AIR AND WATER CONDITION DATA BUOY

Bryan McCombs	EE
Daniel Rosabal	EE
Julian Rojas	CE
Matthew Crouch	CREOL

Table of Contents

1.0 Executive Summary

The purpose of this design project is to integrate several electronic components and devices to implement a fully functioning water condition data buoy. Throughout this paper we will discuss a variety of different elements and components that it will take to produce the complete design.

One of our main goals that we wish to achieve is to deliver a completely functioning buoy system that will be able to monitor and record valuable data used by the average water sports enthusiast. The objective of this project is to implement a microcontroller that monitors various sensors simultaneously to produce a buoy system that allows any person to simply check the water conditions of the surrounding area. This will be a valuable device with intentions to be focused towards the recreational fishing or diving enthusiast. With that in mind, we aim to make the overall cost low while keeping the functionality high. The final objective is to incorporate several electronic sensors, parts and elements to produce a product that can be attainable by the average middle to lower class enthusiast.

The buoy will be intended for the use of checking water salinity, monitoring as well as recording water temperature, air temperature, wind speed, and wave height. With these multiple sensors sampling the surrounding water and air conditions periodically, the buoy system will allow any person to electronically check and monitor these sources at any time. The simplicity of this system will be through the wireless communication interfaced to a computer.

The specifications have been narrowed down by the members of the design group where the buoy itself will be made from a durable plastic or composite type of material housing which will keep the buoy afloat on the water in which will be best for measuring the data needed. The housing will also act as a host to a microcontroller, as well as several sensors, switches and LED’s that will be implemented into the design.

A microcontroller will first be implemented into the design acting as the “brain” of the buoy system. The microcontroller will analyze, measure and record data through the various sensors integrated in the buoy. The microcontroller will be powered by a small portable battery. The battery will charge throughout the daytime with the use of a solar panel attached to the housing. This will help the battery stay charged so that the user does not have to keep removing the buoy from the water and charging or replacing the battery.

The wave height feature of the buoy will be performed through an accelerometer or gyroscope to measure pressure changes. When buoy moves up and down through the waves, the pressure changes with elevation. With the data translated through the microcontroller, monitoring wave height will provide a key feature to this system. The accelerometer or gyroscope will be integrated inside the housing of the buoy, keeping it out of the wet environment and safe of any possible rough conditions.

The solar panel that will be used will charge the small portable battery during daytime hours allowing enough power if the buoy is to be used during night time hours while not charging. The panel will be attached to the top portion of the buoy for maximum sunlight during peak hours of the day. The panel must be waterproof well in a chance of harsh weather conditions such as rain and storms. This also helps in the case that there is always a chance the waves come over the buoy if the surrounding water is rough enough.

The water salinity testing will be performed through the use of fiber optics and photonics. The circuit itself will consist of a laser diode, fiber optic wire, a prism, and a photo detector. The salinity detection feature will be added to the lower portion of the buoy where the fiber optics will reach the water it will be sampling. From the implemented circuit we will be able to detect if the surrounding water has salt content. Temperature sensors will be integrated into the system to monitor the air and water temperatures. The water temperature sensor will be waterproofed and integrated into the lower portion of the buoy suitable for monitoring accurate readings of data. The air temperature sensor will be mounted on the top portion of the buoy which will make monitoring the temperature of the air relatively easy. This sensor will have to be waterproofed as well in case of bad weather or rough waves coming over the buoy. The wind speed feature will be designed and integrated through the use of an anemometer which will monitor the force and velocity of the wind. This feature will be integrated onto the top of the buoy system clear of any other parts of the buoy or surrounding structures. The integration of all these features will result in a fully functioning buoy system that will provide valuable data interfaced wirelessly to the user's computer.

The 5 categories the requirements of this project fall into are:

- Durability
- User Friendly
- Electronically Advanced
- Functionality
- Portability

The design we are aiming to achieve will create a possibility for the average recreational water sports enthusiast to own a cost efficient water and air condition buoy system that can be transported and used wherever needed. With the overall price being a fraction of the cost compared to commercial systems and other competitors, the average person will be able to utilize this technology.

The project postulates a production of an affordable and accurate data buoy that incorporates several different sensors and technology, to produce and easy-to-use and durable data buoy system. The project also entails the buoy system monitors and records data that can be valuable for commercial users as well as recreational users. The buoy will provide sufficient data for knowledge of the local air and sea conditions. After doing some research, we became more familiar with the implementations of other data buoy systems that have been designed and are being used today. With the information collected, we found that most data buoys used today are large, expensive, or aimed mainly toward commercial side of use. With that in mind our data buoy will resolve all of the aforementioned areas.

For the housing of the system we determined that it will be small and consist of a durable plastic or composite. We have taken into consideration the use of a 3-D printer and designing a custom housing that will be sufficient space for the components and sensors. The use of 3-D printing could give us the ability to have the exact dimensions and cutouts needed for the components to fit as needed. The downside to 3-D printing is that buying a 3-D printer or has proved to be costly in which would require a decent investment. As for private companies that 3-D print objects for people in need, will charge each project by the dimensions (x, y, and z) and the time it took to complete printing as well as the overall weight of the object printed. We have done research some companies that will 3-D print objects but found that a group at UCF has one that will print for a minimal cost. The American Society of Mechanical Engineers (ASME) at UCF is giving us the opportunity to 3-D print the buoy at a cost much less than the other companies we have considered.

The sensor for monitoring and recording the wave height must be taken into consideration as well. Whether using an accelerometer, gyroscope, or an altimeter, we have begun research for each. As a result, we have narrowed our decision down to using an accelerometer, as it will perform the necessary task easily and accurately. An accelerometer is found to be fairly inexpensive as well. The research for the water salinity detection component has been concluded to the use of a laser diode, fiber optic wire, prism, and photodetector. All other feature components of the projects design have been researched as well and will be discussed in a later section on this paper.

Further investigation must be done into solar power, battery power, wind speed sensors, and temperature sensors giving us the information needed to achieve the fully functioning circuit for the buoy system. with the microcontroller in mind, a variety of designs that are on the market today have been taken into consideration and more research on requirements and constraints are being looked into. Additional analysis will be needed to get information on a few other sensors we have in mind, such as wind direction and barometric pressure sensors, in order for us to decide on using and implementing them into the design.

2.0 Project Description

With the continuation of advancement of technology in sensors and circuits, the design of weather data has become vast and can be done with a variety of different techniques making more and more possibilities and features for this type of device. Our project will be designed with the use of the latest technologies making the measurements as precise and accurate as possible. The use of the latest equipment will be utilized as well, including a 3-D printer, which will be used to print all aspects of the data buoy housing itself. The buoy will be printed into a spherical or cylindrical shape with cutouts made specifically for the incorporation of the electronic devices and components. With that being said, this design will apply knowledge gained from most courses to create a programmable, technological, and accurate water condition data buoy. To begin the descriptions, we will look into the constraints that can possibly affect this design as well as related standards benefitting it.

2.1 Constraints

The main goal that we are wishing to achieve is to minimize the total cost of creating the water condition data buoy. Our goal is to have the whole buoy system designed and built for as little as $400 or $100 per group member working on the project. From a quick overview on cost and pricing, the 3-D printing looks to be the largest expense of the total cost, possibly resulting in looking for other options for the buoy housing.

Looking into 3-D printing the buoy housing creates a lot of constraints in itself. The material must be buoyant, durable, and water tight. Doing some research into 3-D printing material, we find that ABS and PLA filaments to be the best suited for our needs. The filaments are durable and lightweight making 3-D printing one possibility for the buoy housing.

The area of use is also a constraint, as the data will only be sent electronically the user’s computer which will need to be somewhat close range to the buoy itself. The data being transmitted wirelessly will have a certain range that can be reached. This also means that to monitor the data being measured a computer must be on hand. This makes monitoring from a boat a little harder to do. As well as the user must actually have a computer.

Another constraint is having the use of electronics and electrical systems in our design. A major downside to our buoy system is the case that we are going to be mixing electronics with water. As most know, electronics cannot get wet or the system could be completely damaged and or potentially harmful to the user. Not only is the system going to be tested and used in water, but the water will contain salt. There is nothing worse for electronics than being exposed to salty water over a period of time. Corrosion occurs very quickly if anything electrical is open or any bare metals are exposed. Whether it has a manufacturer defect or caused by the designer, electrical components can fail, making unsafe conditions from heat or shock. The buoy being in the sun all day could make conditions inside the buoy hot or humid. This can affect the performance of the components as well.

2.2 Related Standards

There are many standards used as of today set by governments, IEEE, or even private companies. Standards are created to minimize the chance of hazards, or to keep the safety and health concern precautions. Though our design is small and only affects a small group of individuals, there are still many standards that are related. With regards to safety, the United States Department of Labor's Occupational Safety and Health Administration (OSHA) reports that, "accidents do not occur under normal operating conditions, but rather during programming, adjustment, testing, cleaning, inspection, and repair periods." (OSHA STD 01-12-002, 1987) This is connected to our project because we will be dealing with electrical circuits where programming, building, and repairing, whereas testing voltages, current and power is unable to be avoided.

802.11a: Transmits at 5 GHz and can move up to 54 Mbps. It also uses orthogonal frequency division multiplexing (OFDM), which is a more efficient coding technique that splits radio signals into several sub-signals before reaching a receiver. This is a technique that is used to vastly decrease interference.

802.11b: This standard is the slowest and most inexpensive. The reason for its popularity was its cost; however, now it is becoming less common as faster standards became less expensive. 802.11b transmits in the 2.4 GHz frequency band of the radio spectrum as well as handle up to 11 Mbps. It uses complementary code keying (CCK) modulation to improve speeds.

802.11g: Just like 802.11b, this standard transmits at 2.4 GHz; however, it is much faster and can handle up to 54 Mbps. 802.11g is much faster as it used the same OFDM coding as 802.11a.

802.11n: This is the most widely available of the standards as well as is backward compatible with 802.11a, b, and g. Compared to the other standards, it has considerably improved the speed and range. 802.11m has reportedly achieved speeds as high as 140 Mbps and can transmit up to four streams of data, each at a maximum of 150 Mbps; however, most routers only allow for two or three streams.

802.11ac: This is the latest standard as of early 2013. It is still in draft from IEEE. There are devices that support it on the market today, yet it still has to be widely accepted. 802.11ac is backward compatible with 802.11n, which essentially makes it compatible with the other standards too, with n on the 2.4 GHz band and ac on the 5 GHz band. It is the fastest compared to all the rest as well as is less prone to signal interference. It can reach a maximum speed of 450 Mbps on a single stream.

ANZI Z136.2: Safe Use of Optical Fiber Communication Systems Utilizing Laser Diode and LED Sources.

This standard provides guidance for the safe use, maintenance, service, and installation of optical communications systems utilizing laser diodes or light emitting diodes operating at wavelengths between 0.6 mm and 1 mm. Optical communication systems include end-to-end optical fiber based links, fixed terrestrial point-to-point free-space links, or a combination of both.

ANSI Z136.6: Safe Use of Lasers Outdoors

This standard provides guidance for the safe use of lasers in an outdoor environment, e.g., construction, displays/laser lightshows, scientific/astronomical research, and military (DoE/DoD).

ANSI Z136.8: Safe Use of Lasers in Research, Development, or Testing

The purpose of this standard is to provide guidance the safe use of lasers and laser systems found in research, development, or testing environments, where safety controls common for commercial lasers may either be missing or disabled.

ANSI C18.2M: Portable Rechargeable Cells and Batteries – General and Specifications

Our project includes a rechargeable battery that powers the entire system. Given this, we have decided to include this standard. The purpose of this standard is to provide general details and specifications for portable rechargeable batteries.

UL 1642: Safety of Lithium-Ion Batteries – Testing

Whenever dealing with lithium-based batteries, we have to be extra careful because these batteries can be toxic and dangerous if abused. The purpose of this standard is to provide a guide to safety of lithium-ion batteries whenever a series of tests have to be conducted on one of them.

ISO 50001:2011: Energy Management Systems – Requirements with Guidance for Use

The purpose of this standard is to provide a guide for the use of energy management systems and a detailed requirements specification list whenever used.

ISO 9488:1999: Solar Energy – Vocabulary

The purpose of this standard is to simply provide a list of vocabulary as to have a better understanding of the concepts utilized whenever dealing with solar energy.

2.3 Motivation

Having a design group consisting of two electrical engineers, one computer engineer, and one photonics engineer, our motivation was to come up with an idea that will test our understanding and incorporate the knowledge acquired throughout our educational careers, combined with finding something that is associated with all of our interests. The thought of designing the fully functioning water condition data buoy is more difficult and complex than it seems to be. To simplify our outlook on the buoy design project we separated the concept into four main aspects of this particular project:

- Physical Aspect
- Electronics Aspect
- Photonics Aspect
- Programming Aspect

The physical aspect will be the buoy housing design and the how the components are going to be integrated in the system making it space efficient for its portable design. After doing some research, the physical aspect will be very time consuming with research to gain a depth of knowledge about compactness as well as buoyancy of an object. If the housing is going to be printed in 3-D, the person in charge of the housing design will need to create a sketch in a software that is compatible for 3-D printing, such as Solid Works. The sketch will account for the total shape and dimensions of the overall buoy housing. These dimensions are only the beginning as for the designer will also need to account for size and shape of electronics, wires, and any other components that will be integrated. Being that the main shape of the buoy housing will be mainly spherical or rectangular, a custom design shouldn't be too difficult. There will be no need to use any open-source designs or get permission from anyone to use their design.

The electronics aspect is what we figure will determine the overall capabilities and features that can will be incorporated into the buoy system. To begin, we must choose the microcontroller best suited for our intensions. After coming to an agreement on a specific one, we needed to draw out how the other electronics components and sensors were going to be implemented off of it. Being that the microcontroller is essentially our "brain" of the system, we must design and build from there. Once the components are implemented with the microcontroller, the Micro Controller Unit (MCU) is obtained. The MCU is a chip that has multiple input and output pins to connect to different electronic input and output components, creating a bridge of communication between itself and the other components, which will later be embedded into the design.

To make the MCU work, it will require an external power source; whether a few alkaline batteries (such as 9V batteries), portable electronic device batteries (such as cell phone batteries), lithium ion (Li-Ion) batteries (such as laptop batteries), or lithium polymer (LiPo) batteries (such as remote controlled vehicle batteries). The batteries used will take a charge by the use of a solar panel during sunlight hours and discharge while the solar panel is not being used during night time hours. The microcontroller will be used to control everything, requiring it to be transmitting/receiving analog as well as digital signals. All of the electronic sensors and components will, at some point during use of the device, need to communicate to and/or from the MCU. The MCU will need extensive programming to be able to make all the components work simultaneously as well as deliver an accurate and readable output that will be interfaced to the user's computer. Putting all the electronics components together (MCU, sensors, fiber optics, solar panel, battery, switches) we should receive our complete and final design of a working system.

Looking into the photonics aspect of the project, the views are similar to the electronics aspect. While there are options available commercially for buoys that detect many of the same data we hope to record, it seems there is no current industry standard for measuring salinity or contaminants in the water. In the hopes of creating a new useful module and an overall useful device we are adding photonics aspects to measure the salt content. We must choose a laser diode preferably in the visible spectrum for ease of alignment and correction. Optical fibers will have to be selected, cut and fused to proper length to attach to the system. The whole system will have to be aligned to emit into the fiber array at the output so we obtain a good image on the CCD camera for our data.

As for the programming aspect, we have a determined computer engineer willing to take on the challenge of this project. The electrical and photonics engineering members, though not best suited for this task, have also taken several classes making them able to handle simple and less complex coding and implementation. The reason for programming being such a major portion of the project is because without some sort of code, the microcontroller will have no commands of what to do. The microcontroller will need intensive written code being that we will need the MCU to handle converting digital and analog signals. The MCU will need to be able to know when the sensors are communicating as well as be able to translate the input and output voltages into understandable values such as Volts to degrees Fahrenheit for the temperature sensors. Furthermore, when we look into the wireless interface between the actual device and the user's computer or table, this requires some programming as well.

With all four aspects taken into consideration, combining them creates our overall vision of how this design project will be created. The motivation gives us a general idea of where the design starts, what needs to be incorporated, and what our overall achievement should be.

2.4 Goals and Objectives

Our main goal of this design project is to obviously have a completed, fully functioning water data buoy system. We put together a set of smaller goals to complete and achieve the main goal of this design. Throughout Senior Design I and II we will work toward these objectives and keep track of our progress along the way. The goals we set are as follows:

- Buoy Housing
- Water Salinity Detection
- Wave Height Sensing
- Air/Water Temperature Sensing
- Wind Speed Sensing
- Solar Power Charging
- Power Supply
- Printed Circuit Board and Microcontroller Unit
- Wireless Interface to Computer

As for the buoy and what the housing will be made of, we took into consideration the amount of time it would take to either use something that can be bought and built out of materials from a local hardware store or design and draw a schematic to have it 3-D printed. To design something that we can build from materials found at the local hardware store would be a quick process but may be hard to implement a sufficient and durable housing. On the other hand, finding a 3-D printer and having a company or organization print a design can be quite time consuming. Between designing and waiting for the printing, 3-D printing can take weeks and even months. Both options have positive and negatives concerning time and goals. In terms of goals a basic housing needs to be selected and testing in the ocean to make sure it can be made to float and stay waterproof. Modifications need to be made to make sure the device can be continued to be worked on easily once the components are placed into the housing. Waterproofing must be completed to make sure all internal components do not get wet when the device is being used.

The salinity sensor will consist of a laser diode, fiber optic cables, prism, and CCD camera. It will also require a housing for the sensor itself to hang underneath the main body of the device to allow water to pass through the sensor to be tested.
Goals for this sensor will start with testing the basic premise in lab. Then creating the housing and placing it into the buoy to align and test as a whole. Then the output must be read into the programming in a way that can be easily interpreted by the user.

The Wave Height Sensor will consist of a pre-built circuit board that will be added into our own PCB. This circuit will consist of an accelerometer or a gyroscope, which will be determined once we have done more research into each. This part of the system will be an advantage when dealing with the time given. The most tedious area of this feature will be programming and testing the sensor itself. The sensor will need to be able to average a set of distances over a period of time, which we will need to construct a certain algorithm to do so.

Air and Water Temperature Sensors are pre-built components that can be bought off the market as well. The temperature sensors will consist of either a temperature transistor or thermistor based on which will be more accurate as well as easier to implement. The most time consuming area will be in the testing and programming to receive a readable output.

Wind Speed Sensing can be done with a simple pre-built component as well, called an anemometer. The anemometer will measure an analog signal that will vary in voltage as the wind speed changes. This feature will be simple to implement but can be time consuming in trying to test and program.

When we look into the power supply, we will determine first which type of battery best suited for our system. Considering we will be using solar power to charge the battery, we must also look into a solar panel and how it will be integrated into the charging system. The battery must be rechargeable and be able to produce at least 9 Volts for all the sensors that will be used in the system. We also took into consideration the weight, as for we do not want to overload the buoy and make it too heavy. The charge times and capacities will also be a key factor as to which battery and solar panel we decide to use. The last criteria we used to determine these two components are the overall physical size and how much room in the buoy they will be taking up.

The Printed Circuit Board (PCB) and Microcontroller will be a major factor in our buoy system as for it will be the brains to our design project. The key factors we will be looking at to decide which microcontroller we will be using are number of input/output pins, program memory size, timers/counters, and supply voltages. The microcontroller will need to have a sufficient amount of input and output pins, analog and digital, to handle all of the sensors listed previously. Also, being that the microcontroller will need to be able to monitor and record data, as well as average certain measurements over a period of time, the memory must be large enough to store an ample amount of data. The supply voltage will be factored in once we have decided each sensor needed voltages and determined the exact ones we will be using.

Wireless Interface to a computer will be another goal that we are determined to be an addition to our design. Once the buoy has begun monitoring and recording data, the user will need a way to see the data that has been collected. Using wireless communication to monitor the conditions of the data buoy would be most practical and efficient. This portion of the project will be somewhat of a difficult task and may take some time to program and implement. This method makes for convenience and ease. Through a computer program or website, the user will be able to open a window that will show all the data collected in an organized and easy to read format. This portion of the project will be somewhat of a difficult task and may take some time to program.

2.5 Requirement Specifications

When designing a water and air condition data buoy, we must first determine a few requirements that the components and housing must meet. Throughout this section we will explain in a little detail the specification requirements of each component followed by an easy to understand table. Here are the components and there required specifications:

When considering the Air Temperature Sensor, we figure that it will be required to measure as low as 0° Fahrenheit and up to at least 150° Fahrenheit. We set the required upper temperature high being that we live in Florida and temperatures are scorching. The sensor is also required to be waterproof in case of bad weather as well as waves that come over the buoy. The sensor must be small enough to be fixed to the top of the buoy housing. The wires from the air temperature sensor will need to be long enough to reach the microcontroller.

The Water Temperature Sensor will have basically the same requirements as the air temperature. It is required to measure as low as 0° Fahrenheit and up to at least 120° Fahrenheit. We chose this temperature range because it covers anywhere from frigid to tropic water. The water temperature sensor must be waterproof as well to handle rain and water that comes over the buoy. The sensor must be small enough to be fixed to the bottom of the housing where the buoy will be underwater. The wires must be long enough to reach from the mounting position on the buoy housing to the microcontroller inside the housing.

Looking at the required specifications for the wind speed sensor, it must be able to measure from no wind or 0 miles per hour up to at least 60 miles per hours. This range of wind speeds is suitable for the average location. It is not to be used during hurricanes, tornadoes, or any other natural disasters. The wind temperature sensor is required to be small enough and mountable so that it can be fixed to the top side of the buoy housing. The wiring from the wind speed sensor must be long enough in order to reach the microcontroller from the top of the buoy housing.

Another requirement for our data buoy is that it has to be able to record wave height measurements and transmit this data to the user. This wave height sensor has to be able to measure displacements of up to 10 feet and be as accurately as possible in its measurements, most likely readings that approximate the exact displacement measurement to one fifth of a foot. This wave height sensor does not have to be waterproof because it is most likely going to be placed on the interior of the buoy housing. It has to be small enough to fit inside the buoy housing together with all other components required for our projects overall functionality. However, it must be able to perform over wide temperature range because our data buoy is intended to be used outdoors. Being so, the interior of the buoy housing might reach high temperatures after being exposed several hours under the sun.

Given that our data buoy is intended to be used outdoors in a body of water, we have made one of its requirements for our project to be powered by using solar energy. In further sections we will be taking a look into how this can be achieved, different methods will be discussed, and a decision will be made. One important characteristic for this component is that it has to be waterproof because it will most likely be attached to the exterior of the buoy housing where water can easily reach the component. The dimensions of the solar panel have to be carefully, it has to be big enough as to cover as much surface area as possible but small enough as to fit on top of the buoy housing without causing any issues. By doing so, we would be designing a data buoy that is self-sufficient and environmentally friendly.

