E80 Final Project Report: Field Deployment of an Autonomous Underwater Vehicle

Nandini Garg, Author, Cristian Gonzalez, Author, Erina Iwasa, Author, Jeremy Kim, Author.

***Abstract*— Due to climate change, the overall temperature of the planet is rising and as such the overall temperature of the water is rising. Since sea organisms and creatures need a specific temperature to thrive, they are migrating into deeper waters. The deeper the water, the cooler it is. Our autonomous underwater vehicle's (AUV) goal was to test different characteristics that make up a sea creature's habitat to see if it's viable for sea creatures to survive in deeper waters. The habitat characteristics that we tested were turbidity, temperature and pressure. We found that the temperature sensor and the pressure sensor were giving us accurate readings and were behaving in the way that we wanted them to behave, whereas the turbidity sensor was not giving us the data in line with our hypothesis.**

I. Introduction

Climate change is the reality of the world that we live in, there is scientific evidence backing up the fact that climate change exists. As a result of humankind's decisions and actions, animals' habitats and ecosystems are being destroyed or increasingly becoming unfit for them to live in. One such ecosystem that has been affected is under the water. Due to rising temperatures, sea levels are rising and ocean temperatures are also rising. This change in temperature makes sea creatures' habitats increasingly unsuitable for them, as a result, they are migrating to deeper waters. The problem that arises with sea animals migrating to deeper waters is the fact that the other characteristics of their habitat that are needed for survival are not optimal.

Therefore we are investigating and exploring how marine habitat characteristics such as turbidity, pressure and temperature change as we go deeper into the water. We constructed an autonomous underwater vehicle (AUV) that moves autonomously in the z-axis and goes about 3 meters deep all while collecting turbidity, temperature and pressure data through the sensors that we designed. The goal of this project is to see how turbidity and pressure data have changed with the rising sea levels and temperature levels by analyzing the data using Matlab code. This analyzed data will then be modeled in the form of graphs in terms of time and depth. How our motivation for this project will be achieved is by analyzing the Dana Point experimental turbidity and pressure data, to literature values from years ago.

II. Scientific and Engineering Motivations

When we were choosing our sensors, they had to be in line with the story or the theme that we had created for our robot. To fulfill our goal of investigating how climate change is changing the temperature of marine habitats and various other characteristics as we go deeper into the water, we had to choose sensors that would measure the different characteristics of the marine habitat that are important for the survival of the fish.

Something that we wanted to measure was temperature because it's in line with the theme of climate change. Pressure made sense to us because the deeper you travel into the water, the more pressure there is and we wanted to see how it had changed over time. The last sensor was tricky because we were thinking of using a light sensor, but when we tried to justify using it to a professor, they made us realize that if we used a light sensor, then the time of day at which we are measuring data is really important and also the weather conditions would end up playing a huge role. This wasn't ideal if we were trying to compare our experimental data with data from other institutes so then we chose to measure the cloudiness of water otherwise known as the turbidity of water.

III. Sensor Selection

A. Temperature Sensor

For this project, we decided to use a 47kΩ Murata NXFT15WB473FA2B150 (Murata) thermistor to measure temperatures of the ocean water. The Murata is essentially a resistor that outputs changes in resistance in response to changes in temperature. The Murata datasheet shows the following equation that drives this relationship.

Equation 1. R for the R_{avg} for the thermistor's expected resistance

$$R_{avg} = R_0 * e^{B*(\frac{1}{T} - \frac{1}{T_0})} \quad \text{+/- 0.1 degrees Celsius} \qquad (1)$$

R_{avg} = resistance in ambient temperature T, R_0 = resistance in ambient temperature T_0, and B = B-constant of the thermistor calculated using the following table [1]:

Table 1. B-Constants for Murata

Part Number	Resistance (25°C) (ohm)	B-Constant (25-50°C) (K)	B-Constant (25-80°C) (Reference Value) (K)	B-Constant (25-85°C) (Reference Value) (K)	B-Constant (25-100°C) (Reference Value) (K)	Maximum Operating Current (25°C) (mA)	Rated Electric Power (25°C) (mW)	Typical Dissipation Constant (25°C) (mW/°C)	Thermal Time Constant (25°C) (s)
NXFT15XM202EA□B□□□	2k±3%	3500±1%	3539	3545	3560	0.27	7.5	1.5	4
NXFT15XV302FA□B□□□	3k±1%	3936±1%	3971	3977	3989	0.22	7.5	1.5	4
NXFT15XH103FA□B□□□	10k ±1%	3380 ±1%	3428	3434	3455	0.12	7.5	1.5	4
NXFT15XV103FA□B□□□	10k ±1%	3936 ±1%	3971	3977	3989	0.12	7.5	1.5	4
NXFT15WB473FA□B□□□	47k ±1%	4050 ±1%	4101	4108	4131	0.06	7.5	1.5	4
NXFT15WF104FA□B□□□	100k ±1%	4250 ±1%	4303	4311	4334	0.04	7.5	1.5	4

□ is filled with lead shape (1: twist, 2: without twist).
□□□ is filled with Total-length codes. (60 to 150mm interval 10mm, ex. 060=60mm)
Maximum Operating Current rises Thermistor's temperature by 0.1°C
Rated Electric Power shows the required electric power that the thermistor's temperature rise to 30°C by self-heating, at ambient temperature of 25 °C.
Operating Temperature Range: -40°C to +125°C

Using Equation 1 and Matlab, we were able to write a script that would take the resistance values from our teensy and convert them into temperatures automatically. The Murata allows us to plot change in temperature as the depth of our robot increases. We predict that temperature will decrease as depth increases.

