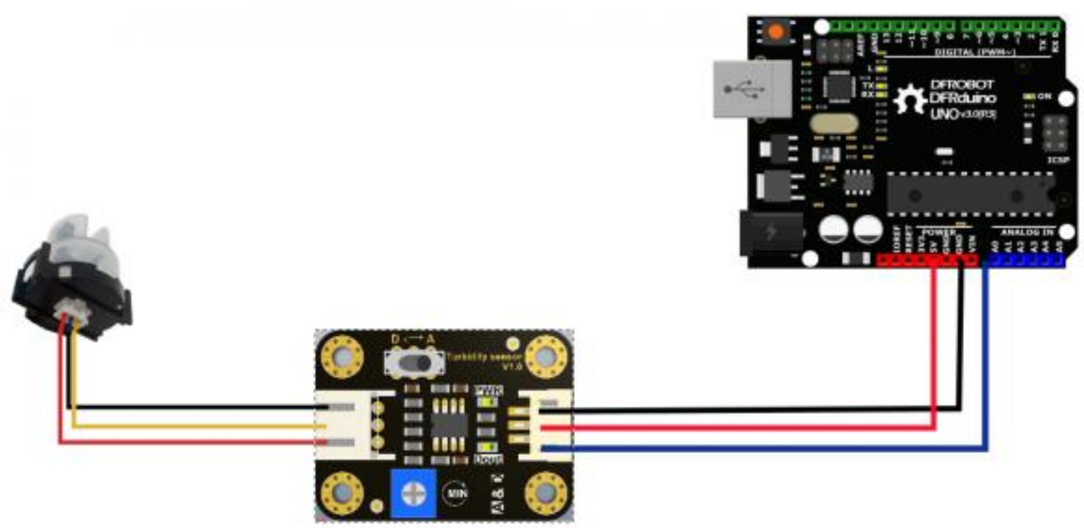


**Figure 2:** Sensor connection to Arduino (red wire = power, black = ground, blue = data)

b. Read a voltage measurement into a variable named `voltage_1` using the `readVoltage` command.

c. Manipulate the sensor output by putting a piece of paper through the middle of the plastic housing. Read another voltage measurement into a variable named `voltage_2` using the `readVoltage` command. If you are reading values close to 0 V or 5 V, check the A/D switch again.

## Questions

1. What are the lowest **and** highest voltages that you can read from the sensor in analog mode? What does this tell you about the working principle of the sensor?

2. From your preliminary understanding of the sensor, how would you use the output from this turbidity sensor to stop a dish washing machine when the dishes are clean? Please be specific.

## END OF LAB 1

# Lab 2: Collecting and Analyzing Turbidity Data

## Overview

In this lab you will write a MATLAB script to interface with your Arduino device and continuously measure turbidity with the sensor. You will enable the user of your script to stop data collection using a button. You will also plan and write code to convert the voltage readings produced by the sensor to the units typically used to measure turbidity.

## Materials

- Arduino Uno
- USB C to B or USB A to B Cable
- Button
- Jumper wires
- Breadboard
- Turbidity sensor (DF Robot PN:SEN0189)
- Computer running MATLAB
- 10-kΩ resistor

## Procedures

### *1. Construct the Arduino button circuit*

a. Construct the Arduino button circuit depicted in Figure 1. You do not need the turbidity sensor to test your button. Note that there is an orientation to the button, as shown in Figure 2.

b. Test your button using the `readDigitalPin` function in the MATLAB command window. If your circuit is constructed correctly, the `readDigitalPin` function should return 1 if the button is pressed and 0 if the button is not pressed.

### *2. Write a script to collect voltage data from the sensor*

a. Read part b. below. Before you start coding, write an algorithm on a whiteboard as a team.

b. Using a loop, write code that collects a voltage (V) reading from the sensor once every second (1 Hz). The loop should run a maximum of 100 iterations and store the voltages in an array named `voltage_vector`.

   i. During each iteration of the loop, check whether the button is pressed. If the button is pressed, stop collecting data and break out of the loop using the `break` command. Remember that the check will only occur once per second – if you press the button for less than a second and miss the check, the button press will not register.

ii. At the end of each loop iteration, use the `pause` command to wait one second.

c. Test your sensor by blocking the light path with a piece of paper while your code runs.