The salinity sensor must be able to detect set levels of change in salinity between pure water and salt levels found in ocean water. The levels will be measured as a percentage of the total volume of the water in part per million with set increments to be minimally detectable. The sensor will be closed to contain the emitted light from the laser diode but will have holes on the sides to allow the water to flow through the sensor to obtain an accurate reading.

For the Printed Circuit Board, we will need to make some required specifications so that we will have sufficient space for all the components needed as well as low cost to keep the overall budget down. When determining the required size of the buoy we must take into consideration all the components that will be mounted on the board itself. The board will be a place for us to mount our Microcontroller, a few passive components, and output pins for our sensors to be connected. Since the PCB will be mounted inside the buoy, it must fit through the hole mounted on the top of the housing, which is 5 inches in diameter. With these things in mind, we can keep the size generally small. The required size of the PCB must be less than 3”x4”. When determining the cost, we are going to be keeping it in a low range being we have a self-funded budget. With the design and printing taken into consideration, we are going to make the overall cost for our PCB under $75.

Also, we will need to set some specifications when looking into the overall buoy housing. The three main requirements for the housing are going to be size, water resistance, and cost. First, when determining the size of the buoy housing, we will be taking all the information from the previous required specifications and combine them to find out how much volume we will need internally as well as surface area we will need externally. After adding all the components sizes and specifications, as well as keeping the overall size fairly small, we have decided to set the buoy housing dimensions to be a maximum of 20” L – 20” W – 20” H. Next, being that the buoy will be used in the water and also have the chance of foul weather and rain, we will make the housing of the buoy waterproof, so that any internal components that are not waterproof, do not get wet. Last, we have decided that to keep the overall design in our budget, the buoy housing cost will be kept reasonably low. If we decide to use our own material and buy something on the market, we will keep the cost under $25. If we decide to go with the 3D-printed buoy housing, our cost requirement will be under $100.

Finally, our data buoy should be able to transmit data recordings wirelessly to the user over a distance of at least 20 feet. This device should be small as to be placed inside the buoy housing and must be able to perform over a wide temperature range like the wave height sensor.

2.6 Block Diagrams

Now that the project concept and requirements have been explained we will go into a visual perspective along with in-depth details on how the buoy system and its components will be assembled.

The temperature sensors will be fixed to the buoy housing itself, protruding a small portion of the sensor enough to contact the sampled area. It will use a simple communication from the microcontroller to send a voltage and the temperature sensor will use a signal wire to send back digital signals that encodes the necessary information. The digital signal sent back from the temperature sensor will be converted from its binary format to a temperature in degrees Celsius and then to Fahrenheit. A block diagram showing how the sensors will receive and transmit the data is shown in Figure 2.6.a.

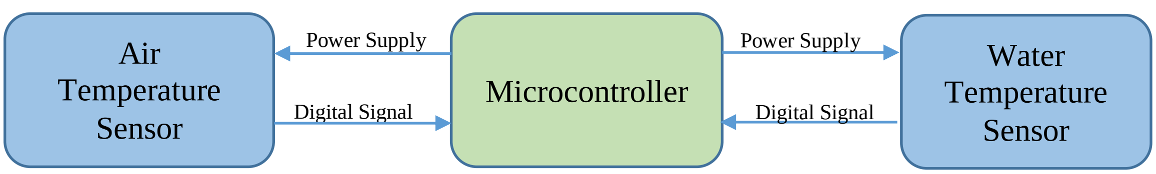


Figure 2.6.a. Data Transfer Between Temperature Sensors and MCU

The wind speed sensor will have a similar structure although it will be an analog signal that will be sent back to the microcontroller. The wind speed sensor will be fixed to the top of the buoy housing. It will communicate through a power wire sending voltage from the microcontroller to the sensor. The sensor will return an analog signal back to the microcontroller in which will be converted from voltage to temperature in degrees Celsius and then Fahrenheit. The following block diagram, Figure 2.6.b, shows the data transfer between the wind speed sensor and the microcontroller.

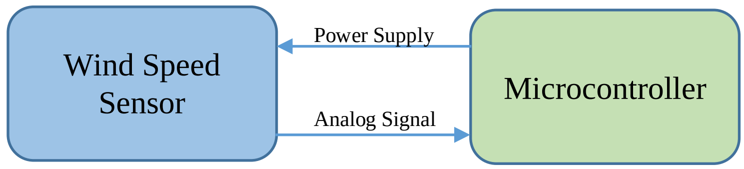


Figure 2.6.b. Data Transfer Between Wind Speed Sensor and MCU

The wave height sensor is similar to the air and water temperature sensors and the wind speed sensor in that it requires a supply voltage to perform the specific task of measuring heave produced by a particular wave and sending an analog/digital signal back to the microcontroller unit. Once the microcontroller receives this signal, it is to be analyzed and converted for the user to be able to interpret the data. Depending on whether the wave height sensor sends an analog or digital signal, the microcontroller will be programmed in order to successfully convert the signal into useful data. Figure 2.6.c illustrates the interaction between the wave height sensor and the microcontroller unit.

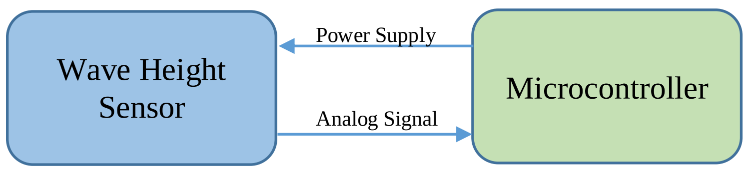


Figure 2.6.c. Data Transfer Between Wave Height Sensor and MCU

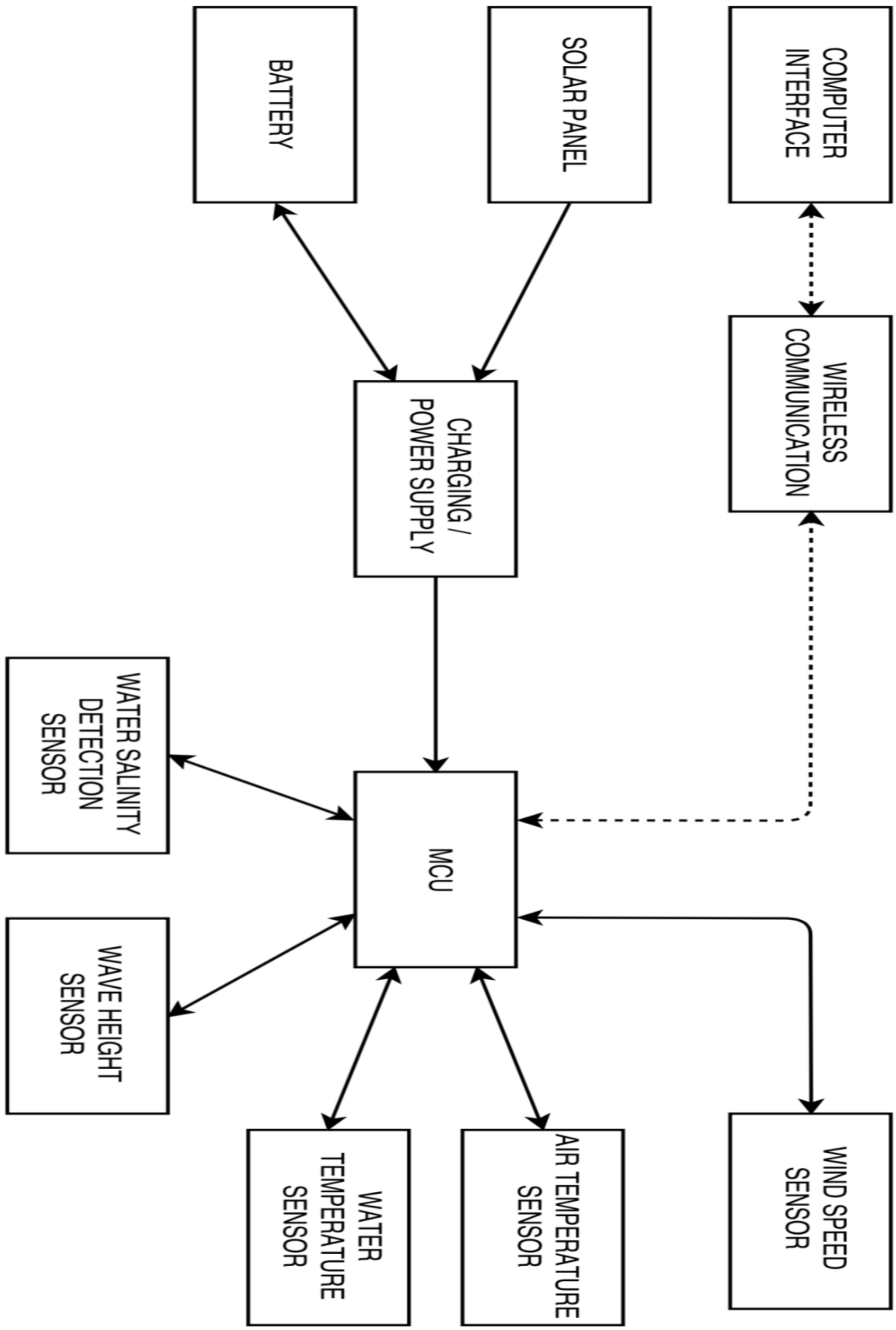


Figure 2.6.d: Overview of entire data buoy system

3.0 Background

According to Merriam-Webster dictionary, engineering is defined as “the application of science and mathematics by which the properties of matter and the sources of energy in nature are made useful to people”. Three engineering major students and one photonics major student have come together to collaborate with each other and share our knowledge in order to be able to come up with a project idea that will fulfill and test what we have learned over the past years. More specifically, two electrical engineers, one computer engineer, and one photonics student. Let’s take a quick look at each field.

Merriam-Webster dictionary simply defines electrical engineering as “a type of engineering that deals with the uses of electricity”. To be more specific, electrical engineering is a field of engineering that usually deals with the study and application of electricity, electronics, and electromagnetism. Within this engineering discipline there are other fields that have a specific focus on a given field. These subfields include: electronics, power, computers, instrumentation, telecommunications, control systems, signal processing, microelectronics, etc.

Now, computer engineering is a field of engineering that combines the fields of electrical engineering and computer science into one. This engineering discipline focuses on developing computer hardware and software. Our computer engineer will be in charge of coding and programming our microcontroller so that a smooth interaction between the user and our project can be achieved.

Finally, photonics is a relatively new career choice here at University of Central Florida. As stated by Merriam-Webster dictionary, photonics is “a branch of physics that deals with the properties and applications of photons especially as a medium for transmitting information”. We will be introducing photonics into our project by utilizing laser diodes, CCD cameras, prism, amongst other things to detect water salinity levels in a body of water.

3.1 Electronics

Electronics is defined as “the science dealing with the development and application of devices and systems involving the flow of electrons in a vacuum, in gaseous media, and in semiconductors”. Most of the components that will be used in our project design rely on electronics. Given that, it is important to have a good understanding of its backgrounds and how they function physically.

In the simplest of terms, electrons flow from on point to another in order to create electricity and electric signals. For our project, we will be using multiple sensors in order to record measurements such as water and air temperature, wind speed, wave height, and water salinity. All of these sensors rely heavily on electric signals –analog or digital– and output these analog or digital signals for the microcontroller to read, interpret, and communicate to the user as understandable data recordings. In the following sections we will be discussion theoretical background for some of our components in order to have a better understanding of their functionality.

3.1.1 Theoretical

Our group is composed of two electrical engineers, one computer engineer, and a photonics student. As students, we have all taken many similar courses but also many other specific to our career paths. We each have contributed to this project by working together on the different aspects of our project while sharing valuable knowledge with each other.

We will begin by discussing the photonics part of our project. Photonics is the field revolving around light which includes imaging, lasers, fiber optics, and many other aspects. Light is considered to be an electromagnetic wave for use in most physics concepts as it has forces on both electrical and magnetic bodies however this does not explain situations where light instead acts as a particle so a new definition is needed. Physicists now think of light in terms of photons which allows for a measurable and calculable amount of light in each situation. A photon is a “packet” of light that can behave as both a particle and a wave depending on the circumstances. Where electrical engineers deal with electrons, photonics engineers deal with photons in many forms whether though color and the visual system, or for communications, or even for the use of creating laser light. Photonics engineers integrate light into electronics through optoelectronics, such as a light detector or emitter, or an entire fiber optic communications network that can be orders of magnitude faster than simple electric wiring due to traveling at the speed of light. While there are drawbacks to the use of optoelectronics in terms of cost and maintenance, the speed and resolution more than make up for the drawbacks.

Laser light is simply light that is Coherent. Coherence is the light all having a similar wave form in terms of period, amplitude and frequency. Laser light is created using a cavity which only allows a thin band of wavelengths to survive and emit. Since all of this light was created in the same cavity from the same medium it is able to be coherent as long as it does not exceed the coherence length of the laser. This is very different from sunlight for example that is incoherent so the intensities do add up if you have a lot of light, but unlike coherent light it does not multiply intensity. Laser light is extremely useful for many applications and can be modified to fit a given situation, such as for use in a medium that is very absorbent there is usually a wavelength or a few that are less absorbent and that wavelength can be used to measure properties of the medium.

In physics we deal with electrons and photons which have a charge and affect the area around them, where light behaves as both a particle like subatomic particles and as a wave. The photons or “packets” carry a certain amount of energy which affects the area around the light. The energy from this electromagnetic wave interacts with materials by being absorbed, travelling through, or reflecting off of them and in doing so transfers its relative energy to the particles in some way. The refractive index of a material is the ratio between the speed of light in a vacuum versus the speed of light in the material. Water has a certain refractive index when it is distilled or pure water versus when there is anything dissolved in the water which changes the material properties. The reason the speed of the light through the material changes is simple if you think about it, if you had a clear glass and shined a light through it then it would appear very bright but if the glass were smudged or even painted the light would appear dimmer or not at all through the material. In a similar way shining a laser light through the material we can measure the change in the beam divergence through the material which can be calculated and used to determine the salinity of the water. The beam divergence is directly proportional to how much extra material is in the water much like the clear glass versus the smudged glass.

In the environment of course there are other materials that can cause a refractive index change in the water and as such the sensor would be an approximation outside of the controlled environment. To obtain a more exact answer would require the use of a different kind of device such as a spectrometer which can read the individual spectrums of all the materials in the water by subtracting the spectrum of water itself and comparing what is left. However, the price and size change to implement such a system is beyond the scope of our project and needs.

Further, the electrical and computer part of our project deals mostly with electronics. Along the last couple of years, several courses have been taken in which the basics and fundamentals of electronics have been taught to us. Elements such as input voltages, resistors, capacitors, diodes, amongst others have been introduced and we have learned how to simplify and understand these components as to determining series or parallel combinations. Many different combinations can be created between these elements in order to create circuits designed for specific purposes. We have learned the concepts of current and voltage division along with techniques of circuit analysis such as superposition. Next, amplifiers were introduced along with different methods on how to calculate or even set the desired gain. We have also learned about transformers. As the name suggests, these are used to transform something. In this case, a transformer is one that transforms a voltage. Whether a voltage is required to step up or step down, a transformer is the one in charge of performing this task.

All of the previously discussed elements and techniques will probably be required at some point while designing our project and making it work properly. Whether we need a voltage to be regulated, an analog signal to be converted into a digital signal for simplicity, or producing an amplified signal. Many different combinations can be made between these components as to design circuits with specific purposes.

3.1.2 Images and Schematics

In previous sections we introduced the three disciplines that have come together to design the data buoy. Also, theoretical background for each discipline has been discussed. In the following section, we will be taking a look at different circuits that may be used for the overall functionality of our design.

We will begin with analog and digital signals. In our project, we will be using many different sensors in order to record measurements such as wind speed, air and water temperature, wave height, and water salinity. As previously mentioned, these sensors all produce a signal that will be received and interpreted by the microcontroller which then will transmit to the user. These signals are some analog and some digital and we might require converting one to another for simplicity purposes. More specifically, converting an analog signal into a digital signal.

Further, sometimes it is required to have an input voltage regulated. In our design, many different sensors will be wired to a microcontroller board. These different sensors all have different requirements for input voltages, some may be high will others low. To solve this issue, we will be need a voltage regulator. Figure 3.1.2 illustrates how different electronic elements have been combined to create a linear voltage regulator circuit. The output voltage is easily adjusted by simply selecting the appropriate R_1 and R_2 values.

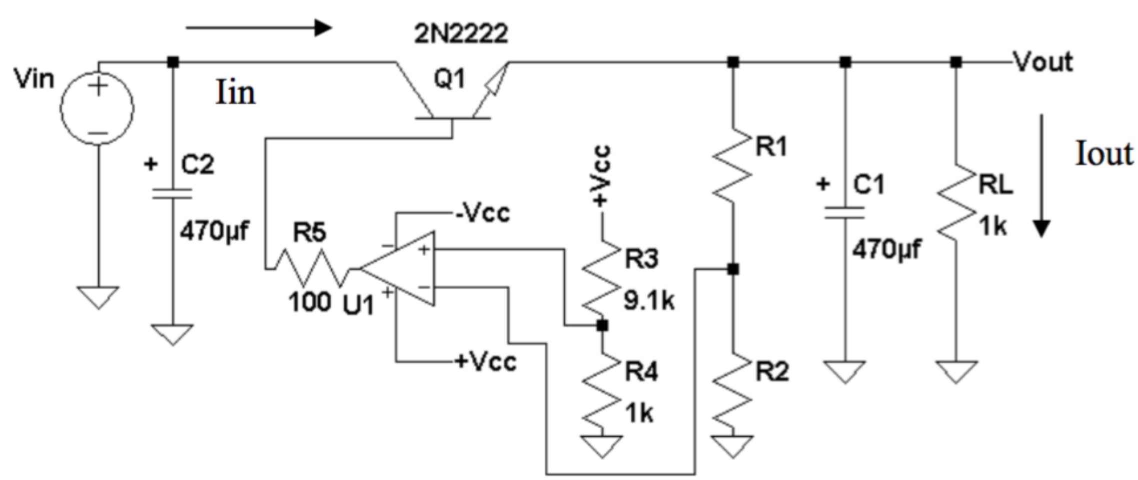


Figure 3.1.2: Linear voltage regulator circuit

3.2 Current Code

When looking into the different types of sensors, components, and microcontrollers, we will also be deciding what we will need for programming them. In order for the data buoy to function as intended we will need the code to be programmed to do the following actions:

- The program must read and send any data being monitored by each sensor when updated.
- The program must save any data to the internal memory on the microcontroller for any sensors that need averaged.
- The program must send and maintain a real time and date stamp with the recorded data whenever updated
- The program must be able to wake the microcontroller and let it know when data is needed to be transmitted to the computer
- The program must continuously check for any actions required or updates from the computer to send the data.
- The program must be able to put the microcontroller into low power or sleep mode when it is not needed for transferring data

With these specific requirements we will be able to have a system that can work on its own as well as be controlled by the user. If the buoy is requested to send data by the user, it will transmit the most up to date measurements recorded by the buoy.

3.2.1 Program Flow

In this section, we will put together some information and objectives for how we feel the program flow will be intended to work. We will discuss the list of what the system will be needed to do as well as create a flow chart showing how the system will work.

To begin, the data buoy will be set in a low power or sleep mode until it is woke by an interrupt from the user. The system will then take any of the data it has recorded for the sensors that need averaged and make the calculations. This will happen simultaneously with when the other sensor will take their readings. Once all the sensors have the data recorded, the microcontroller will send the data to the user via wireless communication to the user’s computer. After being sent the microcontroller will wait until any further instructions or actions are requested. If after a certain time of no user requests, the microcontroller will be put back into the low power or sleep mode until it is awoken again and further actions are required. The following flow chart will show the programs actions during waking the microcontroller all the way through going back into sleep mode.

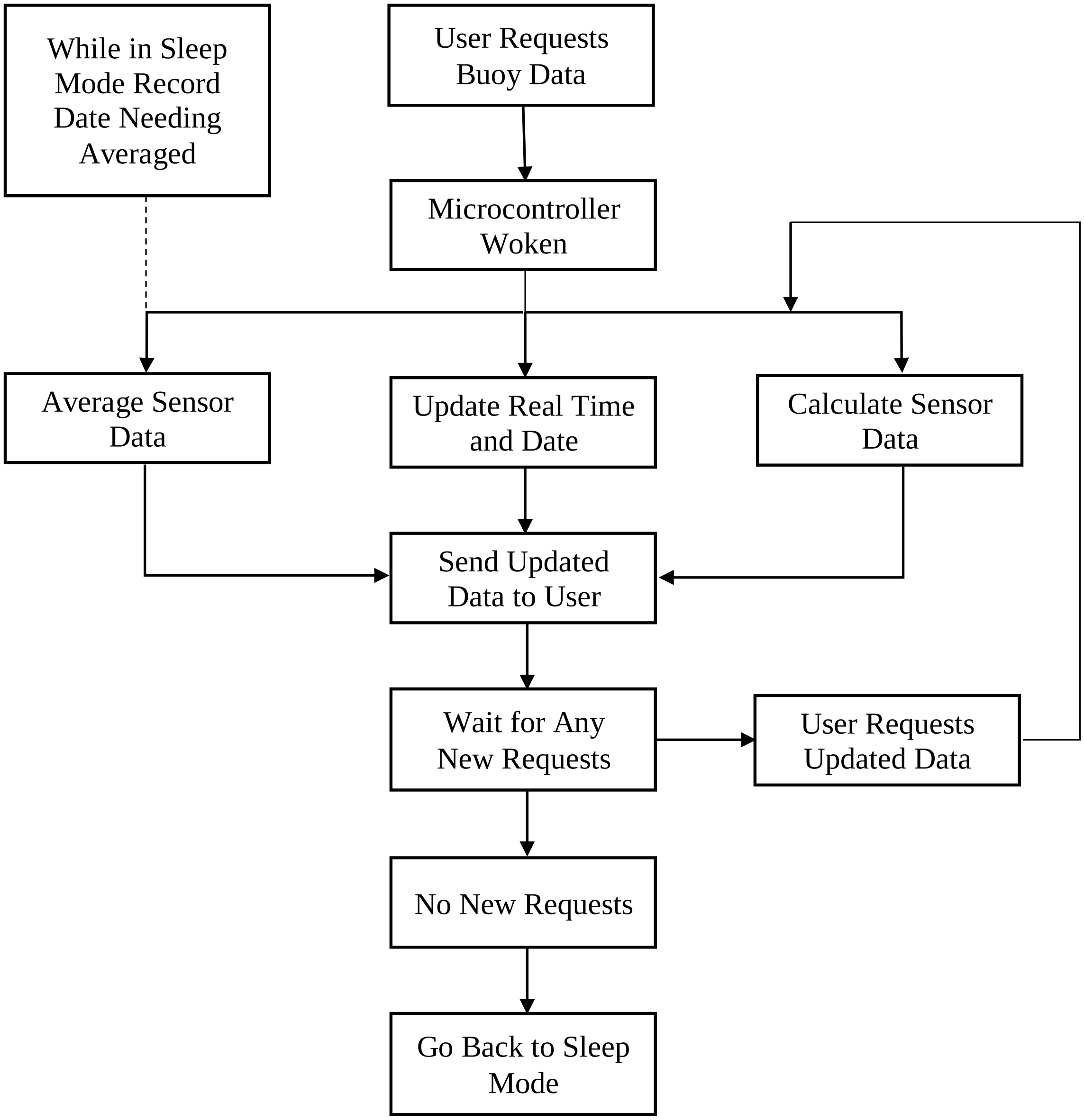


Figure 3.2.1.a. Program Flow of Data Buoy System

3.3 Current Usage

Water and air conditions have been monitored for centuries through the uses of mechanical and electrical devices. With technological advancements, measurements in these areas are improving and becoming much simpler of a task for most people. From television to car radios, and now even cell phone apps, it only takes seconds to determine what the weather is like outside. On the other hand, there are few devices out today that can give all the features of local water and air like the data buoy system we are designing and creating. Intended for anyone who has a passion for the water or possibly a job consisting of boating, this buoy makes checking conditions very simple. This device will allow anyone who has the buoy or is in the local surrounding area to view its multitude of valuable information. From wave height to water salinity, this device will be valuable to any water sports enthusiast.