B. Turbidity Sensor

Turbidity sensors are used to measure the scattering of light in the water. Since our goal is to understand the environmental factors in which fish thrive, we want to measure the amount of light that is scattered in water as opposed to the light intensity. The intensity of light in the water changes according to the weather conditions. If it is too cloudy, the light intensity in the water is a lot lower compared to when it is sunny, and therefore we decided that turbidity would be more effective in determining how the quality of the water itself affects fish populations. To construct our turbidity sensor, we used a 950nm OP950 photodiode and an IR1503 LED. The LED pulses at a constant rate through flowing ocean water and the photodiode reads the amount of scattered light and outputs a respective current, which we converted into a voltage.

C. Pressure Sensor

The pressure sensor will be used to measure the robot's depth. The MPX5700ASX (MPX) is a piezoelectric transducer that outputs an analog voltage signal that is proportional to the pressure it experiences. This sensor allows us to create the foundation for our graphical analysis and is the independent variable. The output of the MPX to time can be used as a backdrop for our temperature and turbidity outputs to time. In other words, changes in the pressure sensor outputs are directly linked to changes in temperature and turbidity.

IV. Circuit Design

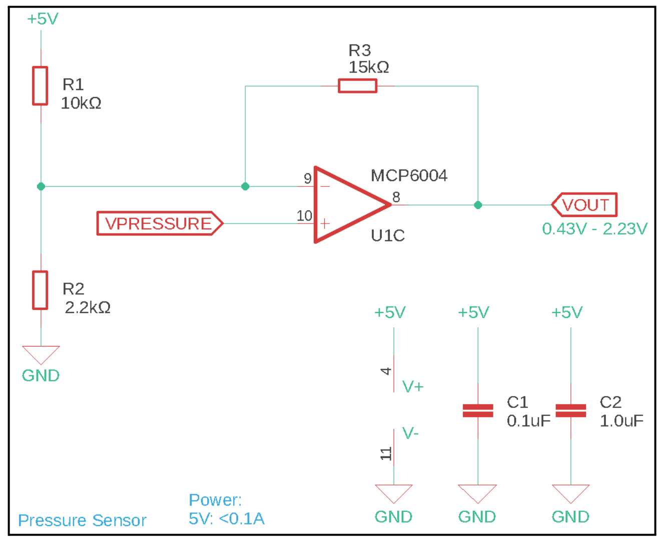
D. General Design Considerations

Each sensor was constructed on a protoboard that interfaced with the main system through analog pins [2]. We used an L7805 voltage regulator to maintain a voltage source of approximately +5V through one rail and grounded the other rail. When designing each circuit, we wanted our sensors to have output voltage swings that could appropriately model data while also staying within the 0 to 3.3V Teensy input range. Three circuit designs used a single-supply op-amp, so we chose an MCP6004 to simplify the circuit assembly process [3]. Two bypass capacitors of different magnitudes, 0.1μF and 1.0μF, were connected to the op-amp's power supply to reduce noise and the effect of transient current. The team also included 0.1μF bypass capacitors connecting the +5V source to the ground in each sensor's design. What follows is a detailed description of each design.

E. Pressure Sensor

When designing the pressure sensor circuit, we first looked at the MPX5700ASX datasheet to learn its output characteristics [4]. Based on our desired depth of three meters, and instructions from section 5, lab 7 [5], the pressure sensor alone had a projected output voltage swing of 0.85V to 1.04V. To increase output swing, the team connected the pressure sensor to an op-amp and used a non-inverting amplifier with biasing design. When testing the pressure sensor's output, the team observed that the output voltage sometimes fell below the expected 0.85V. After analyzing this, the team concluded that this discrepancy was likely due to assumptions made when computing the expected output, primarily density. To account for this behavior and reduce the risk of the output falling outside the acceptable range, the team chose to bias the op-amp at a non-zero voltage at zero meter depth. The system's gain was controlled by all three resistors, while the offset was determined by R1 and R3. We selected component values of R1 = 10kΩ, R2 = 2.2kΩ, and R3 = 15kΩ. Based on the selected components, the op-amp was biased such that when the robot was not submerged, the pressure sensor would return a voltage of 0.43V. When the robot reached the desired depth of 3m, the sensor would read a voltage of 2.23V.

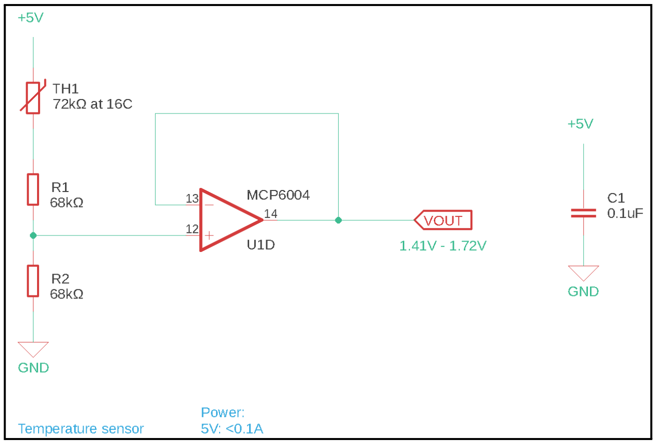
Fig. 1. Pressure Sensor Circuit Diagram



F. Temperature Sensor

With information from section 1, lab 5 [6], the thermistor was in series with two resistors and was used to create a voltage divider circuit that produced a temperature-dependent output. While the voltage divider could have been designed with one resistor, the team incorporated two biasing resistors of equal value to prevent the output voltage from surpassing the 3.3V limit at extreme temperatures. Based on the expected average water temperature at Dana Point in April, 16℃, and a goal output bias voltage of 1.65V, the midpoint of the Teensy input swing range, we chose 68kΩ resistors for our biasing resistors. The output voltage was then fed into a buffer to mitigate the influence of Teensy's low input impedance.