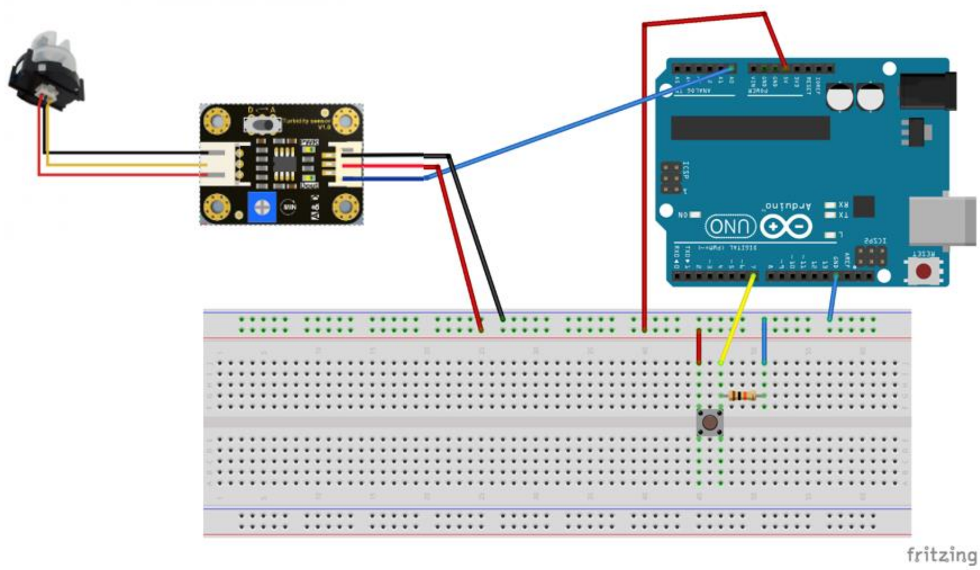


**Figure 1:** Sensor circuit diagram with button

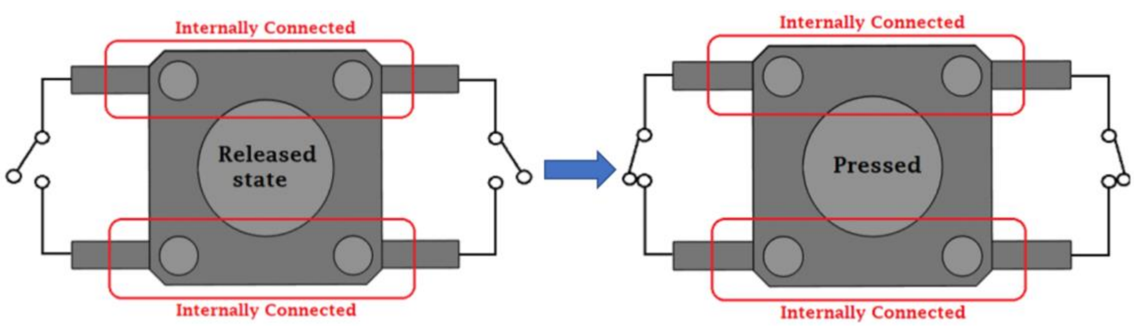


**Figure 2:** Push button diagram (pushing the button connects the two red boxes and lets current flow through them)

### 3. *Convert your voltage data to Nephelometric Turbidity Units (NTU)*

You will not be measuring any water today, but we will introduce a "dummy" formula to calculate NTU. This formula is incorrect, but in the next lab you will replace it with an equation calibrated based on measurements you will take.

a. In a new array named `ntu_vector`, calculate the turbidity (NTU) for each entry of `voltage_vector` using the formula $NTU = -260V^2 + 1800V - 3000$. Use element-by-element operators to do this in a single line of code.

b. Calculate the mean voltage and mean NTU using the voltage and NTU arrays and the `mean` function. Store these values in variables named `mean_voltage` and `mean_ntu`, respectively.

### 4. *Research your water filter*

For the Lab 4 filter design challenge, you will design a water filter to remove turbidity from water. The processes that remove particles in a granular media filter are complex, but include straining, flocculation, and interception. In a typical drinking water plant, water is filtered by introducing water from the top. Clean water exits by filtering through the bottom by gravity. Periodically, the filters are backwashed by reversing the flow of water to dislodge trapped particles. The backflushed water is removed by special troughs at the top of the filter. Common media used in granular filters include gravel, anthracite (coal), and sand. Materials that will be available to you are sand, gravel, cheesecloth, and cotton. You can inspect samples of these materials in the lab.

You will incorporate whatever combination of the provided materials you choose in a provided funnel to create your filter. The filter will be evaluated based on two criteria: how effectively it filters the water and how quickly it filters the water.

a. As a team, do some research, identify criteria you will use to choose your initial design, and brainstorm ideas for the design of your filter. Submit your brainstormed ideas as Question 5 below.

## Questions

1. If the time between voltage samples were reduced to 0.2 seconds, what is the new sampling frequency in Hz? Note: $frequency = \frac{1}{period}$

2. What happens if you use the dummy equation above to convert 2.5 V to NTU? Do think the dummy equation is giving an accurate representation of NTU? Why or why not?