Multitude of features: from wave height to salinity detection, this buoy will have the key features to anyone who needs to check water and air conditions. The buoy will consist of two temperature sensors. The first of which will be used for measuring the surrounding water temperatures just below the surface. The second will be used for measuring the air temperature above the buoy and nearby. Another sensor that will be used is the wind speed sensor. This will be mounted to the top of the buoy as well and monitor the speed of the wind. As stated, the device will measure and record wave height too. This will give anyone the ability to determine if the waves are too big to be boating or too small to be surfing. Last but not least, we have added the salinity detection. Through which, the local fisherman can determine what type of water is in the lake or river they are in, which lets them know what kind of fish they will be catching.

Transportable: The buoy will be designed with a dimension small enough that it is portable and can be transported wherever it may be needed. If the user needs to bring it along to different lakes or rivers, then there will be no hassle. The buoy housing itself will only need to be large enough to fit all the needed sensors and components, thus making the overall size small and compact.

Durable: Being that the buoy will spend most of its lifetime in or around water, we have designed it with durability in mind to minimize the chance of any damage that can be caused. The buoy housing must be able to withstand vigorous waves when the water is rough. This entails having a suitable and very waterproofed system inside and out. Also if the buoy is around any structures, the housing must be strong and solid, making it capable if any contact occurs.

Long Battery Life: The battery we are currently researching and electing to use will have a long battery life and slow discharge rate due to the low power and sleep modes in the microcontroller. During use the microcontroller will be recording data at certain period of time, making it be able to use the low power mode in between samples. Wireless transmission can be power consuming, so we decided that only when the user opens the window to view the data, the microcontroller will understand and transmit the recorded data. The largest contribution to the low power system will be that we will have a solar panel implemented into the system to keep the battery charged.

Affordable: Unlike most data buoy systems being used today, this design will be cost efficient and take minimal funding during ownership. Being that the buoy will be personally owned and operated, there will be no costs tied to the device. For our goal of being able to design this system for less than a few hundred dollars, this can be a valuable product for its price.

After looking into a few of the major benefits of our data buoy system, we can clearly see that in comparison to many data buoys today, this device will be an improvement towards recreational and personal use. Many of the devices seen today are massive, excessively costing, and non-ideal for anyone other than a commercial company or organization. We have made it our goal to design a more down to earth data buoy for the average person to use.

3.3.1 Future Development

We have added many valuable features into our data buoy system and plan on working what we have for now. Although, we have determined that there will always be small fixes and bugs that will need worked out of the first design. From the buoys housing to the sensors that are being used, there are multiple ways of altering and improving them. Also, the user interfaces can be advanced. If in the future we decide that we would like to do a rebuild and improve on any part of the data buoy, we believe that the following areas will be looked into for more research, better implementation, and alterations:

Smaller/Efficient Buoy Housing: While looking into the buoy housing itself, we have come across multiple different ways to implement. For the first design we are going to go with what we decide after researching more. Although, this may not be the best possibility for our data buoy and there could be smaller and more efficient ways of designing and building. Therefore, for future development, the buoy housing could be minimized after realizing and determining how much space we will really need and what things can make the build more efficient. The buoy only needs to be big enough to support the sensors being used and capable of still being buoyant. Also the material it is made out of can be a major factor into a second design.

More Water and Air condition features: After looking into minimizing the size and maximizing the efficiency of the buoy housing, we also determined that a second buoy design could have more sensors improving the capabilities of the observations made by the buoy and the person using it. Although, hand in hand, if more features are added and the buoy becomes more complex, the buoy housing size will increase while the efficiency will decrease. Some features that come to mind that can be added into a future buoy design are: wind direction, wind gust, barometric pressure, precipitation, humidity, wave period, and wave direction just to name a few. These are just a few more features that can be implemented into a future designed water and air data buoy system.

Stronger Durability: When determining how vigorous our design is, there are a few features that increase and decrease the durability of the overall buoy system. In a future design we have determined that with more sensors sticking out of the buoy housing, this decreases the durability of the system itself. An alternative is to find certain sensors that can handle multiple features. Also, in a future design we could look into material that would work best for durability to lengthen the lifespan of the buoy. We are deciding on using a 3-D printed housing or a plastic material that will be sufficient for supporting al the sensors and electronics inside. In a future design, there is a possibility we find that a certain type of metal or composite works better.

User Interface: Improving the user interface can be another possibility for future development. Practically everything can be accessed with a smartphone or app today, having a data buoy application would be a valuable feature. These are all improvements that can be taken into consideration for future development. The more research and comparisons between parts and components done, the more sufficient and efficient data buoy system we will have designed.

4.0 Research

Throughout the research section of this report we will explore the various options available for the design of each component that make up this system of a data buoy. This is where all of the theoretical and abstract concepts that have been mentioned previously will be put together to form what will be a fully functioning device. We will be researching and deciding on the most suitable components to be an addition to our project. Starting with multiple options being extensively researched, discussed and compared, we will choose the most reasonable and applicable one. Form their we will emphasize on how we made our determinations and our intentions on implementing them into the design. With the use tables, diagrams, and images, this section should be clear and understandable.

4.1 Buoy Housing

The Buoy Housing must be large enough in dimension to accommodate the components inside, have space for the solar panel on top, and still float on the water. The Buoy must be waterproof and most likely will have to be painted to reduce the amount of heat absorbed to reduce the core temperature of the device. The housing should also be easily modifiable if components need to be added and lightweight.

4.1.1 Dimensions

The Dimensions for the housing are going to have to be as small as possible to make sure the device will float but allow enough surface area to place the solar panels and allow the device to receive enough power. When researching other similar data buoy systems, we can see that most of them are fairly large in size. This is the case because there are many features and functionalities that are involved.

To begin the buoy system must have enough surface area to be able to hold solar panels and be buoyant enough to float. The internal volume must be large enough to hold all the wiring and components without being too compact. The overall dimensions we decided for the requirements are pretty close to the dimensions we have determined we will be using.

4.1.2 Materials

Material options for the housing are differing kinds of plastic as the light material will allow the buoy to maintain buoyancy where a metal housing would be difficult to keep afloat. Using wood or other similar materials for their buoyancy properties is not preferred due to the device needing to be waterproof and most materials are absorbent. The best option for a material is to find a housing of some kind already on the market that is waterproof as is, then make customized changes as needed.

An on the market oil pan or similar would make an ideal first purchase and could have enough surface area for the solar panels to be placed on.

Some other materials needed will be used for waterproofing the buoy housing itself once the components are added. The housing will need to have some holes made in it to be able to mount the components. Once holes are mad, the buoy will need to waterproof any such areas. To waterproof any of the holes made we will need a marine or water tight caulking. Once all the components are sealed, we must look into making the cap watertight as well. The cap will be simply sealed by just applying the correct dimension O-ring to keep water from being able to come through the plastic threading.

4.1.3 Layout / Appearance

The layout for the buoy will be a sealed container with a top that can be removed so components can be placed and maintained without interfering with the placed solar panels:

Figure 4.1.3.a: Buoy Housing Side and Top View

The buoy body is relatively simple in design and must simply be able to house the components while still allowing access, and when sealed must remain waterproof to meet the standards for the finished product. Several components will have to be accommodated by making holes in the buoy but will be sealed and waterproofed as needed to maintain the necessary integrity of the housing. As the housing is black and will spend long periods in direct sunlight we will most likely be painting the buoy to reduce the amount of heat absorbed.

4.2 Power Supply

One of the most essential components to our project is the power supply. Each and every single component of the Water Condition Buoy System requires this power supply in order to function and provide us with the desired data. Several methods for providing the required power to a system exist. However, given the mobile nature of the Water Condition Buoy System, it is most likely that a battery will be utilized to provide the necessary power for the system to function all together.

4.2.1 Battery

Nowadays, batteries are essential in everyday life given that they are the main power source to many household and industrial appliances. In general, batteries can be divided into two major categories, primary batteries and secondary batteries. The former type of battery is a disposable battery, meaning that once it is used it cannot be used again or recharged. The latter, on the other hand, is rechargeable. This means that they can be reused just by charging them again. Further, within these two categories different types of batteries exist and they are classified by the chemistry behind its technology.

4.2.2 Battery Technology

Primary batteries:

Primary batteries are very useful when charging is not practical o very difficult to do. These batteries come in a vast variety of sizes and shapes and are typically used for remote controls, wrist watches, children's toys, flashlights, and many other things. The standard used for primary batteries is IEC 60086-1, BS 387. The most common or widely used primary battery is the alkaline battery. In the following paragraph some characteristics of an alkaline battery will be mentioned.

Alkaline battery: An alkaline battery is a non-rechargeable, cost effective, and environmentally friendly battery with a high specific energy or capacity. These batteries are very portable and ready to be used whenever needed. Typical values for voltage and current supplied by an alkaline battery are 1.5V and 700mA. This type of battery is designed to have a long life.

Even though alkaline batteries would seem like a great choice for our battery supply, the fact that they are not rechargeable is a huge upset. Further, we will explore some secondary batteries and analyze some of their advantages and limitations. We will discuss their characteristics and decide on which battery technology is best suited for our project.

Secondary batteries:

Secondary batteries cannot compete with primary batteries when it comes to specific energy or storage time levels, usability, readiness, and many other characteristics. However, this type of batteries has a very important advantage over primary batteries and it is that they can be recharged. Many chemistries exist within the secondary battery type and each different chemistry offers a different set of characteristics. Further, several chemistry technologies for secondary batteries will be explored. We will take a look at their characteristics and how a particular type of chemistry would be useful for our experiment. Finally, a chemistry technology will be chosen based on characteristics such as specific energy, charge time, self-discharge, and nominal cell voltage just to mention a few.

Lead Acid battery: A lead acid battery is the oldest rechargeable battery system there is. This type of battery is dependable and inexpensive. Compared to other types of batteries, the lead acid battery delivers much more power at low cost, making it cost effective. Lead acid batteries are heavier and less durable than other batteries with different chemistry technologies. If they are fully discharged, the battery may be damaged and every time the battery is charged/discharged the battery capacity decreases by a small amount. Lead acid batteries are not quickly charged, taking approximately between 14 to 16 hours to have a full charge. One peculiar thing about batteries with this chemistry technology is that they must be stored with a full charge whenever they are not in use. This has to be done in order to prevent sulfating, which is a condition that robs the battery of performance. A typical nominal cell voltage value for a lead acid battery is 2V. This type of battery may not be the best option given that is it heavy and one of our goals is to build a small, portable, and as buoyant as possible.

Table 4.2.2.a Advantages and limitations of lead acid battery

Advantages	- Not expensive compared to how much they deliver - Low self-discharge, 5% - High specific power - High current supply - Perform well at low and high temperatures - No memory loss - Can tolerate overcharge - Very low internal resistance
Disadvantages	- Low specific energy, 30-50 Wh/kg - Slow charge, usually between 14 to 16 hours for full charge - Must be stored with a full charge to prevent sulfating - Not environmentally friendly, toxic - They are heavy - Low cycle life count

Nickel-Cadmium (NiCd) battery: A NiCd battery is another type of rechargeable battery. Batteries with this chemistry technology come in a wide variety of sizes and capacities. This type of battery offers good cycle life and good performance at low temperatures. One of the best things about batteries with this chemistry technology is their ability to deliver its almost all its capacity at a very high discharge rate. This battery type has a specific energy of 45-80 Wh/kg and a typical nominal cell voltage value for batteries with this chemistry technology is 1.2V. They can be charged relatively fast with an approximate of 1 to 2 hours. However, one disadvantage of this battery type is that it has a high self-discharge rate. Because of this, these batteries need to be recharged after storage even if they still had any charge left. However, NiCd batteries can be stored in a discharged state without suffering from any damage. Given this, a NiCd battery might not be the best option for our battery supply.

Table 4.2.2.b. Advantages and limitations of NiCd battery

Advantages	- Relatively inexpensive - Ultra-fast charging, 1-2 hours - Good performance at low temperatures - High current discharge - Available in a wide range of sizes and performance options - Can be charged at different rates - High cycle life count, 1000 - Low internal resistance
Disadvantages	- Low nominal cell voltage, 1.2V - Suffers from memory effect - Not environmentally friendly, toxic - High self-discharge, 20% - Requires a full discharge every 3 months when in use - Low specific energy compared to other technologies (Li-ion)

Nickel-Metal-Hydride (NiMH) Battery: A NiMH battery is another type of rechargeable battery. This battery chemistry technology came after the NiCd battery. The specific energy for NiMH batteries is between 60-120 Wh/kg compared to 45-80 Wh/kg for the NiCd. However, when it comes to charging this battery it can be more complicated than charging a NiCd battery. Similar to NiCd batteries, one disadvantage of this battery type is that it has a high self-discharge rate. Because of this, these batteries need to be recharged after storage even if they still had any charge left. The self-discharge for a NiMH battery is higher than a NiCd battery. Also, the NiMH batteries have a much lower cycle life (300-500) compared to NiCd batteries (1000) and they take double the time to be fully charged. Given the high-self discharge and low cycle life count, a NiMH battery is probably not the best choice for the battery supply.

Table 4.2.2.c. Advantages and limitations of NiMH battery

Advantages	- Higher specific energy than NiCd batteries - Suffers from memory effect but not as much as NiCd - Relatively inexpensive - Environmentally friendly - Available in a wide range of sizes - Wide temperature range - Low internal resistance
Limitations	- Low cycle life count, 300-500 - Low overcharge tolerance - Low nominal cell voltage, 1.2V - Low specific energy compared to other technologies (Li-ion) - High self-discharge, 30% - Required full discharge every 3 months when in use

Lithium-Ion (Li-ion) battery: A Li-ion battery is another type of rechargeable battery that is widely popular within the industry. Given that lithium is the lightest of all metals, Lithium-ion batteries are very lightweight and this can be very useful to for our project. One of the best things of Li-ion batteries is that they are maintenance free and this can be very useful for us. Li-ion batteries do not suffer from memory and do not need a full discharge every given time like NiCd or NiMH batteries do. A typical nominal cell voltage value is 3.6V, which is higher than previous battery chemistry technologies discussed. One of the disadvantages of Li-ion batteries is that it requires a protection circuit to prevent abuse. Also, they are more expensive than other battery types.

Table 4.2.2.d. Advantages and limitations of Li-ion battery

Advantages	- High specific energy - High cycle life count - High nominal cell voltage - Environmentally friendly - Low self-discharge, <5% - Maintenance free - Does not suffer from memory
Limitations	- Requires a protection circuit for safe operation - Low overcharge tolerance - Not as cheap as other battery types - Suffers from aging problems

Within the Li-ion family several different types combinations exist. Amongst the most common are lithium cobalt oxide, lithium manganese oxide, and lithium phosphate.

<u>Lithium Cobalt Oxide ($LiCoO_2$) battery</u>: Lithium-cobalt batteries are a type of Li-ion battery that have high specific energy, a high cycle life count, and a very low self-discharge. However, some of the disadvantages of this battery are its low overcharge tolerance, relatively short life span, and low thermal stability. Its lo thermal stability makes this type of batteries susceptible to thermal runaway whenever it is exposed to high temperatures or if it is overcharged. Also, cobalt is expensive and these batteries are more expensive than other battery chemistry technologies discussed before. A typical nominal cell voltage value for this type of battery is 3.6V. They are maintenance free and environmentally friendly but required a protection circuit to prevent abuse. Its performance can be reduced if exposed to extreme temperatures.

Table 4.2.2.e. Characteristics of $LiCoO_2$ battery (With permission from Battery University)

Voltages	3.60V nominal; typical operating range 3.0-4.2V/cell
Specific energy (capacity)	150-200Wh/kg. Specialty cells provide up to 240Wh/kg
Charge (C-rate)	0.7-1C, charges to 4.20V (most cells); 3h charge typical. Charge current above 1C shortens battery life
Discharge (C-rate)	1C; 2.50V cut off. Discharge current above 1C shortens battery life
Cycle life	500-1000, related to depth of discharge, load, temperature
Thermal runaway	150C (302F). Full charge promotes thermal runaway.

Lithium Manganese Oxide ($LiMn_2O_4$) battery: Lithium-manganese batteries are a type of Li-ion battery that have a relatively lower specific energy and internal resistance than lithium-cobalt batteries. They have the same cycle life count as lithium-cobalt and can be charged in half the time of the previously mentioned. These batteries have low overcharge tolerance, very low self-discharge and a typical nominal cell voltage value of 3.7V. Given that these batteries have low internal resistance, they can be charged and discharged very fast. A lithium-manganese battery usually has a capacity roughly one-third lower than that of a lithium-cobalt battery. However, lithium-manganese batteries are very abundant, not expensive, environmentally friendly, and are thermally stable. They are maintenance free and environmentally friendly but required a protection circuit to prevent abuse. Its performance can be reduced if exposed to extreme temperatures.

Table 4.2.2.f. Characteristics of $LiMn_2O_4$ battery (With permission from Battery University)

Voltages	3.70V (3.80V) nominal; typical operating range 3.0-4.2V/cell
Specific energy (capacity)	100-150Wh/kg
Charge (C-rate)	0.7-1C, 3C maximum, charges to 4.20V (most cells)
Discharge (C-rate)	1C; 10C possible with some cells, 30C pulse (5s), 2.50V cut-off
Cycle life	300-700, related to depth of discharge, load, temperature
Thermal runaway	250C (482F). Full charge promotes thermal runaway.

Lithium Iron Phosphate ($LiFePO_4$) battery: Lithium-phosphate batteries are another type of Li-ion battery. Out of the three Li-ion batteries discussed, this battery type has the lowest internal resistance. However, it also has the lowest specific energy. Its cycle life count is double compared to the other two Li-ion batteries and its charge time is half that of a lithium-cobalt battery. Lithium-phosphate batteries have higher tolerance to overcharge and higher self-discharge than the other two Li-ion batteries. Typical nominal cell voltage values range vary between 3.2 and 3.3V. They are maintenance free and environmentally friendly but required a protection circuit to prevent abuse. Its performance can be reduced if exposed to extreme temperatures.

Table 4.2.2.g. Characteristics of $LiFePO_4$ battery (With permission from Battery University)

Voltages	3.20V, 3.30V nominal; typical operating range 2.5-3.65V/cell
Specific energy (capacity)	90-120Wh/kg
Charge (C-rate)	1C typical, charges to 3.65V; 3h typical charge time
Discharge (C-rate)	1C; 25C on some cells, 40A pulse (4s), 2.50V cut-off (lower than 2V causes damage)
Cycle life	1000-2000, related to depth of discharge, load, temperature
Thermal runaway	270C (518F). Very safe battery even if fully charged

Table 4.2.2.h. Comparing All Battery Options (Reprinted with permission BatteryU)

Specifications	Lead Acid	NiCd	NiMH	Li-ion Cobalt	Li-ion Manganese	Li-ion Phosphate
Specific energy density (Wh/kg)	30–50	45–80	60–120	150–190	100–135	90–120
Internal resistance[1] (mΩ)	<100 12V pack	100–200 6V pack	200–300 6V pack	150–300 7.2V	25–75[2] per cell	25–50[2] per cell
Cycle life[4] (80% discharge)	200–300	1000[3]	300–500[3]	500–1,000	500–1,000	1,000–2,000
Fast-charge time	8–16h	1h typical	2–4h	2–4h	1h or less	1h or less
Overcharge tolerance	High	Moderate	Low	Low. Cannot tolerate trickle charge		
Self-discharge/ month (room temp)	5%	20%[5]	30%[5]	<10%[6]		
Cell voltage (nominal)	2V	1.2V[7]	1.2V[7]	3.6V[8]	3.8V[8]	3.3V
Charge cutoff voltage (V/cell)	2.40 Float 2.25	Full charge detection by voltage signature		4.20		3.60
Discharge cutoff voltage (V/cell, 1C)	1.75	1.00		2.50 – 3.00		2.80
Peak load current Best result	5C[9] 0.2C	20C 1C	5C 0.5C	>3C <1C	>30C <10C	>30C <10C
Charge temperature	–20 to 50°C	0 to 45°C		0 to 45°C[10]		
Discharge temperature	–20 to 50°C	–20 to 65°C		–20 to 60°C		
Maintenance requirement	3–6 months[11] (topping chg.)	30–60 days (discharge)	60–90 days (discharge)	Not required		
Safety requirements	Thermally stable	Thermally stable, fuse protection common		Protection circuit mandatory[12]		
In use since	Late 1800s	1950	1990	1991	1996	1999

Lithium Polymer (Li-Po) battery: A lithium-polymer battery is a secondary battery with a relatively new chemistry technology. The difference between this battery chemistry technology and others is in the type of electrolyte used. Lithium-polymer batteries are very similar to lithium-ion batteries. For example, lithium-polymer batteries have a higher gravimetric energy density than lithium-ion batteries. Also, Li-Po batteries have a better form-factor. Lithium-ion batteries generally come in cylindrical or rectangular shapes. Lithium-polymer batteries, on the other hand, usually come in much thinner shapes. Both lithium-based batteries require extra care for safe usage. However, lithium-polymer batteries have been found to be more tolerant than lithium-ion batteries in terms of overcharge tolerance. This reduces the chances of operating the battery in an unsafe manner. Lithium-ion batteries are slightly cheaper than lithium-polymer batteries though. Further, lithium-polymer batteries have a typical nominal voltage value of 3.6 volts and the voltage may in fact reach as high as 4.2 volts. This characteristic is important to take into account when designing circuits.