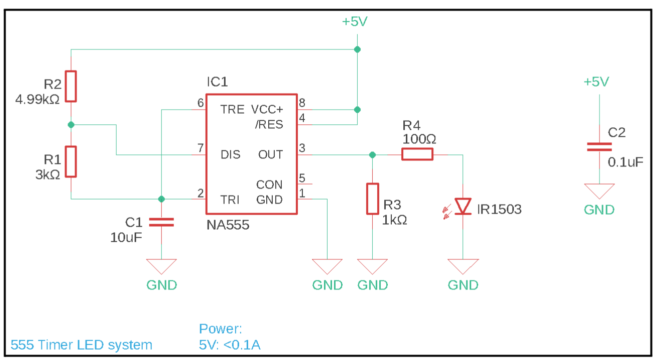
Fig. 2. Temperature Sensor Circuit Diagram



G. Turbidity Sensor

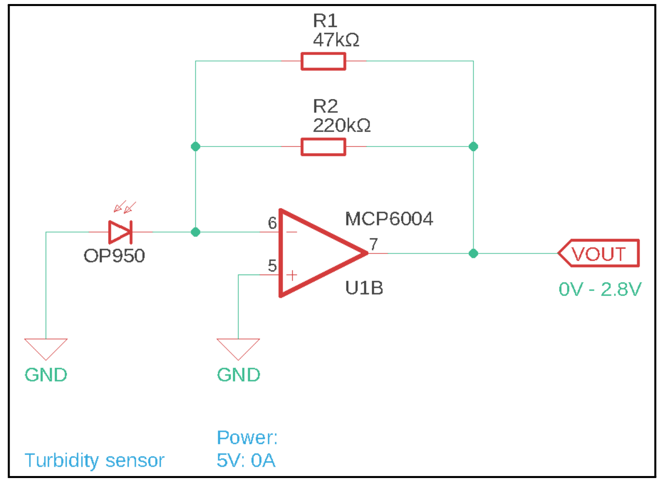
The first step in designing the sensor was constructing the LED circuit. To achieve a square wave with a constant pulse, the team used a NA555 timer, inspiration from section 5, lab 3 [7]. The circuit design was based on the astable circuit design provided in the xx555 timer datasheet [8]. The output of the timer was connected to a 100Ω limiting resistor which fed into the LED. The purpose of the limiting resistor was to mitigate the risk of the LED burning out.

Fig. 3. LED Circuit Diagram



The photodiode [9] was used to create a trans-impedance amplifier, where the photodiode behaves as a current source. Our original design used a 47kΩ resistor to determine the output swing, but after testing, the team noticed that the output surpassed 3.3V. The amplifier's output was computed as the photodiode current multiplied by the feedback resistance. So to solve our problem, a second resistor was inserted in parallel with the 47kΩ resistor to reduce the equivalent feedback resistance. We reduced the equivalent resistance to 38.7kΩ using a 220kΩ resistor. Remeasuring the output voltage, the sensor's maximum voltage was reduced to 2.8V. The minimum output was 0V, which occurred when the LED [10] was completely obstructed, causing the photodiode to behave like an open circuit.

Fig. 4. Turbidity Sensor Circuit Diagram



V. Robot Design

H. Robot Design

We designed our robot to be symmetrical such that the center of gravity and center of buoyancy would allow our robot to submerge in a straight line without flipping over. The components of the robot consist of ½ inch PVC pipes (eight corner junctions, fourteen tee junctions, eight 7" pipes, eight 2.5" pipes, four 12" pipes, four 4" pipes, and four 8" pipes), plastic mesh, one 7.7"x3.5"x7.4" waterproof box, one 12V battery, one motherboard [11], one protoboard, one ribbon cable, four penetrator bolts connecting to pressure tubing, four motors, one turbidity sensor, and one thermistor, ballast weights, two velcro straps, one carabiner, and one 3-meter rope. Our robot is in the shape of a rectangular prism and the model is shown below:

Fig. 5. Side Profile Drawing of Robot

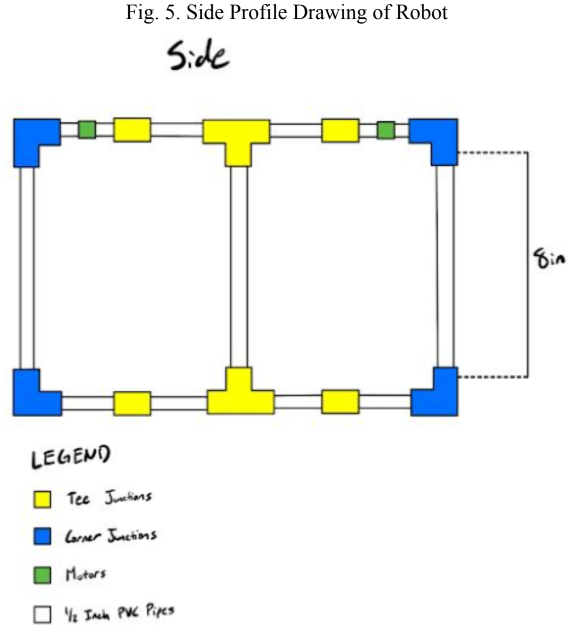
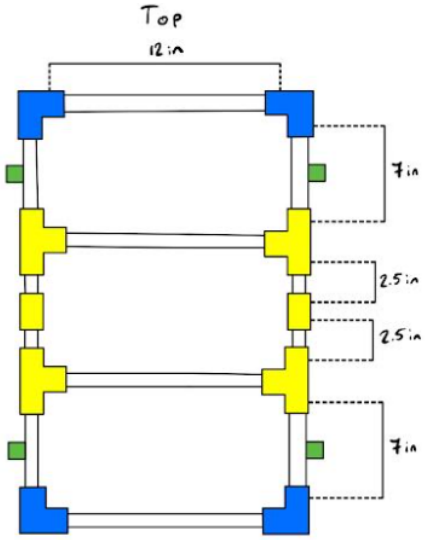