3. Part 2 can be done using either a `for` loop or a `while` loop. Write an algorithm for the problem using the opposite approach your team implemented in the lab (i.e., if you used a `for` loop up above, write the algorithm for a `while` loop here).

4. Looking ahead to the Lab 4, what are the objectives and constraints of filter design challenge?

5. For the Lab 4 filter design challenge, you will have the opportunity to test and iterate before deciding on your final design. Research potential filter designs and brainstorm at least five different prototype filter designs that you plan to test during Lab 4. You can describe your prototype test designs using either a list, table, or sketches. Be sure to include each filter material and approximately how much of each material (e.g., ¼ of funnel depth with sand).

# END OF LAB 2

# Lab 3: Sensor Calibration and Filter Design

For the Module 2, Labs 3 & 4 Report, each team will submit:

- 1 .m file containing code from Labs 3 & 4
- 1 PDF file containing typed answers to the discussion questions for Labs 3 & 4
- 1 PDF file containing the Final Project Planning and Management Task List

Prepare and submit these files in accordance with the ENGR 130 Style Guide and Assignment Submission Guide.

## Overview

In this lab, you will calibrate your turbidity sensor using some provided standard samples. You will also design a granular media filter to remove turbidity from water. You will need this design at the beginning of Lab 4.

## Safety Notes

The turbidity standards can irritate your skin. You must be wearing a lab coat and glasses for today's activities. Any team members who are handling the samples and/or a turbidity sensor being used to measure those samples must wear protective gloves. Do not drink any water, no matter how tasty it looks.

## Materials

- Lab coat and safety glasses
- Protective gloves
- Button
- Jumper wires
- Arduino Uno
- USB C to B or USB A to B Cable
- Breadboard
- Turbidity sensor (DF Robot PN:SEN0189)
- Computer running MATLAB
- 10-kΩ Resistor
- Standard NTU samples

## Procedures

1. *Construct the Arduino button circuit*

    If your Arduino circuit with a button is not assembled, assemble it now using the procedures from the previous lab. Also, test your sensor to make sure it is reading different voltage levels when blocked and unblocked.

## 2. *Calibrate your sensor*

a. At the front sink, there will be four different beakers filled with different turbidity standards. These standards represent 1000, 500, 250, and 0 NTU. (The 0 NTU standard is just tap water.) Use your code from Lab 2 to collect voltage readings from each standard. You will use the mean voltage reading from each standard to fit the sensor voltage to NTU.
   i. Increase your sample rate to 4 Hz to collect data more quickly.
   ii. Leave the sensor standing still; this will give more accurate readings.
   iii. Software can be buggy, or careless errors can be made. Write down your average readings using a pen and paper when you collect them as a backup.
   iv. Once you have collected your data, return to your lab bench so that other teams can easily have access to the samples.

b. Store your mean voltages in a vector named mean_voltages.

c. Plot the known NTU values vs. mean_voltages as shown in Figure 1.

d. Write code to find the equation for NTU from voltage using the polyfit command. Try several values for the polynomial order and use the lowest value that gives a reasonable fit.

e. Using the output from polyfit, create a curve fit to your data using polyval.

f. Add your curve fit to your plot of NTU values vs. mean_voltages. The measured values should be circles and the curve fit should be a solid line.

g. Use fprintf to display the equation in terms of *v*. Your output in the command window should be in the form NTU = <a>v^2 + <b>v + <c>, where <a>, <b>, and <c> are your fitted coefficients inserted using fprintf. Limit the coefficients to 2 decimal places.

## 3. *Make a filter testing plan*

Create a plan for how you will construct and test each of your prototype filters during Lab 4. The plan should include who will be responsible for assembling and testing each design. Alter your Module 2 Project Management Task List as necessary. You should arrive to Lab 4 ready to create and test that design. You will have time in class during Lab 4 to test and iterate the design of your prototype before you demonstrate your final design to the class.

## Questions

1. Between the 1000 NTU and 0 NTU samples, which has the higher voltage reading? Does this match your expectations given what you know about how the sensor measures turbidity?

2. What is the equation for NTU in terms of measured voltage that you determined for your sensor in step 2.g.?

3. What NTU value did your fitted line predict the 250 NTU sample to have, and what was the error with the observed measurement as a percentage of the expected value?

$$\%\ error = \frac{|observed\ measurement - expected\ value|}{expected\ value} \times 100\%$$

4. What are some possible sources of error when measuring the turbidity of the sample solutions? Did your group experience any of these errors?