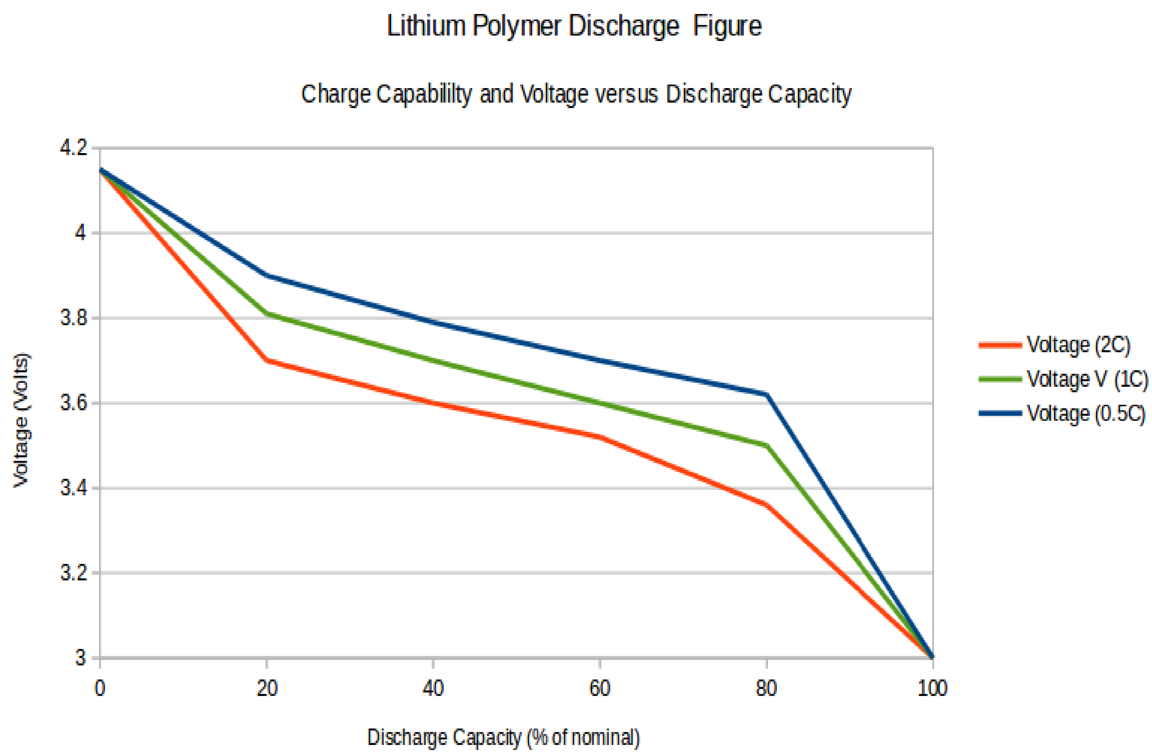


Figure 4.2.2.a. Nominal voltage versus discharge capacity of Li-Po battery (With permission from Battery University)

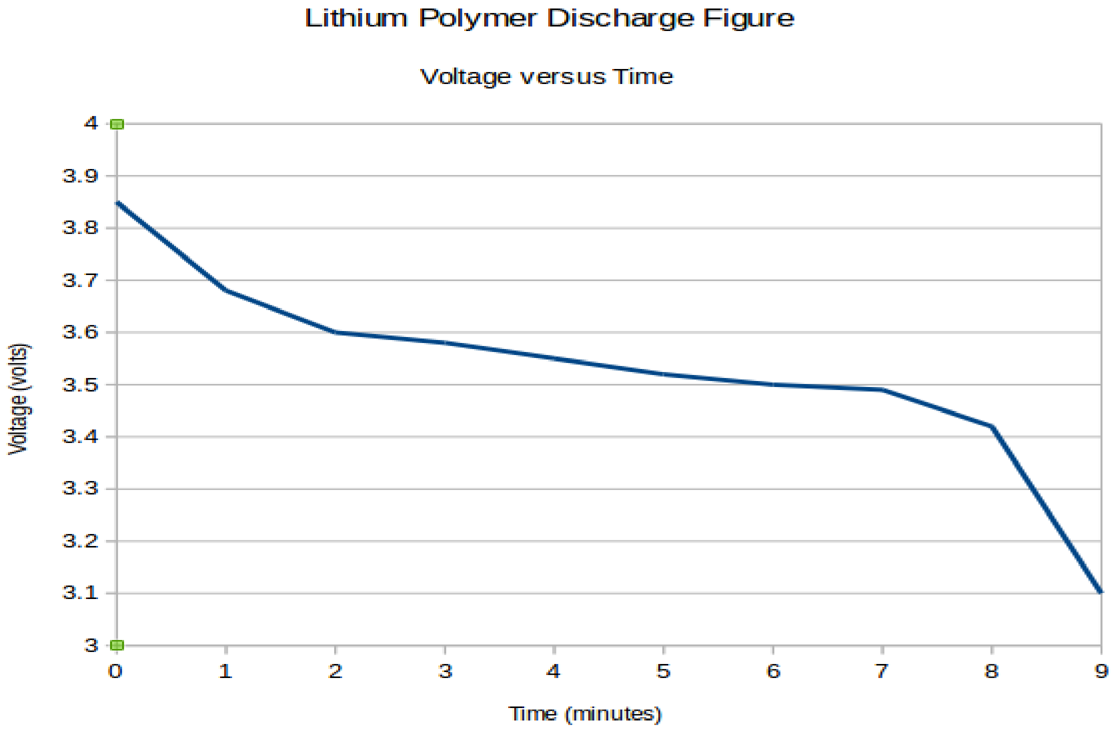


Figure 4.2.2.b. Nominal voltage versus time of Li-Po battery (With permission from Battery University)

Lithium polymer batteries have the highest nominal cell voltage compared to other battery chemistries. These batteries come in small packets and are very lightweight, which is a plus because we do not want our data buoy to be heavy. Further, lithium polymer batteries have a high cycle life and an acceptable self-discharge. More importantly, batteries with this battery chemistry do not require maintenance.

4.2.3 Battery Charging

Now that battery chemistry technologies have been discussed, we need to figure out how to charge them. Battery capacity cannot live forever, so it needs to be charged, and how the battery is charged depends on the chemistry technology of the battery. Because different battery chemistry technologies rely on different chemical reactions, each battery type has its own characteristics and different methods for charging them are applied. Temperature can also play a huge a rule when deciding which charger is better for which battery. For example, lithium-ion batteries cannot be charged when they are cold. Also, lead- and nickel-based batteries can be charged when they are cold but the charging must be done at a lower rate. Another aspect that needs to be taken into consideration is the rate of charge. Some battery types prefer to be charged at a lower rate but some other at a higher rate. Therefore, the chosen battery charger has to satisfy the specification of the chosen battery for our battery supply. After some research, some types of chargers are picked up to be discussed so we can make a final decision which one is best for our design.

Simple Charging: These chargers are also known as constant voltage chargers. They are very cheap and very simple and use a very simple technique for charging. A step down transformer takes an AC voltage from a power outlet and rectifies it to supply a desired DC voltage to charge the battery. Lithium-ion batteries are amongst the battery chemistry technologies that benefit from this type of chargers. An addition safety circuitry is required with these lithium-ion batteries for safety precautions. However, one of the disadvantages is that they take long time to charge, and they are even called overnight chargers, so they got very hot which in fact affect the battery longevity, yet lithium-ion batteries have no problem with that because of the protection circuit.

USB-based Charger: A USB-based charger is often used for a cell phone size battery. Today, most of the vehicles have USB outlet for different services including charging electronics devices such as cell phones. This type of charger is a slow rate charger comparing to using other chargers that are directly connected to an AC outlet. USB-based chargers are environmentally friendly and economically good. They do not contain any complex circuitry, and they can replace number of other individual chargers.

Trickle charger: This type is good for the self-discharge batteries but not for any lithium batteries such as NiMH because of their chemistries. It uses a constant current charging, and the charge rates vary from battery to another depending on their characteristics such as their frequency of discharging. Trickle chargers are designed such that in case the battery is completely charged, they will switch to trickle charging.

Solar Charger: This is another environmentally friendly charger. It is easy to implement it with our project, and the best way to do it is by having a solar cell implemented to the buoy housing. With that attached, the battery will charge while the buoy is exposed to sun. One of the disadvantages of it is it takes a long period of time to charge and also solar chargers are often used for a trickle charging. Another disadvantage is it can only be used in a sunny nice day. One of the requirements for our project is that is has to be solar powered. Given this, we will be looking into this method of charging and find the best way to implement it into our project.

4.2.4 Charging Methods

Next step to think about is the methods of using the chosen charger to charge the chosen battery. There are several techniques or methods to charge a battery with of course taking in account the characteristics of both the charger and the battery. Some of them will be discussed below to gain a better understanding on how to match different chargers with specific battery.

Constant Voltage: This method can be described just like the simple charger. It is simply a step down transformer with a rectifier that transfers and rectifies a DC power supply to a desired small voltage for our device. Having said that means the perfect charger that can go with this method is obviously the simple charger. Lithium-ion cell batteries are suitable to be used with this method, but without forgetting the additional circuitry for protection and safety.

Constant Current: It controls the current flow on making it constant by varying the voltage applied from the source. When the voltage of the battery is fully charged, the charger turns off. This method can be used for NiCd and the NiMH batteries.

4.3 Solar Panel

Solar Panels in general are simply PN Junctions in which light is shined on the junctions causing a flow of electrons that creates a voltage. The amount of voltage is directly related to the light intensity and unlike solar heating for pools has nothing to do with the heat created by the light on the panels. The inside of a solar panel is a photovoltaic cell that absorbs the light which causes an electron to split from the material in the cell which in turn generates a current in the semiconductor device. The solar panels in our system will be put into an electric circuit that will allow the batteries to charge up when enough light is available as well as run the device without draining the batteries in direct sunlight.

4.3.1 Various Options

The options for solar panels are quite varied for this project but all have to meet certain requirements. The size has to be small enough to fit onto the buoy without causing it to tip over or sink. It has to meet a minimum voltage level in order to be able to run all the components inside the device. This voltage is estimated at 10 volts. The Solar Panel should also be weather resistant and water proof.

4.3.2 General Descriptions

The Solar panels chosen all were found using the criteria of waterproof solar panels in a given price range. As such each of the panels are relatively small and should fit onto the buoy body. The Mighty max panel is the largest in terms of thickness but has the lowest voltage and current. The Seachoice panel is a longer panel with higher voltage and current than the Mighty Max panel. The Daily Extreme panel has the highest voltage and current, as well as being the best dimensions of the panels to fit the buoy body.

4.3.3 Comparing Solar Panels

Table 4.3.3. Comparing Solar Panel Options

	Mighty Max Solar Panel	Seachoice	Daily Extreme	Adafruit Solar Panel
Dimensions (in)	6.10 x 5.43	4 x 15 x 0.63	12.7 x 5 x 0.2	4.4 x 5.4 x 0.18
Voltage (V)	12	15	18	6
Current (mA)	60	100	150	330
Watts (W)	1		4.5	2
Weight (oz)	–	–	11.4	3
Waterproof	Yes	Yes	Yes	Yes
Cost	$22.99	$19.95	$24.99	$29

The Mighty Max panel does seem to have a significant thickness and weight to it that is not listed on the page but can be seen from the images of the panel. Taking into account the size and weight, as well as the lower voltage this panel was decided against by the group.

The Seachoice and Daily Extreme panels are both small enough, waterproof and fit the specifications listed when choosing a solar panel. However, for just slightly more cost the Daily Extreme panel offers a good improvement in voltage in the case that the components require more voltage than anticipated. If we need more voltage later, we can rearrange the circuits to allow more to get to the components but we cannot simply change the solar panel after the parts have been chosen.

Size-wise, the Adafruit Solar Panel seems to be one of the smallest out of our four options. It provides a voltage of up to 6V, which is enough for charging our chosen battery. It is significantly less heavy than other solar panels, it is also waterproof, and its price is very similar to the other three options.

4.3.4 Charging Schematic

Charging our lithium polymer battery can be a tricky task. We have to be careful as to not overcharge our battery because these types of batteries do not have a high overcharge tolerance. Also, we must watch out for the charging rate and especially the charging temperature of the battery.

The charging of the lithium polymer battery will be done via an Adafruit solar lithium polymer charger. This component will be very useful for this process because of its simplicity. This component charges any 3.7V/4.2V lithium polymer battery by using a 6V solar panel by simply plugging the battery into the BATT port using a 2-pin JST cable and the solar panel into the DC jack using a 2.1mm adapter cable. This charger is designed specifically for solar charging and it draws the maximum current possible out of the panel whenever exposed to light. Max charge rate for this device is set at 500mA but It can be easily adjusted to be as low as 50mA and as high as 1A by simply soldering in a resistor. Additional features for this device are that it automatically uses input power whenever available and the temperature of the battery can be monitored by simply soldering in a 10K NTC thermistor. Both features are useful for different reasons. The first one keeps the battery from constantly being charged/discharged. This extends cycle life of the battery and boosts battery performance. The second one is very important because batteries will not perform adequately if extreme temperatures are reached. By monitoring the battery temperature, we avoid damaging the battery and we would know when to charging of the battery is optimal. The solar lithium polymer charger is equipped with three color indicator LEDs. One LED indicates that the power connection has been established successfully, another LED indicates that the battery is currently being charged, and the third LED indicates when the battery has reached its maximum capacity.

Table 4.3.4 provides some of the key characteristics of this solar lithium polymer charger. We can see that it is small in size, which is important because it has to be able to fit inside the buoy housing, it can operate over a wide temperature range, which is also very important due to the fact that the interior of the buoy housing might get very hot after being exposed to the sun for several hours. The supply voltage range is not very wide but it can be altered if we are required to provide more or less voltage to any of our components. The supply current is very low whenever the device is not charging and during the charging phase, 2500μA should be sufficient to get our battery charged adequately.

Table 4.3.4. Solar Lithium Polymer Charger Characteristics

Solar Lithium Polymer Charger	
Dimensions (mm)	41 x 33 x 2
Supply Voltage Range (V)	4.4 – 6
Supply Current (μA)	2500 (charging)
	260 (charge complete)
	180 (standby)
	50 (shutdown)
Operating Temperature Range (°C)	-40 to 85

The figures below illustrate typical performance curves for this solar lithium polymer charger. Figure X shows how charge current is affected as the supply voltage increases. As stated in the image, these values are obtained at room temperature while using a 10kΩ programming resistor. It is clear that charge current maintains a steady value of approximately 101mA as the supply voltage increases. It might be the case where we need to regulate our voltage in order to satisfy power requirements for our other components. Figure X shows how battery regulation voltage is affected as the supply voltage increases. Four different curves are plotted in which different charge currents are present. All these values have been obtained at room temperature. We can see that regulated voltage slightly decreases as the supply voltage increases. However, this difference should not be significant enough as to affect our project.

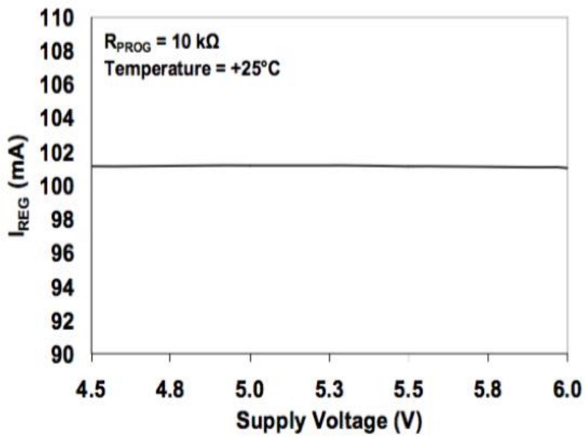


Figure 4.3.4.a: Supply voltage vs Charge current

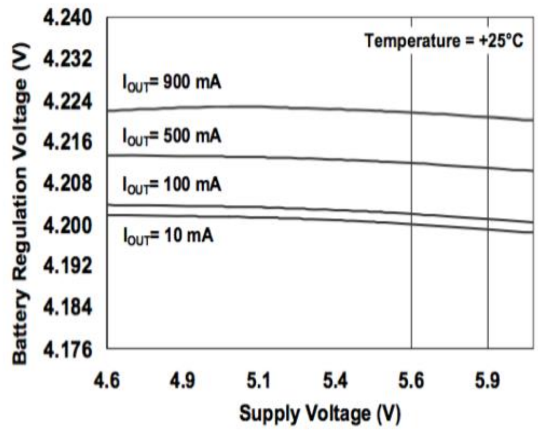


Figure 4.3.4.b: Supply voltage vs Regulation Voltage

Figure 4.3.4.c shows how three components –solar panel, solar lithium polymer charger, and lithium polymer battery– are connected together in order to charge a lithium polymer battery.

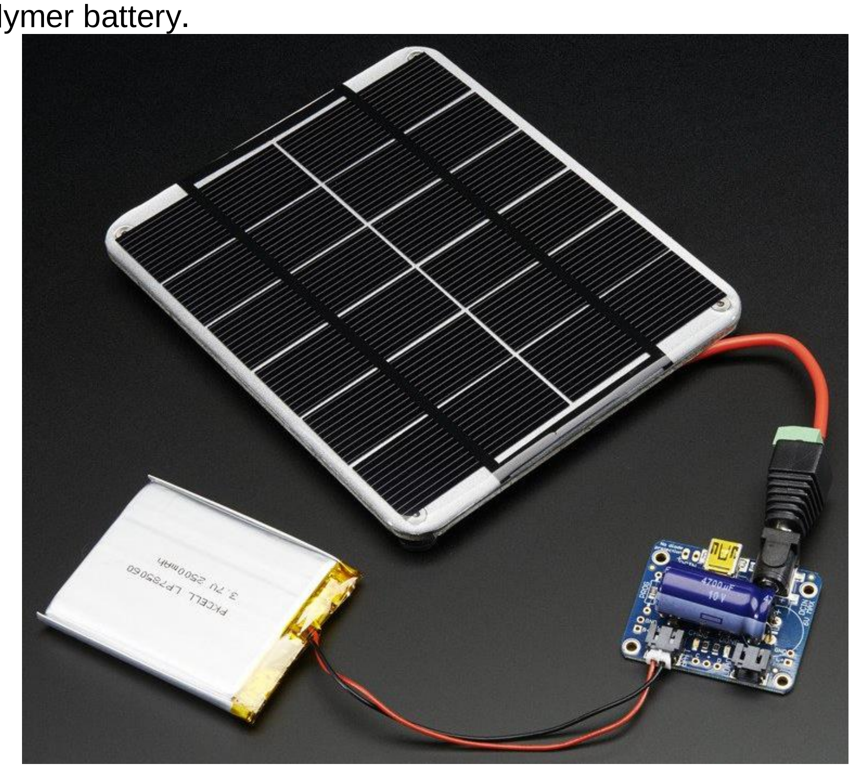


Figure 4.3.4.c: Solar panel + solar lithium polymer charger board + lithium polymer battery

4.4 Water Salinity Detection

Finding a sensor that can detect salinity in water is no easy task but by incorporating photonics we have a few options on how to proceed. When using any kind of photonics technology cost becomes an issue as there is a direct correlation between cost and sensitivity. In addition, size of the device can become an issue as some of the detection systems can be quite large.

4.4.1 Spectrometers

The first idea for a salinity sensor is by using spectroscopy to detect salt in the water but there are both pros and cons. This method offers the highest sensitivity and variety of measurement since it is not limited to just detecting salinity. However, the cost becomes a large issue with building a spectrometer into any device since they can cost hundreds if not thousands of dollars depending on the sensitivity. In addition to cost the size is an issue as the actual spectrometer is as large as the whole device is intended to be and it would be too heavy to place in a floating buoy. The device could simply take data on the spectrum and send it to a spectrometer on the receiving end but this would not significantly reduce cost and there does not seem to be much of a market for such a device currently.

Spectrometry is used to measure the spectrum of the sample versus the spectrum of pure water. This system would be ideal since materials have unique spectrums and it would be possible to measure many different contaminants and molecules in the water, however there are quite a few drawbacks to simply installing a spectrometer for this device. The first drawback is simply cost. Spectrometers are far out of the price range for the project as the goal is a relatively inexpensive device to measure salinity. The second drawback is size since spectrometers tend to be large in size for higher sensitivity and while you could attach a simple probe in the buoy to a spectrometer it would have to be attached for strong data. One final drawback would be complexity of the system as a spectrometer does have so many parts that have to be adjusted and aligned before it works properly and if something were knocked out of alignment it would be difficult to correct, whereas our project emits in the visible spectrum and it is simple to see if the beam is not going where it is intended to.

4.4.2 Fiber Optics

The next idea revolves around measuring salinity using the change in refractive index in the sample as compared to a completely distilled water sample. This method uses a laser diode that is coupled into a fiber optic cable, which in turn carries the light into a distilled water sample, passing through a clear divider into the sample and back into a fiber array. This fiber array would show where the beam has moved to on a CCD camera which would allow detection of salinity when comparing to the position of the beam when the sample has no salt. This method has the advantage of being a cheaper solution and should be sensitive enough to detect enough change in salinity to be useful for application. The biggest disadvantage is in sensitivity since there are many different solvents in water that could cause a change in refractive index and what is detected in the ocean could be more than salt content. However, there could be an option to create a modified version of this device using a more advanced sensor to obtain more than basic salinity information once the basic design is completed.

The laser diode used in this system will have a certain emissions wavelength in the visible spectrum which will be sent through a multimode fiber to a closed compartment underneath the buoy that will allow water to flow through to be tested. The chosen wavelength for this sensor is approximately 650 nanometers which is in the red light region of the visible spectrum. This wavelength is not an extreme danger to the human eye but should still not be looked directly into as it can cause harm to the retina with enough exposure. The light will emit through distilled water first, pass through a clear wall into the sample, reflect through a prism and back through the sample before entering an array of fibers which will carry the light back to a CCD camera that the array is focused onto. The CCD camera will detect the location of the spots for the given sample and by creating a database from no salt content to high salinity content we will be able to determine the salinity of the sample. This sensor works off of refractive index change of water with differing levels of salinity, which in turn moves the spots focused on the CCD camera. Depending on the set of spots received it is possible to determine which level of salt is in the sample. This should provide a sufficient level of accuracy for the sensor to easily distinguish set percentages of salt content independent of other variables.

The laser diode used in this system will emit in the visible spectrum, red light, to allow for easier alignment of the components in the system, reduced price of the diode compared to other wavelengths, and the necessity of choosing a wavelength at which the salinity will affect the refractive index of the material. All materials have different changes of refractive index for different wavelengths of light and choosing a wavelength that is more easily measurable is important. A multimode fiber was chosen to get as much of the light as possible through the system and into the sample due to the high absorption of water in general. Water absorbs energy at a high rate and as light is an electromagnetic wave it will have to travel through as little water as possible to still obtain a significant change in refractive index but without losing too much intensity and becoming unmeasurable. A single mode fiber for the output fiber from the sensor is used to obtain a smaller percentage of light sent through to the CCD camera for higher sensitivity as only a single mode will be allowed to travel through the fiber. Since that is the case then the shift in the mode of the fiber would lead to a shift on the position sensor. The position sensor measures how far the beam travels from its start location when the sample given is distilled water with no salinity. This is then used to calculate the salt content. A CCD camera with a pixel size smaller than the beam diameter and a large enough detection area would be used to obtain more sensitive movement data for the beam so smaller amounts of salinity differences can be detected by the system.

4.4.3 Reflected Beam Movement

A last option for measuring salt content of the sample is to use beam divergence and movement by angling a laser diode into the sensor. The beam would reflect off a mirror and back onto a quadrant cell sensor to detect beam motion with respect to refractive index change of the sample.