Fig. 6: Top Profile Drawing of Robot

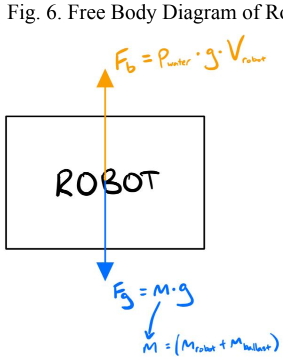


The PVC pipes and junctions made up the frame of the robot. We originally had a pipe above where our waterproof box was situated, but later removed it due to poor accessibility. As a result, the two tee junctions above the middle of the robot perpendicular to the others serve no purpose. The plastic mesh was zip-tied to the base of the robot to hold our waterproof box in conjunction with the velcro straps. The dimensions of our robot allowed for the waterproof box to be snugly fit in the center of our robot. The waterproof box held our battery, motherboard, and protoboard. Also, the waterproof box had four 15/32 inch holes drilled into the front and right sides of the box for penetrator bolts to be screwed in. We used the carabiner and 3-meter rope as a tether in case the robot sank. In determining the location and weight of our ballast, we decided to hang the ballast from the bottom of the robot, right underneath the waterproof box. This placement would make sure that the ballast stays in line with the center of gravity.

I. Ballast Calculation

Drawing a free body diagram, we know that if the center of gravity and center of buoyancy are aligned, then the two forces must be equal and opposite for the robot to be neutrally buoyant:

Fig. 6. Free Body Diagram of Robot



Therefore, $F_g = F_b$. F_b is calculated by multiplying the robot's volume, density, and acceleration due to gravity, which came out to be 67.69 N. After determining the mass of the robot to be 3.83kg, setting F_g equal to F_b allows us to solve for how much ballast we need, shown in Equation 2.

Equation 2. Summation of Forces of Robot

$$F_g = (m_{robot} + m_{ballast}) * 9.81\frac{m}{s^2} = 67.69N$$
$$(3.83kg + m_{ballast}) * 9.81\frac{m}{s^2} = 67.69N$$
$$37.57N + m_{ballast} * 9.81\frac{m}{s^2} = 67.69N$$
$$m_{ballast} = \frac{67.69N - 37.57N}{9.81\frac{m}{s^2}}$$
$$m_{ballast} = 3.08kg$$

We placed 3.08kg of ballast weight in the form of bolts and large screws into a watertight zip lock bag. As stated previously, we zip-tied this bag directly underneath the center of the waterproof box. We also made sure to make the distance between the bottom of the robot and the ballast weight as small as possible such that the current of the ocean would not cause the ballast to become uncentered, leading to an unbalanced robot that would tip over underwater.

VI. Modeling

A majority of the modeling process comes from a Google spreadsheet designed to calculate the output voltage for all 3 of the sensor circuits, based on certain parameters, resistance values, and conversion equations provided. The modeled output voltages would then be compared to experimental data taken in the E80 tank room and at Phake Lake, once processed by Matlab code created by the team.

The Teensy 3.3 device attached to the motherboard has a maximum voltage acceptance of 3.3V., which is why calculating output voltage is prioritized in each of the modeling spreadsheets.

J. Modeling Data for Pressure Sensor

The pressure calculations can be found on this hyperlink. The formulas used in the computation are based on the schematic in Figure 1. Equation 3 details the expected output voltage that we can imagine with chosen resistor values. The R1 resistor was set at a standard 10kΩ, while the rest of the resistance values were based on R1's value.

Equation 3. Pressure Sensor Output Voltage

$$V_{out} = V_{ps} * (\frac{R3}{R2} + \frac{R3}{R1} + 1) - V_{DD} * \frac{R3}{R1} \qquad (3)$$

For the pressure sensor circuit output, the desired voltage swing of 2.7V, with a low minimum voltage input of 0.3V, and maximum voltage output of 3V, was set. This is to ensure that the output voltage lies in between 3.3V. After resistor values are determined by the spreadsheet's calculations, the density of the water tested also needs to be taken into account. Saltwater has a density of 1023.6 kg/m^3, and freshwater has a density of 997.1 kg/m^3. When setting a specific V_{out}, the spreadsheet will provide a resultant V_{ps}, pressure in kilopascals, and depth.

The takeaway from needing to model the pressure sensor is that calibration at Dana Point is required before submersion, as the atmospheric pressure and saltwater content at Dana Point will differ from predictions done in the lab. Calibrating will require the use of a graduated cylinder, and creating a plot that captures a linear relationship between depth and voltage.