5. What criteria did you use to determine which of your brainstormed designs seem the most promising to test?

## Figures

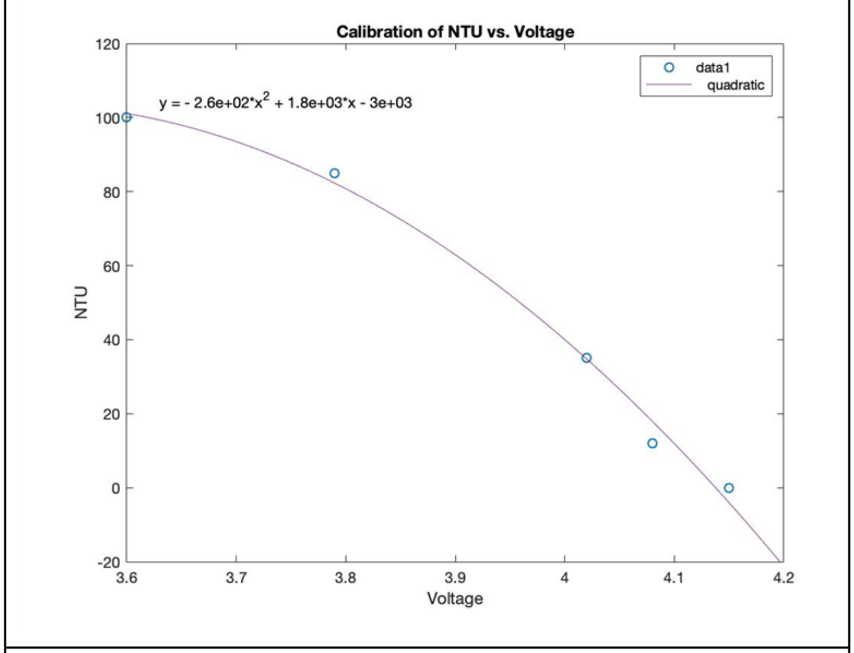


**Figure 1:** Calibration of NTU vs. voltage

**END OF LAB 3**

# Lab 4: Filter Construction and Testing

## Overview

You will build the water filter you designed in the previous lab, test it, and reiterate it as needed to optimize its speed and water filtration effectiveness. You will compare your filter to those of your classmates.

## Materials

- Lab coat and safety glasses
- Button
- Jumper wires
- Arduino
- USB C to B or USB A to B Cable
- Breadboard
- Turbidity sensor
- Computer running MATLAB
- 10-kΩ Resistor
- Cotton balls
- Sand
- Gravel
- Cheese cloth, cut into approximately 2” squares
- Beakers
- Clean water
- Dirty water
- Scissors

## Safety Notes

Limit the dispersal of sand and gravel by gently spooning each substance into your funnel. Also, you must wear glasses and a lab coat during this lab. Do not drink any water, no matter how tasty it looks.

## Procedure

### *1. Construct your water filter*

a. If you haven't already, as a team formulate an experimental plan for testing your design, determining what modifications you will make as you iterate, and keeping a record of your process and findings.

b. Design and construct your water filter using your funnel, cotton balls, cheesecloth, gravel, and sand.

c. Experiment with different layering techniques, water flow rates, and media depths.
   i. The sand and gravel both contain fine particles that should be washed by rinsing your filter with clean water before filtering dirty water.
   ii. A wet filter works better than a dry filter.
   iii. Your filter will be judged using a weighted scale of effectiveness and speed.
   iv. Your design must be able to produce 1 fluid ounce of filtered water in less than 5 minutes.

## *2. Design competition*

a. You and your team will:
   i. Explain your design strategy for developing your water filter.
   ii. Excitedly listen to other teams explain their strategies.
   iii. When told to do so, pour the dirty water into your filter and collect the clear water.

b. A TA will:
   i. Measure the turbidity of the dirty water and the clean water after filtration.
   ii. Award a prize to the group that best meets the stated design criteria.

## *3. Project Planning and Management*

a. Complete your final Module 2 Project Planning and Management Task List. You will turn in a pdf of your final task list with your Labs 3 & 4 Lab Report.

# Questions

1. Presenting clear and neat sketches about a design or prototype is an important communication skill. Using a method of your choice (PowerPoint, CAD, hand sketch), sketch your final filter design. Label the different media components and their depths. Note: Relative depths are acceptable, e.g., ⅓ of the funnel.

2. Did your filter perform as you expected? What was the biggest design factor that impacted the performance of your filter (media type, media depth, flow rate, etc.)?

3. Imagine that the sensor used for completion of this lab was your first prototype. Your goal is to continue to refine this sensor for a client, to make the best possible sensor for measuring turbidity. How would you go about doing this? How would you improve your sensor prototype if it were a real product?

4. Similarly, how would you improve your water filter if you had more time or additional resources?

# **END OF LAB 4**