The beam movement through the system is dependent on the change in refractive index of the water sample. The more salt in the water the higher the refractive index of the sample as it depends on the relative permittivity and permeability of the sample. This changes the angle of transmission at the interface which changes the location of the spot on the quadrant sensor. Using Snell's Law:

$$N_1\mathbf{Sin}(\boldsymbol{\theta}_1) = N_2\mathbf{Sin}(\boldsymbol{\theta}_2)$$

Where N is the refractive index of the medium and θ is the angle of beam versus the y axis of the interaction relative to the surface. Therefore, to obtain the correct amount of movement with regards to the sensor we can adjust the angle, depth of the sample, and if needed even the spacing in between the input and output for the sensor.

This relates to the material properties and concentrations of molecules in the water, and by having the beam travel through distilled water as well we obtain the change in refractive index between material in the water instead of between water and air for a more accurate reading. This also reduces the reflection and lost light of the sample as it will be travelling from glass to water. The amount of light reflected is greater based on the difference in refractive index of the two materials. For example, glass has a refractive index of approximately 1.44 in a fiber, and water has an index of around 1.33. Air however has a refractive index around 1 which is much further from 1.44 than air is.

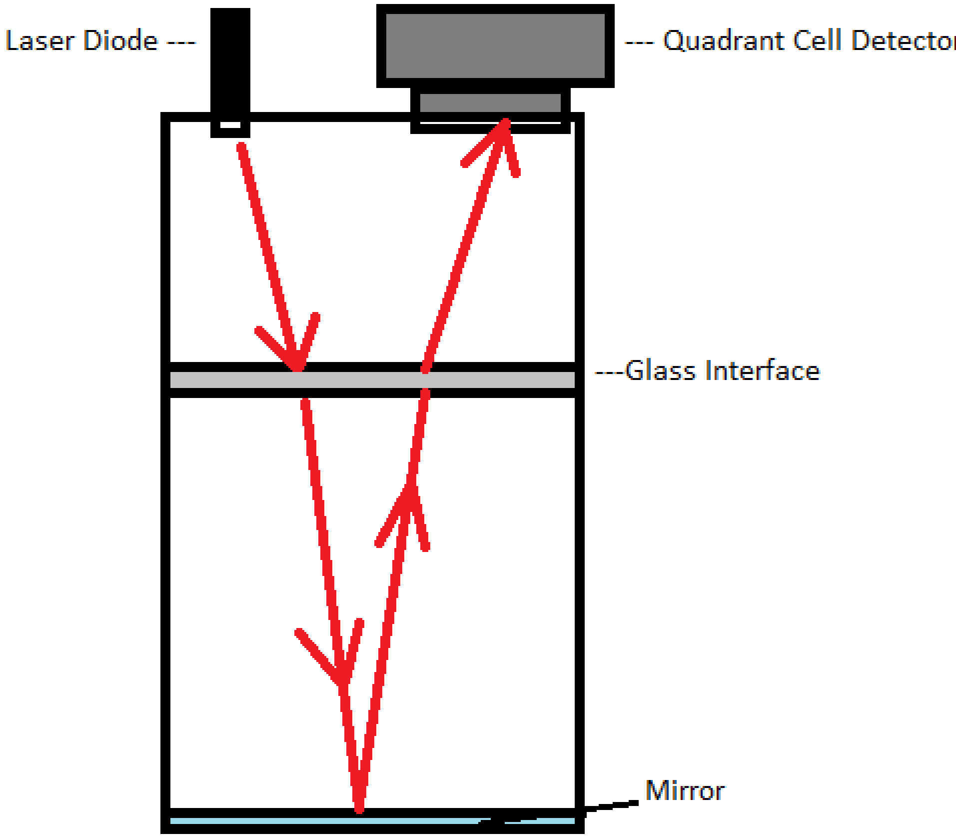


Figure 4.4.3. Salinity Detection System

4.4.4 Sensitivity

The sensitivity of the device is highly dependent on the sensor used but scales quite drastically with cost as discussed in previous sections. The first step towards determining sensitivity will be creating a scale that will need to be detected. A minimum sensitivity for the device will be set to detect a set percentage change in salinity of the water in order to minimize cost while still creating a useful and marketable device. The salinity will be detected in the amount of change in the spots detected on the CCD camera which through testing can be compared in salt percentage based on lab experiments.

4.4.5 Chosen Sensor

The chosen sensor for this device is the fiber optic device to measure change in refractive index. This will allow for more balancing of cost versus sensitivity and a larger fiber optic array can be added later on to allow for more sensitivity if it is needed. While the device will have to be bought and put together piece by piece it means we can afford more customization to our device and we can control cost more directly in this way. Some parts may have to be replaced in the event that the part is insufficient to obtain enough sensitivity, but it is anticipated that even with possible extra parts the cost can be kept well below the alternative price.

The laser diode used in this system will have a certain emissions wavelength in the visible spectrum which will be sent through a multimode fiber to a closed compartment underneath the buoy that will allow water to flow through to be tested. The chosen wavelength for this sensor is approximately 633 nanometers which is in the red light region of the visible spectrum. This wavelength is not an extreme danger to the human eye but should still not be looked directly into as it can cause harm to the retina with enough exposure. The light will emit through distilled water first, pass through a clear wall into the sample, reflect through a prism and back through the sample before entering an array of fibers which will carry the light back to a CCD camera that the array is focused onto. The CCD camera will detect the location of the spots for the given sample and by creating a database from no salt content to high salinity content we will be able to determine the salinity of the sample. This sensor works off of refractive index change of water with differing levels of salinity, which in turn moves the spots focused on the CCD camera. Depending on the set of spots received it is possible to determine which level of salt is in the sample. This should provide a sufficient level of accuracy for the sensor to easily distinguish set percentages of salt content independent of other variables.

The beam movement through the system is dependent on the change in refractive index of the water sample. The more salt in the water the higher the refractive index of the sample as it depends on the relative permittivity and permeability of the material:

$$n = \sqrt{\varepsilon_r \mu_r}$$

This relates to the material properties and concentrations of molecules in the water, and by having the beam travel through distilled water as well we obtain the change in refractive index between material in the water instead of between water and air for a more accurate reading. This also reduces the reflection and lost light of the sample as it will be travelling from glass to water. The amount of light reflected is greater based on the difference in refractive index of the two materials. For example, glass has a refractive index of approximately 1.44 in a fiber, and water has an index of around 1.33. Air however has a refractive index around 1 which is much further from 1.44 than air is.

$$R_0 = \left| \frac{n_1 - n_2}{n_1 + n_2} \right|^2.$$

The equation above shows the amount of reflection off a surface based on incident light, so for the smallest reflection the two refractive indices (n) must be as close as possible to each other.

The laser diode used in this system will emit in the visible spectrum, red light, to allow for easier alignment of the components in the system, reduced price of the diode compared to other wavelengths, and the necessity of choosing a wavelength at which the salinity will affect the refractive index of the material. All materials have different changes of refractive index for different wavelengths of light and choosing a wavelength that is more easily measurable is important. A multimode fiber was chosen to get as much of the light as possible through the system and into the sample due to the high absorption of water in general. Water absorbs energy at a high rate and as light is an electromagnetic wave it will have to travel through as little water as possible to still obtain a significant change in refractive index but without losing too much intensity and becoming unmeasurable. A single mode fiber for the output fiber from the sensor is used to obtain a smaller percentage of light sent through to the CCD camera for higher sensitivity as only a single mode will be allowed to travel through the fiber. Since that is the case then the shift in the mode of the fiber would lead to a shift on the position sensor. The position sensor measures how far the beam travels from its start location when the sample given is distilled water with no salinity. This is then used to calculate the salt content. A CCD camera with a pixel size smaller than the beam diameter and a large enough detection area is used to obtain more sensitive movement data for the beam so smaller amounts of salinity differences can be detected by the system.

Another option for measuring the salt content for the system would be to use spectrometry to measure the spectrum of the sample versus the spectrum of pure water. This system would be ideal since materials have unique spectrums and it would be possible to measure many different contaminants and molecules in the water, however there are quite a few drawbacks to simply installing a spectrometer for this device. The first drawback is simply cost. Spectrometers are far out of the price range for the project as the goal is a relatively inexpensive device to measure salinity. The second drawback is size since spectrometers tend to be large in size for higher sensitivity and while you could attach a simple probe in the buoy to a spectrometer it would have to be attached for strong data. One final drawback would be complexity of the system as a spectrometer does have so many parts that have to be adjusted and aligned before it works properly and if something were knocked out of alignment it would be difficult to correct, whereas our project emits in the visible spectrum and it is simple to see if the beam is not going where it is intended to.

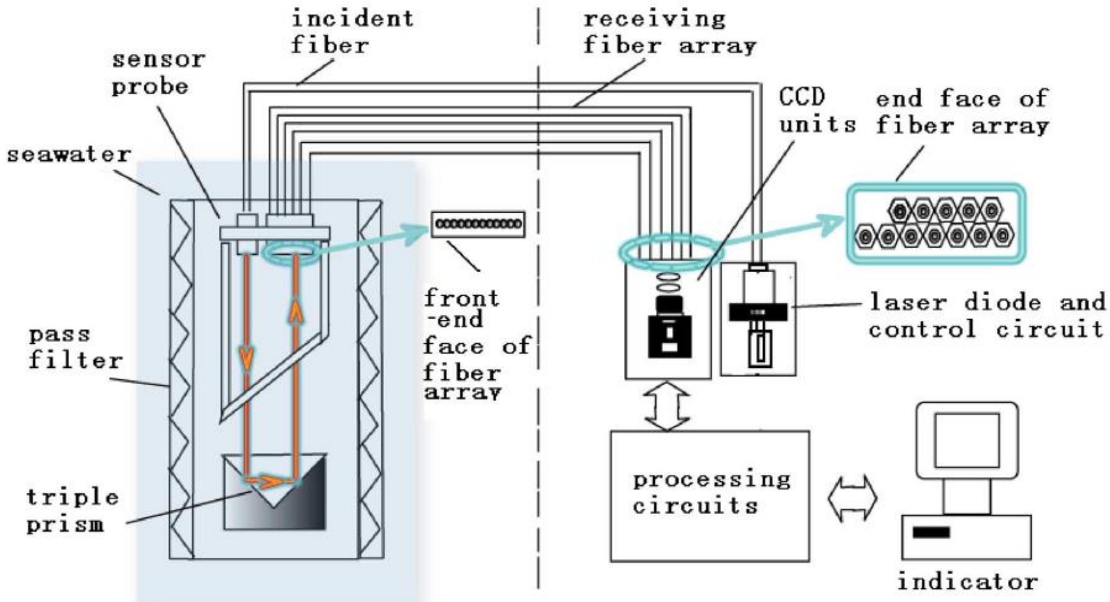


Figure 4.4.5. Optical Salinity Sensor System Based on Fiber-Optic Array, IEEE Sensors Journal 9(9):1148 - 1153 · September 2009

4.4.6 Sensor Design: Open vs Closed System

The sensor will have a closed design to block out as much ambient light as possible to allow for accurate detection in any light conditions. There will be holes to allow the water to enter the sensor but the ambient light should still be minimal enough to not interfere with the detector. While an open design underwater would allow for a more unobstructed flow of water to the sensing area the ambient light could interfere with readings. Considering how much absorption takes place in water it is important to keep as much of the input laser light as possible which means as small a sample as possible with the least outside interference as possible. Removing outside light means a lower signal to noise ratio on the detector which in turn gives us better readings. Water has a high absorption for light so it is important to make sure the path through the sample is not so significant that the output is unreadable. A contained system that blocks out ambient light means a lower signal to noise ratio for better detection.

4.4.7 Salinity Component Options

In the following sections, different components required for salinity detection will be discussed. More specifically, different options for laser diodes, mirrors, quad cells and sensor housings will be mentioned. Each different component will be studied and characteristics required for optimal operation will be mentioned. We will be looking out for these characteristics in each component in order to select the best suited for this part of our project.

4.4.7.1 Laser Diode Options

Table 4.4.7.1. Comparing Laser Diodes

	ThorLabs L650P007	Light in the Box	Aixiz Lasers
Wavelength (nm)	650	Red Light	635
Casing Included	No	Yes	Yes
Operating Voltage (V)	2.2 – 2.6	3.7 – 5	3.2
Power (mW)	7	5	10
Operating Current (mA)	28 – 35	--	--
Cost	$12.42	$1.00	$20.00

Figure 4.4.7.1.a: Laser Diode Option

The laser diode has several specifications to meet for the device. The Wavelength specified is around 635nm in the red spectrum of visible light. The operating voltage should be fairly low since all parts of the device will run off of the rechargeable battery and solar panel. The power of the laser diode must be sufficient to get light through to the output to obtain a decent signal for the device to obtain data. While a casing is not so much a requirement it will make putting the device together significantly simpler and is preferred. Finally, the cost must be kept fairly low so multiple parts can be purchased in case of breakage during testing.

The Thor Labs laser diode has the lowest operating voltage and seems to have a decent power output at 7mW. The laser diode does not come with any casing however which would increase difficulty integrating into the device and coupling into the fiber for the sensor. When factoring in the need to purchase the housing and such, the price becomes higher than desired for the device as well.

The Aixiz laser diode has the correct wavelength for the device at 635nm, and an operating voltage of 3.2V. In addition, the output power is higher than the last diode and it does come with a housing. The biggest drawback was cost for this option at $20.00 per laser diode for a project than can require 3 or more by the time the project is over the group went with the final option.

The light in the box laser diode was sold as a set of 5 diodes for $5.00 online. The specifications listed were slightly lacking so the diodes will go through testing to check wavelength, operating voltage and current, as well as output power are consistent for all five diodes upon arrival. This is the currently chosen laser diode for the device moving forward.

4.4.7.2 Mirror Options

Table 4.4.7.2.a. Comparing Mirror Options

	Edmund Optics	ThorLabs	Newport
Correct Wavelength	Yes	Yes	Yes
Dimensions (mm)	32 x 32	25.4 x 6	25.4 x 6
Cost	$25	$25.50	$108

The mirror will be placed at the bottom of the sample container to reflect the light back into the detector and measure the divergence. As such the mirror must be large enough to cover the bottom and allow the moving beam to reflect at a wide enough angle.

4.4.7.3 Quad-Cell Options

Table 4.4.7.3.a. Comparing Quad-Cell Options

	Mouser	Newport
Size (mm)	3.2 mm	
Peak Wavelength (nm)	633	1050
Pixel size (mm^2)	1.44	–
Responsivity (A/W)	0.4	0.5
Price	$44.54	$1,561

The selection of a Quadrant cell sensor for our device requires a large enough sensor size to be able to track the movement of the light at different salinity levels, but not so large as to not detect small enough changes. In addition, the operating voltage, and current, should be kept as low as possible in order to limit the power needed by the entire device but not at the expense of failing to obtain sufficient data for interpretation. As expected with this device, price becomes an issue when choosing a component and balancing sensitivity versus cost. It is easy for such sensors to become quite expensive and simply finding one in the project price range is difficult enough without worrying about specific power and size requirements. This did limit the choices available for the part however the parts listed seem capable of performing as needed.

When comparing products available the variation in price and usability became clear as the more expensive products give a considerable amount of information about the output but more than is necessary for our device. We will simply need to record the voltages of the output on the quadrants of the sensor to determine how far the beam has traveled across the surface which will allow us to determine the divergence.

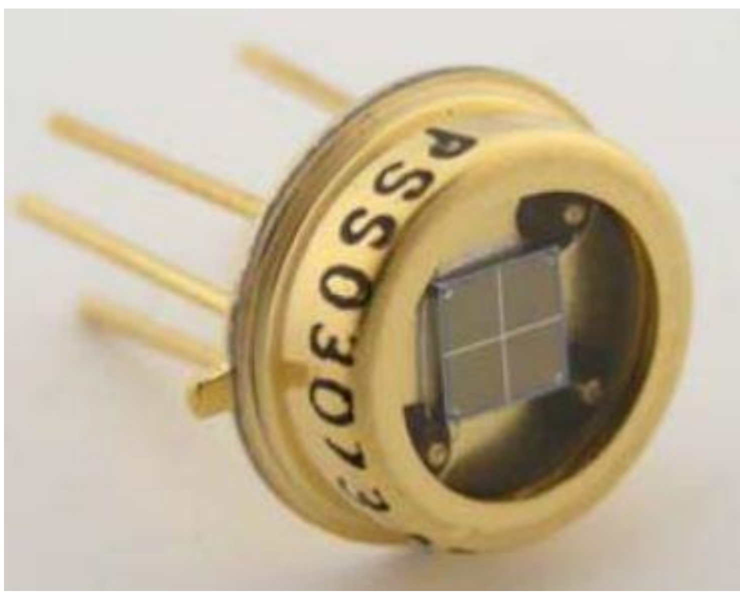


Figure 4.4.7.3.a. Quad-Cell Physical Appearance

4.4.7.4 Sensor Housing

The sensor housing has to be lightweight and able to be placed into the buoy body. In material should be rigid and not warp with exposure to the environment. The sides of the housing will have to be perforated to allow water to pass through to be tested, and the material should not break easily once perforations are made.

4.5 Wave Height Sensors

This section is focused on researching different methods and sensors that are currently being used to measure wave height. To begin doing the research we have looked at several existing data buoys that are similar to our project. Many of these buoys are used for commercial purposes and are very big and heavy compared to what we are trying to design. However, most data recording buoys utilize accelerometers to measure wave height. Whether they are measuring wave height only or movement in other directions as well, this electronic device is the most popular to get the job done. Given this, we have decided to utilize an accelerometer to be able to implement this measurement type in our buoy design.

In the following sections, a brief discussion of accelerometers will be made in which an overall description of what an accelerometer does is provided. Also, some of the most important features to take into consideration when selecting the appropriate accelerometer for our project will be mentioned. Further, we will take a look at some options we find suitable for our project and discuss their characteristics in order to be able to determine which of these options would be better to implement in our design. This discussion of different accelerometers currently found in the market will lead to a more detailed comparison and finally we will choose the best option for our project.

4.5.1 Accelerometers

Acceleration is defined as the rate of change of the velocity of an object and it is measured in meters per second squared (m/s^2). As its name suggests, accelerometers are devices that measure acceleration, which is expressed in G-forces (g). These measurements of acceleration can be either static or dynamic forces of acceleration. A static force of acceleration is the force caused by gravity and dynamic forces of acceleration include vibrations and movement. To better understand the concept of G-forces, a single G-force is equivalent to 9.8 m/s^2 here on planet Earth.

When selecting the appropriate accelerometer for our project, characteristics such as range, interface, number of axes measured, and power usage where taken into consideration. The range of a specific accelerometer refers to the upper and lower limits of what it can measure. Generally, the smaller the range, the greater the reading sensitivity will be for the accelerometer. Typical accelerometer ranges stretch from ±1g to ±250g. The interface for a given accelerometer is one on its most important specifications. Accelerometers can have one of three different interfaces: analog, digital, or pulse-width modulated (PWM). These three interfaces will be discussed in further sections. Another characteristic of an accelerometer is its number of axes. Accelerometers can be single-, double, or triple-axis, the number of axis indicates in how many directions it can measure acceleration. The most common type is the triple-axis accelerometer. For the purposes of measuring wave height, it is most likely that only a single-axis will be needed. However, all three will be taken into consideration given that prices do not vary much between these three and generally they are equally sensitive. Finally, given that our project is battery powered, we have to take into consideration how much power our accelerometer will consume.

4.5.2 Accelerometer Options

Many different accelerometer options are currently available in the market. The technology behind these devices improves every year making them more accurate by the year. Accelerometer options vary in range, size, current supply, voltage supply, price, number of axes, and many other characteristics. As mentioned before, the range of an accelerometer can vary between ±1g to ±250g. This characteristic of an accelerometer determines how precise or sensitive an accelerometer can be. For example, an accelerometer with sensitivity of ±1g is extremely precise and is typically used for measurements such as seismic readings or vibrations. On the other hand, an accelerometer with sensitivity of ±250g is typically used to measure the acceleration of rockets when launched into outer space.

The main feature to consider when selecting the correct accelerometer is the range. For the purpose of wave height, we do not need an accelerometer range that exceeds ±8g. Given that, we have selected to look into four different accelerometer options currently available in the market. The LIS3DH is a triple-axis accelerometer with selectable scaling range that varies between ±2g to ±16g. The ADXL335 is triple-axis accelerometer with three analog outputs and a range of ±3g. The ADXL345 is another triple-axis accelerometer with a selectable scaling range that varies between ±2g to ±16g like our first option. Finally, the MMA08451 is a triple-axis accelerometer with a selectable scaling range that varies between ±2g to ±8g. In the following section, we will take a closer look at the four accelerometer options. Several other characteristics will be considered and a more detailed description will be provided for each option.

4.5.3 General Descriptions

The LIS3DH is a very small, lightweight triple axis accelerometer with selectable full scaling of ±2g, ±4g, ±8g, or ±16g that can measure acceleration with output data rates within a range of 1Hz to 5kHz. This accelerometer comes with an embedded temperature sensor. This could be useful to our design because another requirement for our project is to measure air and water temperature. Given this, the LIS3DH is a good option if we need to save some space and money. Also, the accelerometer is to be embedded inside the buoy housing and exposed to the sun whenever it is at use. This accelerometer can operate on a very wide temperature range starting at -40°C to 85°C. This is very important because the inside of the buoy housing can get very hot depending on how the weather is outside on a particular day. Some of the key features for this accelerometer include:

- Triple-axis sensing
- Selectable scaling
- I2C and SPI digital output interface options
- Interrupt output
- Low-power mode consumption
- 16-bit data output
- 1.71-3.6V supply for operation

The ADXL335 is a very small, lightweight triple-axis accelerometer with signal conditioned voltage outputs. This accelerometer comes with three analog outputs, one for each axis, mounted on a small breakout board and it measures acceleration with a minimum full scale range of ±3g. This accelerometer can measure static acceleration due to the force of gravity as well as dynamic acceleration due to motion, shock, or vibration. Available bandwidths for this model range from 0.5 to 1600 Hz for the X and Y axes and from 0.5 to 550 Hz for the Z axis. Some key features for this accelerometer include:

- Triple-axis sensing
- Low power
- 1.8-3.6V supply for operation
- Adjustable bandwidth
- Excellent temperature stability
- Small dimensions
- Analog output

The ADXL345 is a small, lightweight, triple-axis accelerometer with digital I2C and SPI interface breakout. This accelerometer comes with high resolution measurement that can be set to ±2g, ±4g, ±8g, or ±16g for range. Similar to the LIS3DH, this accelerometer can operate on a very wide temperature range starting at -40°C to 85°C. This is very important because the inside of the buoy housing can get very hot depending on how the weather is outside on a particular day. Some key features for this accelerometer include:

- Triple-axis sensing
- Selectable scaling
- Wide temperature range
- User selectable resolution of up to 13-bit
- I2C and SPI digital output interface
- 2-3.6V supply for operation
- Adjustable bandwidth
- Low power

The MMA8451 is a very small, lightweight triple-axis digital accelerometer that communicates over I2C. This accelerometer comes with a high resolution acceleration measurement that can be se to ±2g, ±4g, or ±8g. The MMA8451 can be configured to remain in a low-power mode whenever it is not being used. This can be useful because our buoy will not be monitoring wave height at all times.