K. Modeling Data for Temperature Sensor

The temperature model calculations can be found on this hyperlink. For modeling the expected temperature at Dana Point, there needed to be a temperature range (min/max) defined that was expected from a Dana Point deployment, so the range ended up being T_{min} = 10.5 C. & T_{max} = 19.11 C.

Looking back to Equation 1, R_{avg} represents the calculated expected resistance of the thermistor. Once the range of temperature is converted to the thermistor's resistance value using Equation 1, Equation 3 solves for the output voltage of the temperature sensor circuit, assuming that the thermistor takes up the average resistance predicted in the calculations. Equation 3 is based on the circuit diagram in Figure 2.

Equation 3. V_{out} for Thermistor Circuit.

$$V_{out} = V_{DD} * \left(\frac{R2}{R1+R2+(1000*R_{max})}\right) \qquad (3)$$

Once the spreadsheet model was finalized, a breadboarded model of the Figure 2 circuit was created, and utilizing an oscilloscope, the output voltage was recorded and tested on the thermistor for liquid samples of different temperatures (of about +15 +/- 5 degrees Fahrenheit). The oscilloscope's recorded data was still in Volts, which still needed to be converted to Celsius, using Equation 4 shown below.

Equation 4. Voltage to Temperature Conversion Equation

$$T_C = \frac{1}{\frac{1}{B}*ln(\frac{R_{therm}}{R_0})} + \frac{1}{T_0} - 273.15 \qquad (4)$$

n

Once the data gathered from the oscilloscope was converted to temperature (in degrees Celsius), it was compared to the spreadsheet model, and the predicted values matched the experimental value. Multiple samples were conducted to ascertain the accuracy of the temperature sensor, and after the accuracy was validated, the temperature sensor was mounted to our robot and was ready for deployment testing and Dana Point.

L. Modeling Data for Turbidity Sensor

The modeling for the turbidity sensor deviated from the modeling process for the pressure and temperature sensor. An experiment was set up that involved gathering 6 samples of tap water with different quantities of dry milk powder to mimic liquids of differing turbidities. Using a TU-2016 calibrated laboratory turbidity meter, the turbidity of each sample was measured in Nephelometric Turbidity Units (or NTU). Afterward, each sample was placed into small cuvettes, which were slotted into the constructed turbidity meter. The construction of the turbidity meter is listed in Reference Designs on the E80 website [12].

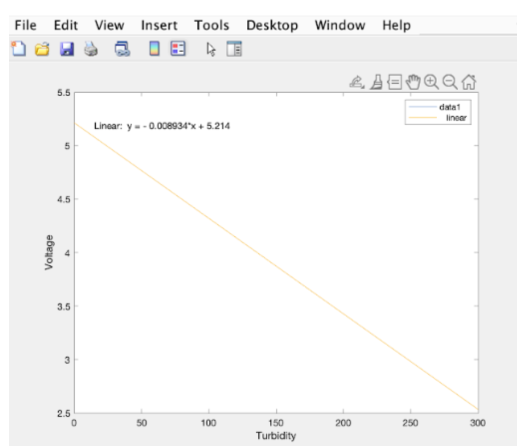
The output voltage of the turbidity circuit, based on Figure 4, is shown in Equation 5, which maps the relationship between the output voltage and turbidity readings. This result was displayed using an oscilloscope, and a constructed model of the turbidity circuit on a breadboard. A spreadsheet was also created that allowed for multiple testing, which is hyperlinked here.

Equation 5. Output voltage for turbidity circuit

$$V_{OUT} = V_{DD} * \frac{R2}{R1+R2+LDR} \qquad (5)$$

Whenever a particular dry milk-water solution was inputted into the turbidity sensor, the voltage read on the oscilloscope would be recorded and paired with the earlier turbidity reading of the same sample. This leads to six pairs of sample readings of both turbidity and output voltage for each dry-milk water solution. On Matlab, a plot of Voltage [V] v.s Turbidity [NTU] is shown in Figure 7, using the sample readings to form a best-fit slope equation.

Fig. 7. Turbidity vs. Voltage



From Figure 7, the best-of-fit line equation, Equation 6, is below. This plot and equation revealed that the maximum voltage was higher than 3.3V., so resistor values in our initial circuit diagram needed to be changed, so the Teensy was not overloaded with voltage.

Equation 6: Turbidity to Voltage equation.

$$Voltage = -0.008934 * Turbidity + 5.214 \quad (6)$$

For reference of scale, when the turbidity meter measured cuvettes with no liquid inside (meaning the vessel was completely clear, the output voltage reading from the turbidity circuit was at its maximum. The TU-2016 calibrated turbidity meter also displayed an NTU reading of 0 NTU. On the contrary, when the cuvette contained a liquid with an overwhelming amount of dry milk powder (or a solid was placed inside), the output voltage was 0V. The TU-2016 calibrated turbidity meter displays an NTU reading of about 300 NTU.

The takeaway from the turbidity sensor model was that the hole of the turbidity meter must not be pre-maturely filled with unwanted epoxy or debris, to guarantee the most accurate turbidity readings. This was ensured before deployment at Dana Point.

VII. Experimental Procedure

The first thing that we did when we arrived was set up a workstation, where we first uploaded our code to the teensy, just to make sure that the right code was uploaded. Then we tested our motors to see if they were 1. running 2. Running in the way that we wanted them to run (direction and speed were the bigger considerations). Then we decided to take our robot to shallow waters and test to see if the AUV was Neutrally buoyant. This wasn't the case we found, so we then started to experiment with the different places we could put out ballasts to make the robot neutrally buoyant.

Ultimately, we decided that this method of trying to make the AUV neutrally buoyant wasn't working so we switched our approach. We decided to hang a bunch of ballasts from underneath the motherboard box, which meant that most of the weight of the robot was concentrated towards the center of the robot. The reason we believed that this would work is that if most of the weight is low in height, then it will be tough to flip the robot over as the center of gravity will never extend beyond the frame in ideal circumstances. We also made sure that the weight was not too far from the box because if it was then the weights would slosh around and that's something we didn't want to happen. That also defeats the purpose of trying to lower the point of the center of gravity.

After we got our robot to be neutrally buoyant we decided to head to the pier where we decided to do our first round of testing. Going into this we knew that the first round of testing was not going to be good enough and that we would need to test multiple times. Not because we didn't have faith in our robot but because there were a lot of factors that were out of our control. Sure enough, when we first tried to start testing, our robot was not moving in the way that we wanted it to. This was since one of the motors was reaching full capacity of power faster than the other motors.

We also noticed that the same motor seemed to be running more efficiently than the other motors, which means that it was stronger than the other motors while being at the same power setting as the rest of them. This was frustrating because technically speaking our code was correct and everything was correct but this motor decided to act strange. So then we went back to our workstation and decided to lower the power of this specific motor and again it was a bit of a trial and error methodology to see what would be the right power setting to put it at. Then we went back to the pier and everything worked this time. We were able to get the robot to move autonomously for 1 minute which was the goal. Our sensors were fully functional and were giving data that was in line with the way we expected the system to work.

VIII. Comparison of Data to Model

The data shown below is from the deployment at Dana Point and displays the data from the three sensors: temperature, pressure, and turbidity. Figure 8 displays Depth from the surface, Figure 9 displays Turbidity, and Figure 10 displays Temperature (in degrees Celsius), all with respect to time. The data was gathered during the robot's 130-second autonomous deployment.

Fig. 8: Depth from Surface v.s Time @ Dana Point

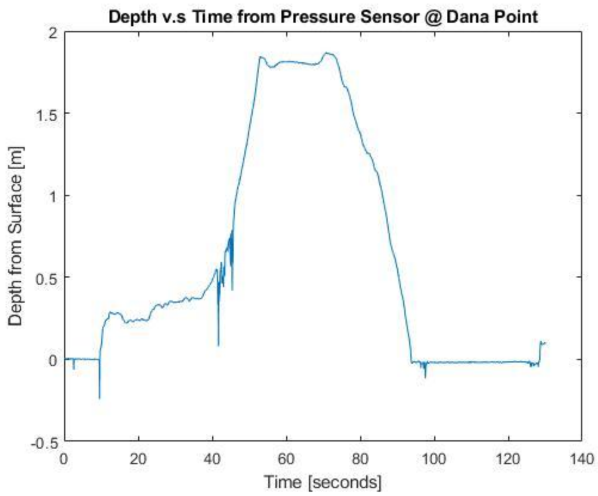


Fig. 9. Turbidity v.s Time @ Dana Point

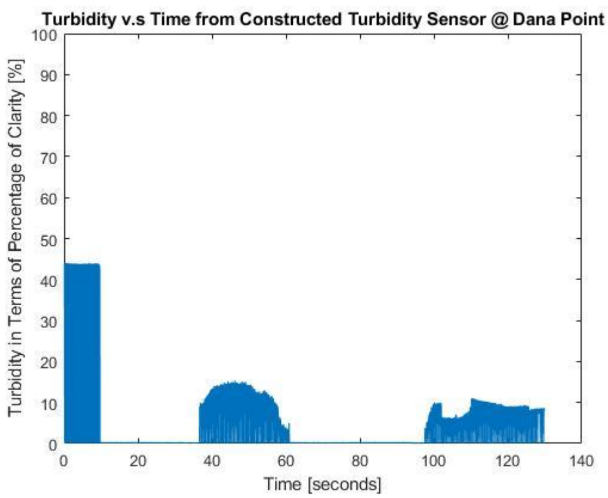
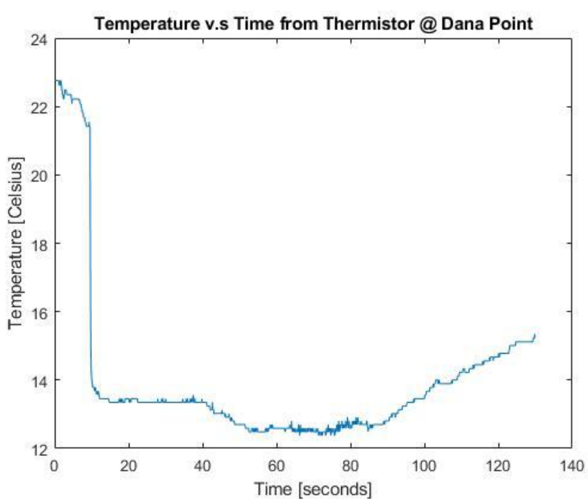