Instead, it will be prompted to measure accelerations in the y-axis over a specified period of time and obtain an average displacement measurement from those recorded values. By entering into low-power mode it can save a significant amount of power consumption to our design. Some key features for this accelerometer include:

- Triple-axis sensing
- Selectable scaling
- 1.95-3.6V supply voltage
- 14-bit and 8-bit digital output
- I2C digital output interface
- Low power

4.5.4 Comparing Accelerometers

Now that our four selected accelerometer options have been introduced, it is time to discuss their different characteristics and how these can be utilized to benefit our project design. Table 4.5.4.a provides a detailed summary of the most important characteristics that were taken into consideration when selecting between the four accelerometers.

Table 4.5.4.a Summary of characteristics for selected accelerometer options

	LIS3DH	ADXL335	ADXL345	MMA8451
# of Axes	3	3	3	3
Scaling Range (g)	±2 to ±16	±3	±2 to ±16	±2 to ±8
Output Type	I2C, SPI	Analog	I2C, SPI	I2C
Output Resolution	10-bit	–	10-bit to 13-bit	14-bits
Output Data Rate (Hz)	1 – 5000	–	0.1 – 3200	1.56 – 800
Dimensions (mm)	20.62 x 20.32 x 2.6	19 x 19 x 3.14	25 x 19 x 3.14	21 x 18 x 2
Weight (g)	1.5	1.27	1.27	1.3
Supply Current (A)	–	350µ	30µ – 140µ	6µ – 165µ
Operating Voltage Range (V)	1.71 – 3.6	1.8 – 3.6	2 – 3.6	1.95 – 3.6
Operating Temperature Range (°C)	-40 to 85	-40 to 85	-40 to 85	-40 to 85
Price	$6.95	$14.95	$17.50	$7.95

The first characteristic in the table is the number of axes that each accelerometer has. Even though some single- and double-axis accelerometers where found on the market, triple-axis accelerometers are much more popular, more advanced than single- or double-axis accelerometers and can get the job done as well. All four accelerometer options have 3 axes: x, y, and z. For the purposes of our project, we will only be needing one axis to measure upward/downward acceleration. Next, the scaling range. This characteristic does vary depending on which accelerometer we look at. The LIS3DH and ADXL accelerometers come with a selectable scaling range of ±2g, ±4g, ±8g, or ±16g. Similarly, the MMA8451 also comes with a selectable scaling range but does not go up to ±16g, it stays at ±8g. The ADXL335 does not come with a selectable scaling range. Its range is strictly ±3g. However, this does not mean that this accelerometer has to be discarded. While having a selectable scaling range is useful in case we are required to add more or less sensitivity, the wave height measurement we intend to obtain should require a range somewhere between ±2g and ±4g.

The next characteristic observed is the output type. Two output types are available for our chosen accelerometers, digital and analog. Three of the four accelerometer options have a digital output and the remaining option is analog. Further, two communication interfaces can be utilized within the digital output category, serial peripheral interface (SPI) and inter IC control interface (I2C). The LIS3DH and the ADXL345 come equipped so that both communication interfaces can be used. The MMA8451 is only available with I2C communication interface. The ADXL335 is the only one out the four accelerometer options that has an analog output type. All four accelerometers have similar dimensions. They are very small and lightweight.

Operating voltage range for all four accelerometers is also very similar. While the maximum operating voltage for all of them is 3.6V, the minimum operating voltage does vary from accelerometer to accelerometer. The LIS3DH operates at a minimum voltage of 1.71V, the ADXL335 operates at a minimum voltage of 1.8V, the ADXL345 operates at a minimum voltage of 2V and finally the MMA8451 operates at a minimum voltage of 1.95V. The operating temperature range is the same for all four accelerometer options, -40 to 85 °C. Finally, one of the most important, the price. As seen in figure X, prices vary for each different accelerometer. The LIS3DH and the ADXL345 have very similar characteristics. However, when it comes to price there is a significant different. The ADXL345 is approximately ten dollars more expensive than the LIS3DH. Also, the MMA8451 has similar characteristics to both LIS3DH and ADXL345. The only significant difference is that it only offers one communication interface while the other two accelerometers offer both communication interfaces. When it comes to price, the MMA8451 is only one dollar more expensive than the LIS3DH. This is not a significant difference to take into consideration.

After discussing all previously mentioned accelerometers, we have decided to narrow it down to two accelerometers. These are the LIS3DH and the ADXL335.

The former has a digital output type and the latter has an analog output type. Both will be tested in order to select the most appropriate for our design.

The actual LIS3DH sensor is the square little chip located right in the middle of the board in figure X. This is what does all the motion sensing that we need to measure the wave height. All other components and pins are used for the overall functionality of this accelerometer. There is power, I2C, SPI, and other pins that need to be connected to the microcontroller in order to get an interpretable measurement. By looking at the accelerometer board we can see the directions for the positive x-, y-, and z-axis. The x-axis runs from left to right of the accelerometer board, the y-axis runs from top to bottom of the accelerometer board, and the z-axis runs right through the accelerometer board making a perpendicular angle with the surface of the accelerometer board.

There are three power pins: Vin, 3Vo, and GND. Given that the actual sensor on the board requires an input voltage between 1.71 and 3.6V power to function, the board comes equipped with a 3.3V regulator in case a 5V microcontroller like Arduino is used.

- Vin: This pin is the power pin and is located at the bottom left corner of the accelerometer board. The sensor is powered via this pin and since it requires a maximum of 3.6V, the board has been equipped with a 3.3V regulator. To power the board, this pin is connected to the microcontroller.
- 3Vo: This pin is the regulated 3.3V output from the onboard voltage regulator. It is located at the bottom left of the accelerometer board right after the Vin pin.
- GND: This pin is the common ground for power and logic. It is located right after the Vin and 3Vo pins. This pin has to be connected to the GND pin on the microcontroller board.

There are four I2C pins: SCL, SDA, CS, and SDO.

- SCL: This is the I2C clock pin and it needs to be connected to the microcontroller's I2C clock line. This pin is located at the bottom of the accelerometer board right after the GND pin. It has a 10kΩ pullup resistor already on it.
- SDA: This is the I2C data pin and it needs to be connected to the microcontroller's data line. This pin is located at the bottom of the accelerometer board right after the SCL pin. It also has a 10kΩ pullup resistor already on it.
- CS: If I2C communication interface is used, this pin hast to be kept either disconnected or tied to a high logic level. This pin is located at the bottom right corner right before the INT pin.
- SDO: This pin is used for address selection. If it is connected to ground, its address is 0x18. If connected to 3.3V, its address is 0x19. This pin is located at the bottom right corner of the accelerometer board.

The same four pins are used as SPI pins: SCL, SDA, CS, and SDO.

- SCL: This pin is used as the SPI clock pin. This pin is located at the bottom of the accelerometer board right after the GND pin. It is an input to the chip.
- SDA: The Serial Data In / Master Out Slave In pin is used for data sent from the processor to the LIS3DH sensor. This pin is located at the bottom of the accelerometer board right after the SCL pin.
- CS: This pin is used as the Chip Select pin. To begin an SPI transaction, drop it low. This pin is located at the bottom right corner right before the INT pin. It is an input to the chip.
- SDO: The Serial Data Out / Master in Slave Out pin is used for data sent from the LIS3DH sensor to the processor. This pin is a 3.3V logic level out and is located at the bottom right corner of the accelerometer board.

The last pin on the accelerometer board is the INT pin. This is the interrupt output pin and it is located at the bottom right corner of the accelerometer board. This pin can be configured to trigger for many different reasons which include motion, tilt, data ready, and other. It is a 3.3V logic output.

The actual ADXL335 sensor is the square little chip located right in the middle of the accelerometer board in figure 5.5.b. This is what does all the motion sensing that we need to measure the wave height. All other components and pins are used for the overall functionality of this accelerometer. By looking at the accelerometer board we can see the directions for the positive x-, y-, and z-axis. The x-axis runs from left to right of the accelerometer board, the y-axis runs from top to bottom of the accelerometer board, and the z-axis runs right through the accelerometer board making a perpendicular angle with the surface of the accelerometer board. This accelerometer board has seven different pins: Vin, 3Vo, GND, Zout, Yout, Xout, and Test. Each of these pins has a different function.

- Vin: This pin is the power pin and is located at the bottom left corner of the accelerometer board. The sensor is powered via this pin and since it requires a maximum of 3.6V, the board has been equipped with a 3.3V regulator. To power the board, this pin is connected to the microcontroller.
- 3Vo: This pin is the regulated 3.3V output from the onboard voltage regulator. It is located at the bottom left of the accelerometer board right after the Vin pin. This pin can be used to obtain better accuracy and precision by simply connecting this pin to the analog reference pin on the microcontroller board.
- GND: This pin is the common ground for power and logic. It is located right after the Vin and 3Vo pins. This pin has to be connected to the GND pin on the microcontroller board.
- Zout: This pin is the analog output for the z-axis. It outputs the data recorded when the accelerometer senses movement in the z-axis. This pin has to be connected to an analog input pin on the microcontroller board. This pin is located at the bottom of the accelerometer board right after the GND pin.

- Yout: This pin is the analog output for the y-axis. It outputs the data recorded when the accelerometer senses movement in the y-axis. This pin has to be connected to an analog input pin on the microcontroller board. This pin is located at the bottom of the accelerometer board right after the Zout pin.

Xout: This pin is the analog output for the x-axis. It outputs the data recorded when the accelerometer senses movement in the x-axis. This pin has to be connected to an analog input pin on the microcontroller board. This pin is located at the bottom of the accelerometer board right after the Yout pin.

4.6 Temperature Sensors

To begin researching temperature sensors for the buoy system we decided to look into what type of sensors are on the market today and what they are typically used for. Being that we will need two temperature sensors, one for air temperature and one for water temperature, there are a few other specifications that we will need to consider when determining what components will be best suited for the system we are using. There are various capabilities for measuring temperature, which will be discussed in the following few pages. After further review, a decision on the right two temperature sensors, will be determined.

Temperature sensors are a major part of any water or air data systems being used today. Before we begin the research into the wide variety of electronic temperature sensors on the market, we will take a few key factors into consideration. For both the air and water temperature sensors, we will be researching each separately, though we may have the possibility of using the same component in two different locations on the buoy

4.6.1 Various Options

Being that there are many different options available on the market today, determining the best suited temperature sensor for measuring water and air can take a good amount of research. For each sensor, we will be analyzing the most imperative features of each component we look into. More specifically, input voltages, temperature ranges, precision and accuracy, as well as physical appearance and durability of the sensor itself. We have looked into a select few sensors that seem to be closest to the data buoy system's intended uses. The temperature sensor must be waterproof, rugged, and be small enough in size to fit into our buoy housing. As we researched, we found three that matched our criteria needed for the data buoy system.

As talked about before, we are looking into a few key features that each temperature sensor will need and should be able to handle. The first sensor we found is the DS18B20 digital thermometer. The next possible component we found is the TMP36 low voltage temperature sensor. These two components are similar though one uses a digital signal and the other uses analog, respectively. The last temperature component we researched is the 3950NTC. These are just three of the many temperature sensors out today. We chose to look further into these because they seem to match the description closest to what we are needing for the buoy system.

4.6.2 General Descriptions

The DS18B20 is a programmable resolution 1-wire digital thermometer that can provide 9-bit to 12-bit Celsius temperature measurements. ''The DS18B20 communicates over a 1-wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. In addition, the DS18B2 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply. Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-wire bus. Thus, it is simple to use one microprocessor to control many DS18B29s distributed over a large area." (Reference 5)

As for the TMP36, this sensor has the same TO-92 package as the DS18B20 but a few different features. "The TMP36 is a low voltage, precision temperature sensor. It provides a voltage output that is linearly proportional to the Celsius temperature scale. The TMP36 does not require any external calibration to provide a precise an accurate measurement inside the intended temperature ranges." (Reference 6) The TMP36 is a simple 1 wire configuration like the DS18B20 though it uses analog signals instead of digital.

The last temperature sensor we will research is the 3950NTC or better known as a 10K Precision Thermistor. A thermistor is a thermal resistor, or a resistor that changes is resistance due to temperature. Thermistors are made so that the resistance changes drastically, as much as 100 ohms per degree, so that the temperature can be easily measured. The NTC stands for Negative Temperature Coefficient, meaning that the higher the temperature, the lower the resistance. With lower resistance, more current is able to pass through, causing smooth flow, for better temperature measuring. The 10K precision is good for measuring temperatures between -55°C to 125°C or (-67°F to +247°F). The particular thermistor we are researching is called 10K is because the resistance at the ideal temperature of +25°C is 10Kohms of ±1% precision.

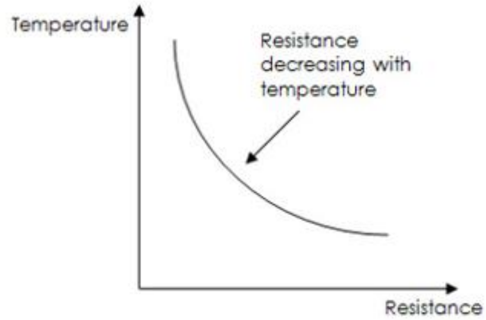


Figure 4.6.2a. Negative Temperature Coefficient (NTC) Thermistor Resistance Rate

4.6.3 Comparing Temperature Sensors

The DS18B20 is very commonly used for design projects and measuring outdoor and indoor temperatures. This sensor needs 3.0V to 5.5V supplied power to be used. This will be perfect for our system being that we will be supplying around 3.0V to 6.0V outputs from our microcontroller. The active current is around 1.0 to 1.5 mA. This sensor can measure temperatures from -55°C to +125°C or (-67°F to +257°F). The accuracy of the sensor is within ±0.5°C in the range of -10°C to +85°C. One of the main advantages of this component is that it can be found online, ready to be used in a waterproof shielding and probe. The sealed sensor lets you precisely measure temperatures in wet environments. This is a great advantage since it would be fairly difficult to implement our own way of waterproofing this sensor. For finding the conversion from digital bits to a degree in temperature we have followed the following table given.

Temperature (°C)	Digital Output (Binary)	Digital Output (Hexadecimal)
+ 125	0000 0111 1101 0000	07D0
+ 85	0000 0101 0101 0000	0550
+ 25.0625	0000 0001 1001 0001	0191
+ 10.125	0000 0000 1010 0010	00A2
+ 0.5	0000 0000 0000 1000	0008
0	0000 0000 0000 0000	0000
-0.5	1111 1111 1111 1000	FFF8
-10.125	1111 1111 0101 1110	FF5E
-25.0625	1111 1110 0110 1111	FE6F
-55	1111 1100 1001 0000	FC90

Figure 4.6.3a. DS18B20 Digital Bits to Temperature Conversion

One disadvantage of the DS18B20 is that compared to the other two sensors, it has the highest cost. The chip itself is pretty cheap, but with the waterproof design it can be costlier. Another downside to the DS18B20 is that the coding can seem to be a little complex and confusing. The chip needs two specific libraries, one for being a single wire signal device and one for the actual Dallas Temperature library used for the component itself. These are needed to perform the calculations needed to convert the voltage readings into hexadecimal and then to degrees Celsius.

When looking into the specifications of the TMP36, we can see that during low voltage operation it needs a supply of 2.7V to 5.5V. The active current is very low being around 0.05 mA. The specified temperatures that can be measured range from -40°C to +125°C or (40°F to +257°F). This accuracy of this sensor is within ±1°C while measuring inside the specified temperature range but can be as low as ±0.1° while at the ideal temperatures. One advantage to this particular sensor is that it needs simple and very little programming to be able to use the chip. The conversions from output voltage to temperature can be simply calculated using a few basic formulas:

When using a 3.3V input to the sensor, and connecting the signal to an analog pin on the microcontroller, the 10-bit analog reading can be converted into Celsius.

First the input voltage must be converted from volts to millivolts (when using 5V input):

$$\text{Voltage at pin in millivolts} = (\text{reading from ADC}) * (3300/1024)$$

This formula converts the number 0-1023 from the ADC into 0-3300mV
Now converting millivolts into temperature or degrees Celsius:

$$\text{Centigrade temperature} = [(\text{analog voltage in mV}) - 500] / 10$$

Comparing the previous two sensors to the thermistor is fairly one-sided being that they are totally different types of sensors, but in determining the best one for our buoy, we must take into account all of our options. One advantage to using the 10K precision thermistor is that, like the DS18B20, it can be bought with a special coating to help waterproof the thermistor and prevent any moisture from getting to the component itself. Another advantage is that it is very accurate for the cost. They can be found very cheap and as accurate as ±0.25°C with an accurate enough analog converter. One disadvantage is that they require a little more work than the compared sensors to interpret the input readings.

Comparing the three different sensors we find many advantages and disadvantages for each particular component. The key comparisons of specifications can be seen in Table 4.5.2a.

Table 4.6.3b. Comparing DS18B20, TMP36, and 10K Precision Thermistor Specifications

	DS18B20	TMP36	10K Precision Thermistor
Minimum Voltage	3.0-5.5V	2.7-5.5V	No Requirement
Waterproof	YES	NO	Water Resistant
Accuracy @ 25°C	±0.5°C	±0.1°C	±0.25°C*
Cost of Component**	$1.49	$1.50	$0.26
Cost w/ waterproofing**	$12.95	NA	$8.00

*This accuracy value is reached as long at the analog converter used is accurate enough.
**These costs are provided for component alone without the addition of shipping and handling or local taxes.

4.6.4 Pros and Cons

Looking back on all the advantages and disadvantages, as well as the specifications, we can put all the gathered information together and choose the component we find most suitable for our data buoy system.

DS18B20:

Pros:

- Wide range temperature measurability
- Simple 1-wire technology
- Digital signal output
- Resolution of the temperature conversion can be selected from 9 to 12 bits
- Low voltage supply needed
- Precise accuracy within ±0.5°C between -10°C and +85°C
- Can be purchased with waterproof coating
- Durable

Cons:

- Fairly complex programming code
- Costly due to waterproof coating

TMP36:

Pros:

- Wide range temperature measurability
- Simple 1-wire technology
- Analog signal output
- Low voltage supply needed
- Accuracy as low as ±0.1°C around room temperature
- Simple voltage to temperature conversions
- Inexpensive

Cons:

- Not sold waterproof
- Not Durable

10K Precision Thermistor:

Pros:

- Easy resistance to temperature conversion table
- Simple 2-wire resistor-like component
- Work at any voltage
- Accurate to ±0.25°C with accurate analog converter
- Very inexpensive
- Sold water-resistant

Cons:

- Not durable
- Needs analog converter

4.7 Wind Speed Sensor

When finding the right sensors for a water and air condition data buoy, we must take into consideration what kinds of information we will want to monitor and record. Since we have determined we will be observing air temperature, we decided that we should add wind speed as well. Wind speed is a valuable feature to our system because too strong of winds can cause bad weather or rough waves. Also changes in recorded wind speed can also help for future observations and readings.

In this section we will be researching various ways to implement a wind speed sensor into our data buoy. The sensor must still be small enough to be integrated into the buoy system housing. Also, it must be waterproof in the chance of bad weather and rain, as well as rough water conditions, where waves can come over the top of the buoy itself. After we have done extensive research and comparing multiple sensors, we will determine which sensor will be most suited for our system and look into pricing and availability. For any case that the sensor is not available we will do more research and pick the second best suited sensor.

4.7.1 Various Options

When looking into the various options of wind speed sensors, we found that there are not many options for the type of configuration we are looking for. In order to be implemented into our data buoy system, we must find a sensor that can be used through a microcontroller. Many of the sensors found on the market today are used through data loggers or have their own built in Graphical User Interface or GUI. For the sensors we have found to work with our data buoy system, there are three that stick out. The sensors are all 3-cup anemometers made for measuring wind speed. The first one is the S-WSB-M003 Wind Speed smart sensor. Next, we look into the FST200-201 Wind Speed Sensor. Last, we researched the Adafruit Anemometer Wind Speed Sensor.

4.7.2 General Descriptions

The S-WSB-M003 is a smart wind speed sensor that is made for using a HOBO station data logger. The smart sensor is made for being easily implemented for plug-in modular connection to the HOBO system. All sensors are then stored inside the smart sensor, which automatically communicates configuration information to the logger without the need for any programming or extensive user setup. Although the wind speed sensor is made for a certain data logger, we can alter the setup and implement it into our own system. The measurements are made by averaging the wind speed over a short interval period. The sensor and cable jacket are made to be waterproof for purpose of rain, although we will need it for rough waves as well. The sensor comes with multiple cable lengths available to make different distances capable if needed.

Next, we look into the FST200-201 Wind Speed Sensor. With excellent resistance against impact, overload and shock, this sensor is made for durability and can withstand most harsh environments. Although very durable, the physical device is small and exquisite, making it easy to transport and simple to install. There is no shortage of quality with this sensor. It has an alloy metal finish with a reasonable structure design making a nice outlook condition. This sensor is would be a good addition to our system because we need something durable and able to withstand the sun and water.

Last we researched the Adafruit Anemometer Wind Speed Sensor. This device has been well-made and is designed to sit outside and measure wind speed with ease. It uses a simple 3-wire configuration to power and transmit signals to any output device. The sensor itself is rugged and easy to mount. The cable can be easily connected and disconnected with a few twists. The connector has been made waterproof for harsh weather and rain. Another reason this particular device seems to be a good representative is because it is plastic and won’t be able to rust or corrode with the salty or wet conditions the buoy will be used in. Though this looks like the best sensor to be used so far, we are limited on specifications being that the datasheet can only be found in Chinese. This causes difficulty being that none of the design team members are Chinese.

4.7.3 Technical Description

When determining the right wind speed sensor for our data buoy system we must compare and contrast each device. Each sensor has its own advantages and disadvantages from physical material used to build the component to price to how the component can be implemented into the data buoy system.

The S-WSB-M003Wind Speed Smart Sensor is made of a hardened beryllium shaft with a three cup polycarbonate anemometer using modified Teflon bearings. This gives the advantage of being rugged and able to withstand any harsh environment. The sensor has 2 channels with 8 bits of data for each channel. The measurement range is from 0 m/s to 76 m/s or (0 to 170 mph). The accuracy is ±1.1 m/s (±2.4 mph) or ±4% of the reading, whichever is greater. The resolution of the device is around 0.5 m/s (1.1 mph). The starting threshold is ≤1 m/s (≤ 2.2 mph). The service life is typically greater than 5 years, which is a factory replaceable mechanism. The overall operating temperature will be more than enough falling between -40°C to 75°C (-40 °F to 167°F). The disadvantage to this component is that it does not have a measurement averaging option. This is need since we will be taking averages over periods of time.