Fig. 10. Temperature v.s Time @ Dana Point



M. Pressure Sensor Data Analysis

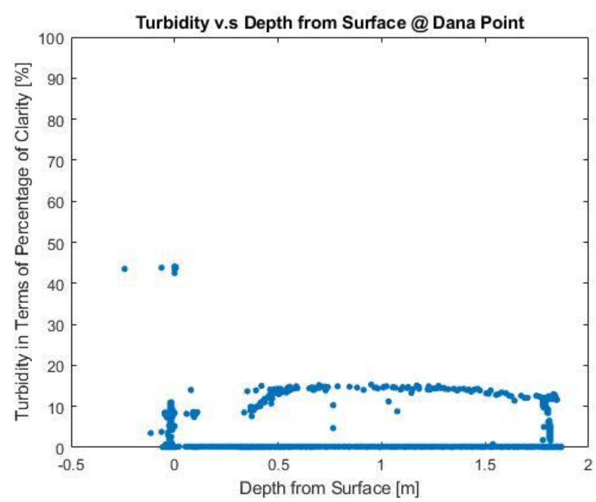
Referring to Figure 8, the data from our pressure sensor indicates that, from our location of deployment (a pier near the coast of Dana Point), the surface of the ocean to the ocean floor was roughly 2 meters. The intended submersion depth planned for was 3 meters, but the location of our deployment likely hindered this as a possibility.

However, the data provided still matches the behavior expected of our deployment. The robot reached the ocean floor at around 1 minute, which was when, during our deployment, the tether attached to the robot stopped giving way, indicating the robot had nowhere else to go. The graph exhibits a plateau during that time, and depth begins to decrease again because the robot starts to re-emerge to the surface. This is similar behavior to how the model spreadsheet calculated the deployment to be.

N. Turbidity Sensor Data Analysis

The data from our turbidity sensor indicated that at the surface level, the ocean water had a turbidity of about 45% clarity (or about turbidity of 165 NTU). As the robot began its autonomous descent, the percentage of clarity decreases to below 20% (above turbidity of 240 NTU). This behavior matches with our modeled data, as the more the robot submerges, the more particles that are suspended in the water, which indicates a lower percentage of clarity of the water. The turbidity sensor's behavior is mapped for depth in Figure 11, shown below.

Fig. 11. Turbidity v.s Depth @ Dana Point



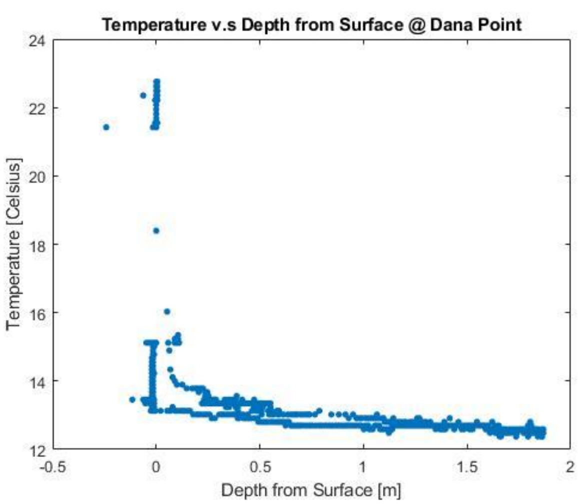
The small gaps in data in Figure 11 are likely a result of debris entering the turbidity meter's hole, obscuring the photodiode from intaking IR light from the LED. The points that display 0% clarity (or turbidity of 300 NTU), are not actual turbidity readings, but a result of the use of a 555 timer in our turbidity sensor circuit. The 555 timer's output is a pulse square wave that fluctuates from 0V to its peak voltage, and the output is what powers the IR LED. Because the photodiode gets its reading from the LED, therefore, the output of the photodiode will also fluctuate with the same frequency. Therefore, the 0% readings on the graph are the pulse's result.

After conducting the deployment, a couple of improvements could be made to get a more accurate reading from the turbidity sensor. Small, thin mesh can be used to cover both sides of the hole in the turbidity meter to prevent potential debris from getting in. Another implementation could be to elongate the length of the turbidity meter, to prevent the turbidity meter from shaking more during deployment. More surface area means more locations to secure the turbidity meter.

O. Temperature Sensor Data Analysis

The data, in Figure 12, gathered from the temperature sensor indicates that, from the moment the robot was submerged into the ocean, there was a spike in our temperature plot. This is a result of the thermistor having to adjust to the sudden change in temperature in its surroundings, from atmospheric temperature to surface water temperature. This sudden temperature change was from about 23 degrees Celsius to about 14 degrees Celsius.

Fig. 12. Temperature vs. Depth @ Dana Point

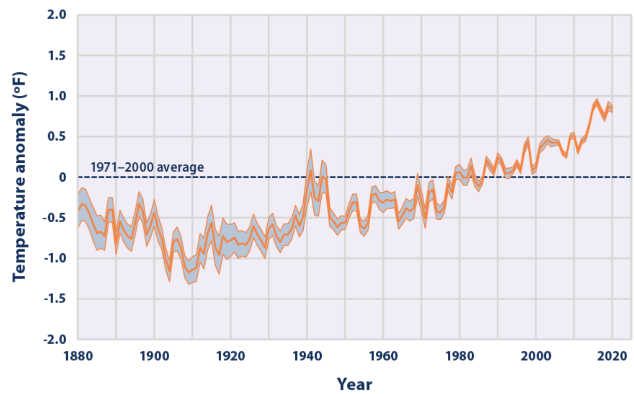


After the spike, the robot submerges down to 2 meters, and the temperature decreases from 14 degrees Celsius to 12.3 degrees Celsius. When the robot re-merges back to the surface of the water, the temperature doesn't immediately spike back up because the thermistor requires more time to dry off.

P. Comparison with Dana Point Data and Climate Change Research on Saltwater Temperature Changes

As indicated in section *O. Temperature Sensor Data Analysis*, the temperature range of seawater from the surface to about 2 meters in depth is about 14 to 12.3 degrees Celsius or 57.2 degrees Fahrenheit to 54.14 degrees Fahrenheit. In an article written by the United States Environmental Protection Agency (EPA), the change in sea surface temperature across the world from 1901 to 2020 has been about 1.68 degrees Fahrenheit Figure 13 below shows the EPA's graph for plotting the average surface temperature of the world's oceans across decades. [13]

Fig. 13. Average Surface Temperature of World's Oceans from 1880-2020



This would imply that in the 1900s, the surface temperature of Dana Point's ocean water would likely have been between 55.52 to 52.46 degrees Fahrenheit, or between 13.07 to 11.37 degrees Celsius. Although the change may seem insignificant in comparison to how humans perceive temperature, even a degree Celsius change in seawater is more than enough to force fish to migrate downwards into the ocean depths.

IX. Conclusion

The data recorded from our deployment is in line with our predictions. As our robot went deeper, the temperature decreased and turbidity decreased from 100% out of the water to 45% after submersion. For future experiments, as previously mentioned, we would fasten our turbidity meter in such a way that no debris could enter and block the photodiode from receiving light pulses. With data from our temperature sensor, we noted a salt-water temperature nearing 12.37 degrees celsius, or 54.27 degrees Fahrenheit, at 1.8 meters. For common coldwater fish, such as the Arctic Char, temperature nearing 12 degrees Celsius is a reasonable temperature to live in [14]. However, the Arctic Char and much other coldwater fish are not present in California. We can assume from the data recorded that if we had measured at greater depths, the temperature of the water would decrease even more. Therefore, aside from a baby fish found near the beach shore at a significantly higher temperature, this offers a reason for the lack of fish observed at Dana Point.

For further experimentation, we would like to test the temperature and turbidity of much warmer ocean waters in areas where fish are much more likely to be present. We would also like to implement a sonar sensor to see the depth of the ocean floor at the place of deployment, and potentially track if fish are present due to reflected sonar signals from the fish's body. The sonar sensor would allow for a better understanding of the fish populations at depths that are not visible to the human eye from the surface. Furthermore, changing the testing site to an area with warmer water, and consequently, more fish, would allow us to more clearly see the relationship between water temperatures and quantities of fish.

X. Acknowledgments

The team would like to thank Lynn for always motivating us without making us feel lesser. We would also like to acknowledge Xavier for his help when it came to debugging our turbidity sensor. We would also like to extend our gratitude to Sam, for always helping us find the materials we need and always having them available. We would like to thank Prof. Spjut for helping us become better engineers by giving us tips and tricks that we now use regularly. We would like to thank the proctors, for being our friends and giving us some real, good advice when we needed it.

XI. References

[1] *NTC Thermistors - Murata Manufacturing*. https://www.murata.com/-/media/webrenewal/support/library/catalog/products/thermistor/ntc/r44e.ashx?la=en-us&cvid=20201028054203000000.

[2] "Protoboard_schematic.PDF." *Google Drive*, Google, https://drive.google.com/file/d/1pDpDoSiCln73AbEeGSCm-o7L3ETTjlG9/view.

[3] *MCP6001/1R/1u/2/4 - 1 MHz, Low-Power OP Amp - Microchip Technology*. https://ww1.microchip.com/downloads/en/DeviceDoc/MCP6001-1R-1U-2-4-1-MHz-Low-Power-Op-Amp-DS20001733L.pdf.

[4] *Integrated Silicon Pressure Sensor On-Chip Signal ... - NXP*. https://www.nxp.com/docs/en/data-sheet/MPX5700.pdf.

[5] "Lab 7 Navigation." *Lab 7 Navigation*, https://sites.google.com/g.hmc.edu/e80/labs/lab-7-navigation?authuser=0.

[6] "Lab 5 Temperature." *Lab 5 Temperature*, https://sites.google.com/g.hmc.edu/e80/labs/lab-5-temperature?authuser=0#h.p_BMJyEjjmlr87.

[7] "Lab 3 Op-Amps." *Lab 3 Op-Amps*, https://sites.google.com/g.hmc.edu/e80/labs/lab-3-op-amps?authuser=0#h.p_wn1Vs4L1kDgL.

[8] *Diodes Incorporated*. https://www.diodes.com/assets/Datasheets/NE555_SA555_NA555.pdf.

[9] *OP950 Series - Farnell.com*. https://www.farnell.com/datasheets/2702437.pdf.

[10] *American Opto Plus LED Super Bright Lamp L-513SRD-C*. https://www.aopled.com/AOP_PDFs/L513SRD-C.pdf.

[11] "MTB_Schematic_3.0.2021.PDF." *Google Drive*, Google, https://drive.google.com/file/d/1pk_mORYsxhedKjjqzhMrxj9H5ZVH3zqp/view.

[12] "Reference AUV Turbidity Meter Procedure." *Google Docs*, Google, https://docs.google.com/document/d/1RYaInz-txbC4iaJD83_0HSwKYBlVGDN1HlEs4yDEMjM/edit.

[13] Environmental Protection Agency, "Climate Change Indicators: Sea Surface Temperature" NOAA, April 2021. Available from: https://www.epa.gov/climate-indicators/climate-change-indicators-sea-surface-temperature#%20

[14] Potential impacts of global climate change on freshwater fisheries - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Temperature-tolerances-of-some-common-coldwater-coolwater-warmwater-and-tropical-fish_tbl1_225498321

1