The FST200-201 is a great option for our system because its main applications are for weather stations and environment testing and control. The output signal of this device can be a current mode, voltage mode, or pulse mode with a working voltage of 0 - 5 V. The current used in this range is varied between 4-20 mA. The device can measure wind speeds from 0.5 m/s to 50 m/s (1.1 mph to 112 mph). The start wind seed is less than 0.5 m/s (1.1 mph). The accuracy is ± 0.5 m/s (1.1 mph). The max limit wind speed is 70 m/s (157mph) for 30 minutes. The working environment temperature is between -20°C and 85°C (-4°F to 185°F). the disadvantage of this sensor is that it is made in Deutschland and may not be available in the United States.

The last sensor that we are researching and comparing is the Adafruit Anemometer Wind Speed Sensor. This sensor uses an analog voltage output for easy use of conversions from voltage to wind speed. The output signal is similar to that of the FST200-201 and falls between 0.4 V to 2 V. The testing range of the sensor is from 0.5 m/s to 32.4 m/s (1.1 mph to 72.5 mph). The starting wind speed is around 0.2 m/s (0.45 mph). one big advantage is that is has a resolution of 0.1 m/s (0.225 mph) and the accuracy as low as ±0.5 m/s (1.1mph). One advantage to this sensor is that it has an equation for conversion. The analog output signal voltage can be converted to wind speed with the following equation:

$$\mathbf{Wind\ Speed\ Value}\ \left(\frac{\mathbf{m}}{\mathbf{s}}\right) = \frac{(\mathbf{Output\ Voltge} - \mathbf{0.4})}{\mathbf{1.6}} * \mathbf{32.4}$$

This equation gives the wind speed converted from voltage to meters per second. From there we can calculate the wind speed in miles per hour. Comparing the three different sensors we find many advantages and disadvantages for each particular component. The key comparisons of specifications can be seen in Table 4.5.2a.

Table 4.7.3. Anemometer Comparison

	S-WSB-M003	TFD200-201	Adafruit Anemometer
Working Voltage	12-24 V	12-36 V	7-24 V
Testing Range	0-76 m/s	0.5-50 m/s	0-32.4 m/s
Accuracy @ 25°C	±1.1 m/s	±0.5 m/s	±1 m/s
Resolution	0.5 m/s	0.25 m/s	0.1 m/s
Cost of Component*	$239.00	$153.00	$44.95
Material	Polycarbonate/Beryllium	Metal alloy	Polycarbonate
Waterproof	YES	YES	YES

*Comes with sensor and wiring

4.7.4 Pros and Cons

Looking back on all the advantages and disadvantages, as well as the specifications, we can put all the gathered information together and choose the component we find most suitable for our data buoy system.

S-WSB-M003

Pros:

- Small and lightweight
- Wide Test Range
- Waterproof
- Digital
- Long Lifespan > than 5 years

Cons:

- Not as accurate

- Low Resolution
- High Starting Threshold
- No Measurement Averaging Option
- Mostly used with Data logger
- Very Expensive

TFS200-201

Pros:

- Easy Assembly
- Small and lightweight
- Good Accuracy
- High Resolution
- Durable
- Wide Test Range
- Low Starting Threshold
- Waterproof

Cons:

- High Working Voltage
- Sold in Deutschland
- Expensive

Adafruit Anemometer

Pros:

- Small and Lightweight
- Easy to Install and use
- Low Starting Threshold
- Low Working Voltage
- Good Accuracy
- High Resolution
- High-Quality and Durable
- Waterproof
- Simple Conversion Equation
- Current, Voltage, or Pulse Signal Output
- Inexpensive

Cons:

- Smaller Testing Range

4.8 Code and Software

For the software aspect of this project, we will be programming the microcontroller using the C-language and its libraries. The environment we will be using will be Code Composer Studio from Texas Instruments or the Keil uVision IDE. Using C will be an advantage over the assembly language needed to program the MCU since we already have a broad knowledge of the language, its syntax is friendlier, and the amount of resources online is unmatchable, overall the ease of use fits our project perfectly. The compiler however will be in charge of converting our C-code to Assembly, in order to interface with the hardware.

When dealing with Assembly language we would have to worry about a whole new language (in a certain extent) since most microcontrollers have their own format for their assembly instructions, we would also have to learn how to control each register individually, which increases complexity. Many mistakes could be made, such as erasing pertinent data or misdirecting it. Also, the creation of routines would make this task be more challenging since for example a For-Loop is one line in C, in Assembly it takes about 3 or 4 lines.

By opting to develop in C, we decrease the amount of time needed for software development. This time can therefore be used elsewhere. Additionally, C language allows to do simple computations with ease, for example, diving two numbers and storing them in a variable (with a meaningful name) to use later requires a simple line of code, while Assembly would require several lines of code and memory/register accesses. Lastly, debugging the program will be easier and if we go back to it to add more features, the code can be easily adjusted and reused.

4.8.1 Flowchart

The following flowcharts show the program flow and how it is supposed to function:

General Program Flow: When the device is powered up, it initializes the components (clocks, pins, inputs, outputs) it then enters some type of infinite loop that will constantly but timely check the sensors connected to the input pins. Water Salinity, Wave Height, Water Temperature, and Wind Speed, each at their own rate since some of them data won't change abruptly in a matter of seconds, or even minutes (Water Temp and Water Salinity), and others will need more frequent updates (Wave Height and Wind Speed). Once it has acquired some data, it will save it to a buffer in memory. When the buffer becomes full, it will transmit this meaningful data wirelessly. The main loop will also check for user requests, where it will transmit the contents of the buffer and then empty it.

Depending on the rates at which the data will be collected, we will decide on how big the buffer will be. Also, whether or not the full contents of the buffer will be transmitted or just the most recent data, since transmitting data wirelessly will require a lot more power and we are aiming for a low-power consumption application.

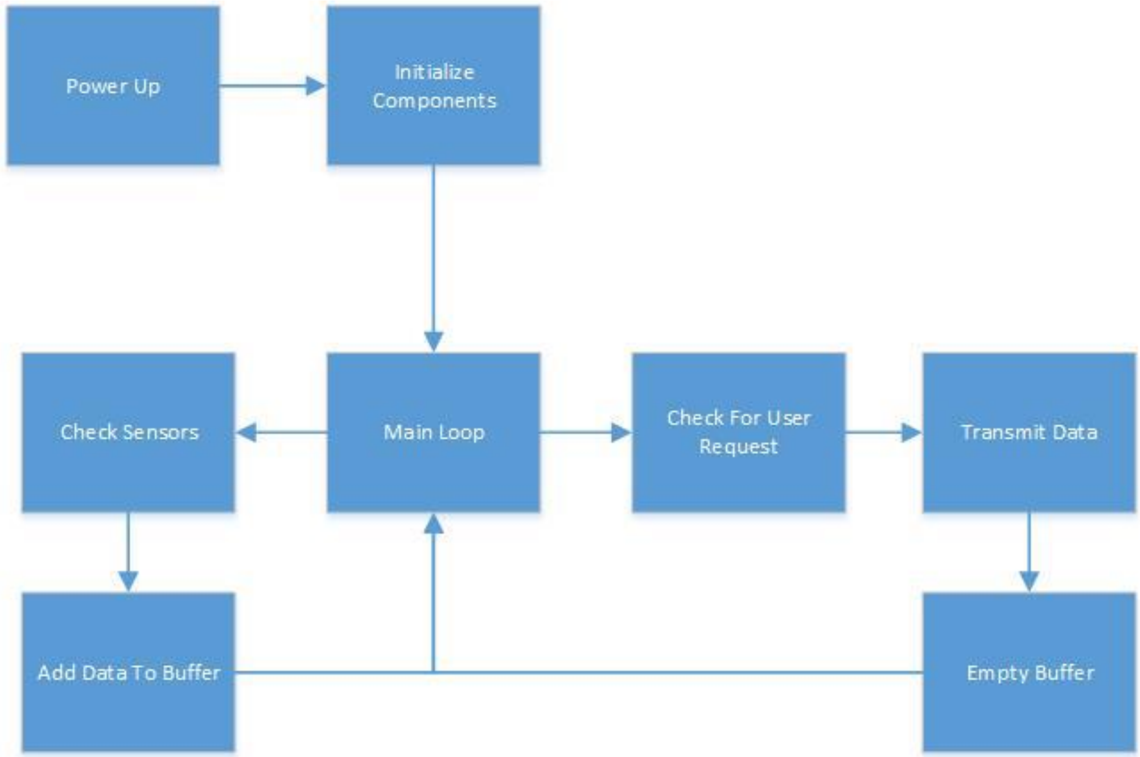


Figure 4.8.1.a: General Program Flow

Power Up and Initialization Flow: The following flowchart shows the power up and initialization sequence in more detail. Once the device is turned on, the power is directed to the Wi-Fi module and the microcontroller, after that it is then directed to the sensors individually. Then all the components (input and output pins) are initialized. After that it continues to the Main Loop.

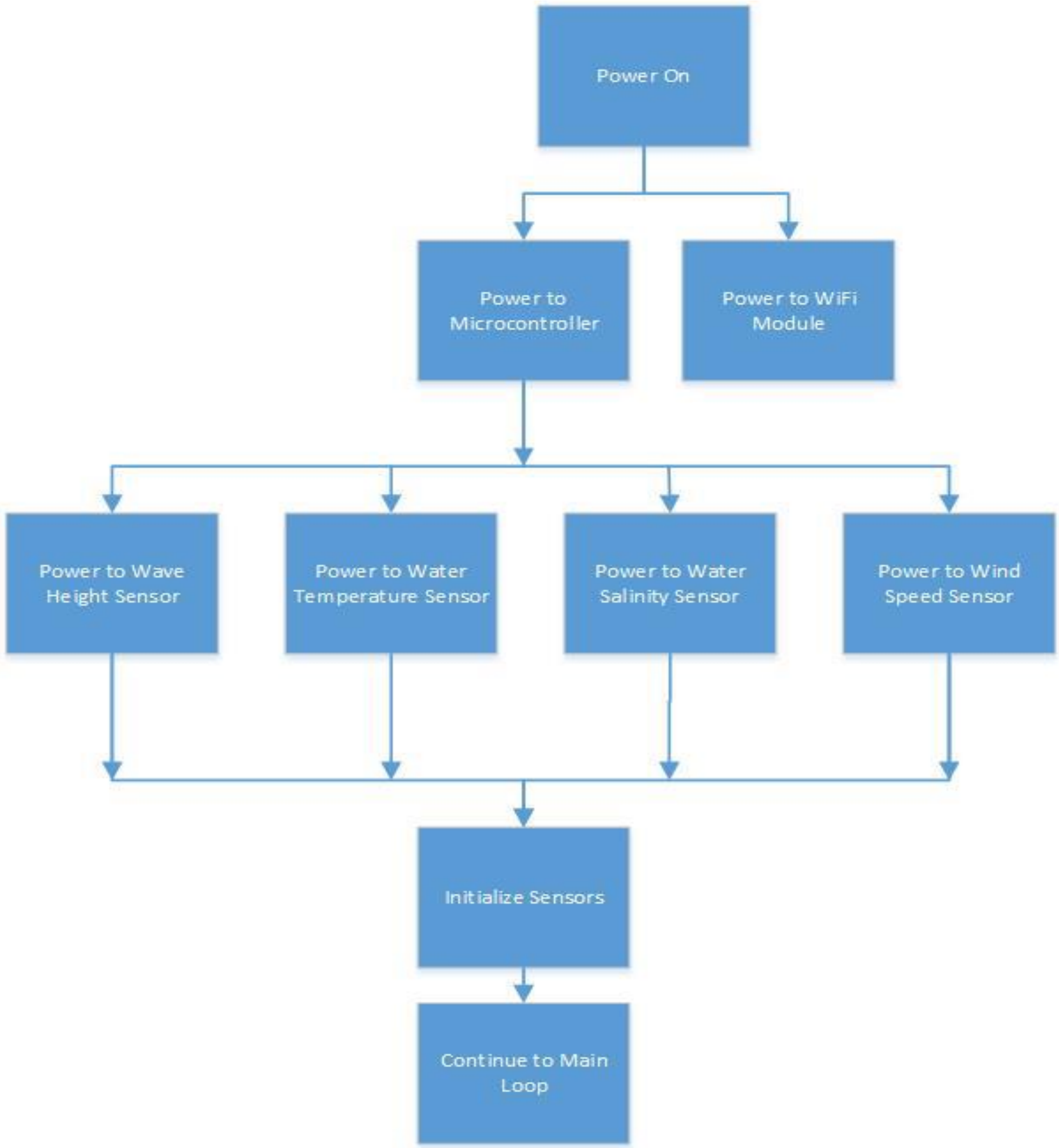


Figure 4.7.1.b: Power up and Initialization Flow

4.8.2 Libraries Used

Throughout this section we will be discussing the valuable libraries that were of importance to the programming of our design. From these libraries we will see some of the functions that were stated in the earlier section 3.2.2. We will go over the reasons for using these libraries and in the next section explain how the specific functions in these libraries will be implemented for integration of our buoy data system.

Digital I/O- First, we will be using the digital input/output function. This Function is designed to make reading and writing from a digital sensor or component much easier to implement in the programming of the microcontroller. The Digital Read function will read the value from the specific digital pin on the microcontroller where the sensor is connected and realize either high or low. The Pin Mode function allows certain pins to act as either inputs or outputs.

Analog I/O- For this library, it can be used in the same way as the digital I/O. There are read and write functions. These allow the user to determine if they want to read or write from the microcontroller from a certain pin to the sensor or component. This function will be needed in order to read the data being recorded from our wind speed sensor since it has an analog output.

Time- We will be using the time library so that we can recorded the exact time when transmitting the data from the microcontroller to the computer. This library also uses the delay function that will be needed in order to set a period between reading and writing to the microcontroller. This function also gives the microcontroller time to complete all the tasks before running another cycle.

Math- The math function library will be a very valuable resource to our programming of the microcontroller and implementing the sensors as well. In the math library we can find functions like min and max or mean and absolute. These functions allow the user to simply do math for any parts that calculations are needed. In our project design we will need to implement the microcontroller to be able to take average of data while the sensors are recording. For example, the accelerometer that will be used for the wave height will monitor the motion of the waves. In order to do this, we will need to take an average of a certain number of recordings so that it is not reading too little if a sample and giving inaccurate readings. We can also use this for averaging any other data recorded by any of the other sensors.

Interrupts- This library contains some of the functions that allow us to determine when the microcontroller needs to start monitoring and recording information found from the sensors. One of the functions is interrupts, which lets the microcontroller to allow certain important tasks to be done in the background and are enabled by default. Some functions will not work while interrupts are disabled and incoming communication may be ignored. Interrupts can be disabled as well in any part of the code so that other functions can work as intended.

Communication- The Communication library will be very valuable for the wireless interfacing of the device. When the device is awake and recording data it will need to send text data from the microcontroller to the computer, but the data must have a place to be visualized. This is where the function Serial come in to play. The Serial function allows the Tx and Rx or transmit and receive pins to communicate using TTL logic levels. The Arduino IDE has a built in serial monitor where the data can be seen and monitored.

oneWire- The One Wire library is common for simple sensors such as the temperature sensor we are using for our buoy design system. The one wire library is used for simple sensors that use single wires for sending data from the sensor to the microcontroller via analog or digital input pins. The DS18B20 Temperature Sensor we are using is only three wires; one for input voltage, one for ground, and the last for sending data through.

DallasTemperature- The Dallas Temperature library was also created for making the DS18B20 digital temperature sensor easier to implement. It uses a series of calculations and control to convert the digital output from the sensor into a logical output from the serial monitor on the computer.

4.8.3 Functions Used

The following discussion describes the functions that we will be using in the design of this system. To begin, the buoy will be sending the current date and time as well as the data being monitored by the sensors. For the microcontroller, we will need serial functions so that we can send the text data that the sensors are monitoring to the computer from the wireless communication system on the buoy. Certain functions will need to be for startup and putting back to sleep mode. Also, there will need to be different times when the buoy begins and ends sending the data which can be implemented using the interrupts functions. There will be other functions that will be able to read and write for both digital and analog inputs and outputs. Lastly, we will need some functions from the math library to do some calculations with the data recorded by the sensor.

Avg – This function is part of the math library and will be used for averaging some of the data sent to the microcontroller from certain sensors. One example is the wave height sensor. Since the data will only be sent every so often to the computer, this data will need to be averaged over a certain period of time being that no single recording will be accurate.

analogRead- We will be using this function so that any sensors recording an analog signal can be read through a certain pin set on the microcontroller. One example will be the wind speed sensor since it is output an analog voltage being sent to the microcontroller.

digitalRead- This function is used so that we can set specific pins to read digital signals. Some of our sensors have digital outputs and need the microcontroller to understand what it will be receiving.

4.8.4 Component Sourcing

Battery- The battery we ordered was found on the Adafruit online store and was order at the same time as the charging system and the solar panel to combine shipping costs.

Solar Panel- To get the Solar Panel we also found it online. It was order from Adafruit online store and was combined in shipping with the battery and the charging system.

Charging- To get the charging system we had to go online to order it. We found the one needed on the Adafruit online store. The charging system was ordered at the same time as the solar panel and the battery.

Microcontroller- We decided to order the Atmel ATmega to order online being that they were not readily available near us. We were able to find chip sockets on the same electronics shop site so we ordered them at the same time. These parts were ordered from Jaycon Systems website.

Photodetector- The photodetector was purchased from mouser electronics due to lack of availability near us.

Laser Diode- The laser diode was purchased from an online website called light in the box. It was purchased in a quantity of five.

Mirror- The mirror for the laser diode and the salinity system was also purchased online. The online site is called EdmundOptics. We also have some alternative mirrors in the CREOL building if needed.

Accelerometer- The accelerometer was also order online due to availability. We were able to cut and combine shipping costs since we order it from the same site as the Anemometer. The accelerometer was bought and shipped from Adafruit online store.

Anemometer- As mentioned above, the Anemometer was order from the same online store at the same time as the accelerometer. This saved shipping cost being that they were able to be combined in shipping. The Anemometer was bought from the Adafruit online store.

Temperature Sensors- The two temperature sensors, one for water temperature and one for air temperature, were found online being that there are no sources around that have availability. They were ordered from China on Amazon.

Buoy Housing- The buoy housing was easy to find, being that we were able to purchase it from our local Wal-Mart. This also saved us from shipping and handling costs.

Wireless Transceiver- The wireless transceiver was bought online due to no sources local having anything to meet our specifications. It was order online from Ebay and shipped from China. The shipping was free although it took almost a month to arrive.

Printed Circuit Board- The printed circuit board has not been ordered yet, although we have a few places in mind. If we decide to order online, we will go through expressPCB or OSHPark. This will delay the time of getting the part in hand, but may be the better option overall. If we decide to go locally, we will use Jaycon Systems. This company is nearby and it would be much faster which helps in any case of first-print error.

4.9 Comparing Wireless Signals

In this section of the paper we will be discussing and comparing the multitude of different wireless signals such as Bluetooth and Wi-Fi for example. Being that our buoy will need to transmit text data wirelessly to a user's computer, this is a very significant part of our overall design We will be explaining the different uses for each different type of components and what they are best used for. With the used of tables, graphs and flowcharts, we will be able to determine the best suited type of wireless connection for the user interfacing in our buoy design. Once we have researched all the different ways of sending and receiving data from the buoy, we will determine which is the most valuable and reasonable signal for us to work with.

In this table we show a broad comparison between the two main types of wireless communication. When looking into Bluetooth and Wi-Fi specifically we can compare the following general as well as technical means of wireless transmission. In the general purpose, we can see that Bluetooth is the better choice in terms of cost being that finding a low cost Bluetooth module is fairly easy and Wi-Fi can sometimes be somewhat costly. Although, this goes hand-in-hand with the functionality of the modules and their capabilities. From which, one of the capabilities that Wi-Fi has over Bluetooth is that the Range of use is much better, having the typical range of around 30 to 100 meters while the Bluetooth only ranges around 5 to 30 meters.

As for following the requirements of our design, the Bluetooth have a relatively low power consumption and which is needed for our design so that we are not wasting too much battery sending data to the computer. Comparing the uses of Wi-Fi and Bluetooth, they are for the most part used for the same applications. Bluetooth is found mostly with mobile phones, computer accessories and automation devices, while Wi-Fi is found with servers, televisions, latest mobile devices, and notebook computers. When it comes to specifications and standards, Bluetooth has their own department called Bluetooth SIG, while Wi-Fi uses the Institute of Electrical and Electronics Engineers and the Western Electrical Contractors Association.

Table 4.9. Comparing Bluetooth to Wi-Fi

	Bluetooth	Wi-Fi
Frequency	2.4 GHz	2.4, 3.6, 5 GHz
Cost	Low	High
Bandwidth	Low (800 Kbps)	High (11 Mbps)
Specifications authority	Bluetooth SIG	IEEE, WECA
Security	It is less secure	Security issues are already being debated.
Year of development	1994	1991
Primary Devices	Mobile phones, mouse, keyboards, office and industrial automation devices.	Notebook computers, desktop computers, servers, TV, Latest mobiles.
Hardware requirement	Bluetooth adaptor on all the devices connecting with each other	Wireless adaptors on all the devices of the network, a wireless router and/or wireless access points
Range	5-30 meters	With 802.11b/g the typical range is 32 meters indoor and 95 meters (300 ft) outdoors. 802.11n has greater range.
Power Consumption	Low	High
Ease of Use	Fairly simple to use. It is easy to switch between devices or find and connect to any device.	It is more complex and requires configuration of hardware and software.
Latency	200ms	150ms
Bit-rate	2.1Mbps	600 Mbps

There are two main standards for wireless communication, come of the key differences between them include portability, power consumption, range, application, security, and speed.

4.9.1 Wi-Fi

Wireless LAN is one of the main wireless interface for embedded systems, it is used in order to communicate devices with the monitoring or controlling counterparts using the TCP/IP mechanism.

There are many hardware integration constraints such as the size, power consumption, and supporting components. A well-developed Wi-Fi subsystem must include a power management unit that helps decrease the amount of power that will be consumed. While on the software side of things we have to worry about the type of interface between the microcontroller and the module itself, it all depends on the complexity of the code and the data throughput of our application. Since embedded systems, like ours require low-power interfaces, the most popular and commonly available in small (16-bit or 8-bit microcontrollers) are the SPI (Serial Peripheral Interface) and the UART. While the Universal Asynchronous Receiver/Transmitter (UART) is likely the easier of the two interfaces to use, it does not provide a very high throughput (only about 115 kbps) compared to the SPI's 15 Mbps. There are several categories within the networking standard 802.11.

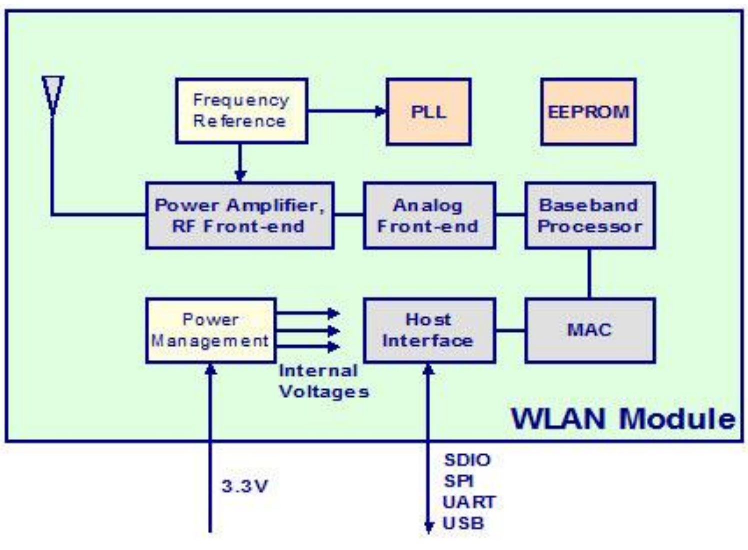


Figure 4.9.1. WLAN Module Flowchart

4.9.2 Bluetooth

This is the low-cost, low-power, wireless technology used in embedded systems today. The tradeoffs are also very limiting, since it has relatively low baud rates and its range is considerably shorter. Though, the more power it consumes, the longer its range can be. These are actually categorized into 3 separate classes.

Table 4.9.2. Comparing Bluetooth Classes

Class Number	Max Output Power (dBm)	Max Output Power (mW)	Range
Class 1	20 dBm	100 mW	100 m
Class 2	4 dBm	2.5 mW	10 m
Class 3	0 dBm	1 mW	1 m

For the hardware implementation of a Bluetooth module, a radio and a link controller are needed as well as a baseband and RF. As for the interface between the microcontroller and the Bluetooth component, it would be using the SPP (Serial Port Profile) instead of the UART.

4.9.3 GSM

The Global System for Mobile Communications, used mostly in the mobile communications industry, with about 90% of the market share and roughly 2 billion users worldwide. It was first implemented in 1991, and since there have been several optimizations commonly referred as 2G, 3G, and most recently 4G (LTE Advanced).

GSM supports various frequencies, such as 900 MHz, 1800 MHz, and 1900 MHz. Depending on the country or the continent, they use different ones. For example, in North America, the 1900MHz is used.

The frequencies are also known as Frequency Bands; there are three different types. Dual Band, Tri-Band and Quad Band. GSM has the ability to transmit 64kbps to 120 Mbps of data. It has a low cost of implementation and a high quality service.

4.10 Wireless Communication Implementation

During this section, we will be researching multiple ways of sending our data recorded by the buoy wirelessly to the user's computer. We will research a variety of different modules and find the best suited way to implement wireless text data transfer for our system. We will compare a few different Wi-Fi modules as well as Bluetooth modules. This will give us a better comparison to what our final decision on how we will be interfacing our data buoy system. We will eventually decide between Wi-Fi or Bluetooth, and then determine which specific wireless module we will want to use. In the following few paragraphs we will see the various components on the market today and observe the specifications and technical details found throughout our research.

4.10.1 Wi-Fi Module

The first one would be Texas Instrument's CC3000 Module. "The CC3000 module is a self-contained wireless network processor that simplifies the implementation of Internet connectivity and using TI's Simple Link Wi-Fi solution minimizes the software requirements of the MCU and as such, it is the ideal solution for embedded application which uses a low cost and low power MCU." (TI SimpleLinkTM CC3000 Module – Wi-Fi 802.11b/g Network Processor, 2012)

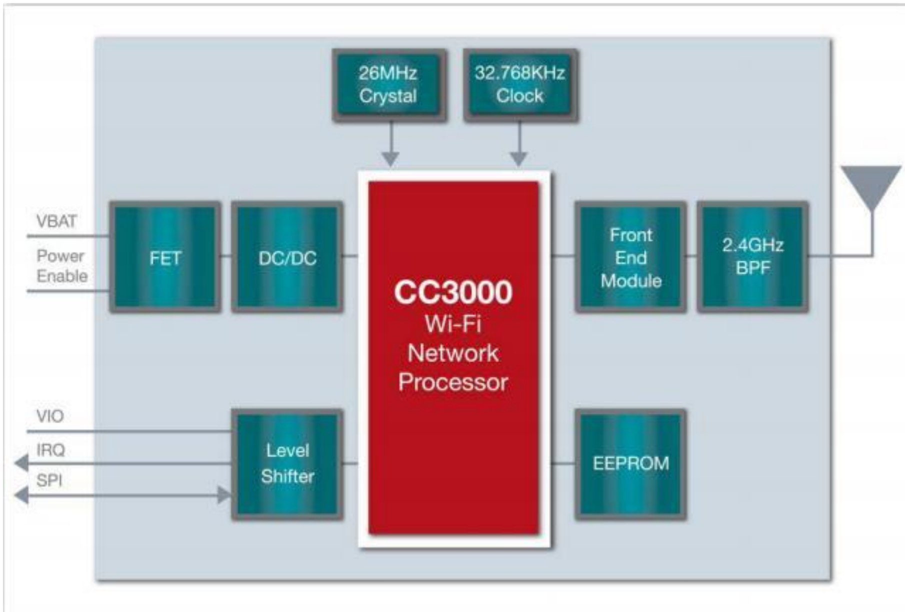


Figure 4.10.1. Wi-Fi Solution for TI Simple Link CC3000 Module, reprinted with permission from TI

With the TI's schematic we are able to convert it to a PCB layout in order to print it. In addition to the schematic, TI has a design recommendations section where they describe the layout for the CC3000 module, the RF trace, and the antenna that will be used. It was recommended for us the use of an antenna, as TI recommends using the antenna on a separate board, or cut the board so that the antenna will not interfere with the RF lines and signals. TI has implemented the CC3000 design into a few different modules, such as the Wi- Fi SMD Module, the Wi-Fi Shield, and the Wi-Fi Breakout.

The second Wi-Fi module we will be discussing is the XBEE Wi-Fi module with a wire antenna available from DIGI. The XBEE Wi-Fi "Embedded RF provides simple serial to IEEE 802.11 connectivity. By bridging the low-power/low-cost requirements of wireless device networking with the proven infrastructure of 802.11, the XBEE Wi-Fi creates new wireless opportunities for energy management, process and factory automation, wireless sensor networks, intelligent asset management as well as other features. This module gives developers the fastest IP-to-device and device-to-cloud capability possible. Focused on the rigorous requirements of these wireless device networks, the module gives developers IP- to-device and device-to-cloud capability" (XBee Wi-Fi) Needless to say, all three modules prove to be useful if we decide to embed any of them into our own design. To discuss this further, we have created a table and all their summarized specifications can be viewed in figure 4.10.1. below:

Table 4.10.1. Wi-Fi Module Specifications

Wi-Fi Module Comparison		
	CC3000	XBee
Transmit Current	260 mA	309 mA
Receive Current	92 mA	100 mA
Supply Voltage (DC)	3.6 V	3.5 V
Frequency Band	2.4 GHz	2.4 GHz
Cost	$34.99	$17.50

From the table we can see that comparing the two Wi-Fi modules they are very similar. The current is low and makes for keeping the overall system power low. Low power usage will be valuable when we are needing to send the text data that will be monitored by the buoy.
From the specifications summary table, it is worth noting the Texas Instrument's CC3000 module is a good fit regardless of its power consumption, and would be ideally a more suitable option if we decide to go with a Texas Instruments microprocessor.

4.10.2 Bluetooth Module

The first Bluetooth module that we are going to look at is made by Roving Networks, it's the Bluetooth SMD Module RN-42. It's designed to replace serial cables, it's supposed to be compatible with the previous version (RN-41), but the main difference between the two is that the RN-42 is a class 2 device, meaning is about 10-20 meters and its power consumption decreased. The RN-42 is perfect for short range, low-power consumption applications (specially battery-powered). It drives a 26uA current in sleep mode while still being able to maintain a discoverable mode. It supports UART Data rates of up to 3Mbps, and over the air rates of 721kbps to 2.0Mbps. Requires a 3.3V to operate.

Next, the RN-41 Module, which is a Class 1 Bluetooth device, meaning that it has the maximum possible range, but drives the most power consumption of about

250uA in sleep mode (while still being able be discoverable and connect properly). It is still recommended for battery-powered applications, and by default it is ready to use the SPP (Serial Port Profile) configuration. If no antenna is needed or an external antenna is included in a design, the RN-41N can be used. It has UART rates of up to 3Mbps and over the air data rates of 721kbps to 2.0Mbps.

It operates at an average of 3.3V. Both modules have the same parameters in the following table.

Table 4.10.2.a. Class 1 Bluetooth Specifications

Parameter Name	Value
Bluetooth Classic-Data/SPP	Yes
FCC Certified	Yes
ASCII Interface	Yes
Memory Type	Flash
GPIO	11
Pin Count	35
Package Type	Surface mount module
Package Size	13.4 x 25.8 x 2.4 mm
Min Temp Range	-40C
Max Temp Range	85C
Op Voltage Min	3.0V
Op Voltage Max	3.6V

The last Bluetooth module we will be researching is the HC-06 Bluetooth Wireless Serial RF Transceiver. This Bluetooth device can be found in many open source projects online using the Atmel aTmega328p, in which case it will make implementing and integrating into our system very simple. This module has a built in 2.4 GHz antennae and is powered with the low working voltage of 3.1 to 4.2 volts. The HC-06 has a very simple circuit as well as very small overall size that makes it extremely easy to implemented into our data buoy and configure with the rest of our sensors and components.

Table 4.10.2.b. Bluetooth Module Summary

Bluetooth Module Comparison			
	RN-41	RN-42	HC-06
Class	1	2	2
Current (Pairing)	8 mA	12 mA	8 mA
Current (Communicating)	30 mA	25 mA	30 mA
Voltage	3.3 V	3.3 V	3.1 V
Range	20 m	30 m	10 m
Bluetooth Version	2.1/2.0/1.2/1.1	2.1/2.0/1.2/1.1	2.0
Cost	$39.99	$42.00	$4.99

In the previous table we can see the comparison of the three Bluetooth modules researched. From the data collected, we are able to take the valuable information gathered and put it toward out decision of which module we may be sing for our data buoy design. After determining that the buoy will be using Bluetooth we can precisely pick the exact module that will be used for our design project.

4.10.3 Wireless Communication Module Schematic

If we use the TI microcontroller, it would be perfect to also use the TI wireless communications module since it would be compatible right from the get go. The following schematic taken from TI's website helps us understand the module's design.

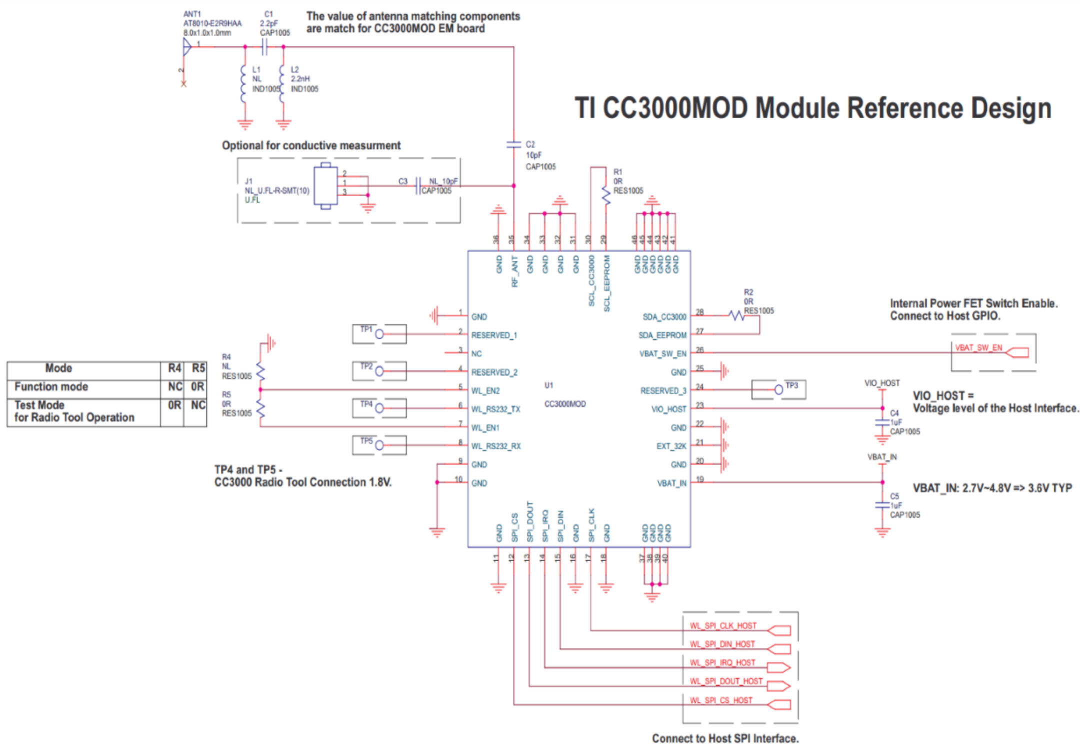


Figure 4.10.3. CC3000 Module Reference Design (Reprinted with permission from TI)

4.11 Microcontroller

A microcontroller, also known as MCU, is basically a very small computer on an integrated circuit. MCUs generally contain a processor core, memory, and programmable input and output peripherals. In order to integrate all our sensors – air and water temperature, wind speed, wave height, and water salinity– we need to select a microcontroller that is equipped with enough input and output pins, has sufficient memory to store the recorded data, and is capable of handling this information and transmit it wirelessly for the user to read when needed. In the following sections we will briefly discuss the various options we have narrowed down to from a wide variety of options. Later, we will get into more detail and describe each microcontroller unit separately while discussing key features that are essential for our project's functionality. Once we have discussed each microcontroller individually, we will begin comparing them in order to be able to choose the most appropriate one for our purposes.

4.11.1 Various Options

Selecting the most appropriate microcontroller for our project is one of the most essential parts of our design. This selection is one that determines whether or not our project can be a success or failure given that our microcontroller is the "brain" of our entire design. Several different criteria have to be taken into consideration in order to select the best option. Selecting the best option does not necessarily mean the one with the best performance or capacity. We will be taking into consideration not only operating frequency and memory capacity but also the price given that we are trying to build a low cost data buoy. Our goal is to be able to select the least expensive microcontroller while fulfilling system specifications such as low power, performance, reliability, amount of input/output pins, and other characteristics.

Some of the key features we have to take into consideration are: required number of I/O pins and ports, required peripherals, affordability, availability, clock speed, size, processing technology, power consumption, amongst others. One important characteristic we also have to consider is the operating temperature range given that our data buoy will be used in an outdoors environment in which solar power will be a key factor necessary for our project's functionality. Given this, the interior of the buoy housing might experience high temperatures and we have to make sure all of our components can support these elevated temperatures and not be affected performance wise.

After exploring several different microcontroller options available in today's market, we have selected three microcontrollers to further discuss previously mentioned characteristics and select the most appropriate for our design. These three microcontrollers are: the MSP430G2553 and MSP432P401R by Texas Instruments and Atmel's ATmega328P. In the following sections we will briefly introduce each microcontroller individually and we will be comparing all three based on their characteristics in order to select the one that is best suited for our design.

4.11.2 General Descriptions

The MSP430 is a microcontroller family from Texas Instruments that incorporates a 16-bit RISC CPU, peripherals, and a flexible clock system that are interconnected using a von-Neumann common memory address bus and memory data bus. This microcontroller family is designed for low power consumption applications. It is very small, inexpensive, and easy to work with. Also, it comes with a number of built-in peripheral devices. Within the MSP430 microcontroller family, several different series exit. We will be taking a look at the MSP430G2553 microcontroller. Some of the key features for the MSP430G2553 microcontroller include:

- Low power supply required for operation
- Ultra-low power consumption
- Multiple low power modes
- Ultra-fast wake up from standby mode
- 16-bit RISC architecture
- Basic clock module configurations
- Wide operating temperature range
- Supports SPI/I2C communication protocols
- Flexible clock system

- Affordable and available

The MSP432 is another microcontroller family from Texas Instruments. This microcontroller family is based on a 32-bit ARM Cortex-M4F CPU. Compared to the MSP430 family of microcontrollers, the MSP432 family has a larger address space for code and data and a faster integer and floating point calculation. Within the MSP432 microcontroller family, several different series exit. We will be taking a look at the MSP432P401R microcontroller. Some of the key features for the MSP430G2231 microcontroller include:

- Low power supply required for operation
- Ultra-low power consumption
- Multiple low power modes
- Flexible clocking features
- Wide operating temperature range
- Supports SPI/I2C communication protocols
- Affordable and available

The ATmega328P is a single-chip microcontroller from the megaAVR family created by Atmel. This microcontroller is capable of executing powerful instructions in a single clock cycle while having an ultra-low power consumption. More specifically, this microcontroller can achieve throughputs of approximately one million instructions per second (MIPS) per MHz. Some of the key features for the ATmega328P microcontroller include:

- High performance
- Advanced architecture
- Wide operating voltage range
- Wide operating temperature range
- Ultra-low power consumption
- High throughput
- Multiple I/O pins
- Affordable and available

4.11.3 Comparing Microcontrollers

Now that our three microcontroller options have been introduced, we will begin to discuss their different characteristics and compare one with the other in order to determine which microcontroller option has the most favorable features for the purposes of our project. Table 4.11.3 provides a detailed summary of the most important characteristics that have been taken into consideration when selecting the appropriate microcontroller for our design.

Table 4.11.3. Comparison of MCU characteristics

	MSP430G2553	MSP432P401R	ATmega328P
CPU	16-bit RISC	ARM 32-bit Cortex-M4F	8-bit AVR RISC
Frequency (MHz)	16	48	20
Non-volatile Memory (KB)	16	256	32
RAM (KB)	0.5	64	2
GPIO	24	84	23
I2C	1	4	1
SPI	2	8	2
ADC	ADC10 - 8ch	ADC14 - 24ch	ADC10 - 8ch
Timers (16-bit)	2	4	1
Timer (32-bit)	0	2	0
Wakeup Time (μs)	1.5	8	–
Operating Voltage Range (V)	1.8 – 3.6	1.62 – 3.7	1.8 – 5.5
Operating Temperature Range (°C)	-40 to 85	-40 to 85	-40 to 85

The MSP430G2231 microcontroller has a 16-bit RISC CPU, the MSP432P401R microcontroller is based on an ARM 32-bit Cortex-M4F core, and the ATmega328P microcontroller has an 8-bit AVR RISC CPU. Comparing operating frequencies for each microcontroller, the MSP432P401R comes in first with the highest maximum operating frequency at 48MHz while the ATmega328P and MSP430G2553 microcontrollers operate at maximum frequencies of 20MHz and 16MHz respectively. Again, by looking at flash memory characteristics for each microcontroller, the MSP432P401R has the highest memory capacity with up to eight times what the ATmega328P microcontroller has and sixteen times what the MSP430G2553 has.

All three microcontroller options operate at similar voltage ranges. The minimum operating voltage for the MSP430G2553 and the ATmega328P is 1.8V while the minimum operating voltage for the MSP432P401R is slightly lower at 1.62V. The maximum operating voltages for the MSP430G2553 and the MSP432P401R are almost the same at 3.6V and 3.7V respectively. However, the ATmega328P microcontroller has a significantly higher maximum operating voltage, 5.5V, compared to the other two microcontrollers. All three microcontrollers have a wide operating temperature range that extends from -40°C to 85°C. As mentioned before, this is an important characteristic to consider given that our data buoy will be exposed to the sun whenever in operation and the interior of the buoy housing can experience high temperatures.

The MSP432P401R also has the highest amount of general input/output pins with 84. The other two microcontrollers have 24 and 23 for the MSP430G2553 and ATmega328P respectively. However, for the purposes of our experiment, 84 I/O pins seems a little excessive and the pin count for any of the other two microcontroller options discussed should be sufficient. All three microcontrollers support both I2C and SPI communication protocols and are equipped with an analog to digital converter. The MSP430G2553 has two 16-bit timers and a 10-bit analog to digital converter. The MSP432P401R has four 16-bit timers and two 32-bit timers. Also, it has a 14-bit analog to digital converter. The ATmega328P has only one 16-bit timer and a 10-bit analog to digital converter.

The MSP430G2553 microcontroller is a viable option for our design given that it fulfills most of our requirements. It requires a low voltage supply for operation and has an ultra-low power consumption. This microcontroller has one active mode and five low power modes. Power consumption is 230μA at 1MHz in active mode and 0.5μA in standby mode. While in standby mode, this microcontroller has an ultra-fast wakeup feature of less than 1μs. Also, given that we have gained some knowledge by working with this microcontroller in previous courses, this microcontroller option might be easy to implement in our design. However, not everything is perfect. While we might have some experience working with this microcontroller, implementing the low power modes can be quite complex and might give us some trouble. In addition to the complexity of implementing the low power modes, this microcontroller has been associated with having issues with the clock signals being inaccurate.

The MSP432P401R seems to exceed our requirements with a very high maximum operating frequency, flash memory, and input/output pin count. Similar to the MSP430G2553, this microcontroller requires a low voltage supply for operation and has an ultra-low power consumption. Power consumption for this microcontroller is 85μA at 1MHz in active mode and 0.85μA in standby mode. While in standby mode, this microcontroller has an ultra-fast wakeup feature of less than 8μs. This is eight times longer than the MSP430G2553 but still very fast considering it is measured in microseconds. The drawbacks are that we have no previous knowledge nor experience for this microcontroller. This lack of experience might bring some potential issues our way while building and programming our data buoy. In addition to the lack of experience, this microcontroller is relatively new and there are very few resources available for the user to use as a guide. While there is no harm in having extra input/output pins or flash memory, this microcontroller might not be the best option because we would be wasting resources. However, this microcontroller might come in handy in the future if we decide to add more and more features to our data buoy.

The ATmega328P is also a viable option for our design given that it fulfills our requirements and at the same time it does not have an excessive amount of pins or flash memory. Similar to the two previously mentioned microcontrollers, this microcontroller requires a low voltage supply for operation and has an ultra-low power consumption. Power consumption for this microcontroller is 0.2mA at 1MHz in active mode, 0.1μA in power-down mode and 0.75μA in power-save mode. Also, this microcontroller can achieve high throughputs of approximately one MIPS per MHz by executing powerful instructions in a single clock cycle. Given that it maximum operating frequency is 20MHz, a maximum throughput of 20 MIPS can be achieved. A positive aspect about this microcontroller is that we have gained some experience by working with this microcontroller in the past and there is also a wide variety of resources available online which will be very helpful when it comes to programming our microcontroller.

4.12 Printed Circuit Board (PCB)

Being that one of the main requirements of our design project is to design and implement our own printed circuit board to hold our microcontroller and the power regulating components. A PCB is a critical component in our data buoy system because it will be the looked at as the “head” of the system being that it will the source that braches out to all other sensors and components. In this section we will be looking into some research about printed circuit boards and how to implement and integrate them into our data buoy system. We will look into the size requirements, how we plan to design the PCB, and what software we will use to draw our schematics. In the end we will look through a few places that offer printing and decide on which place has the best offer.

4.12.1 Requirements

As stated in the Requirement Specifications section of this document, we have taken into consideration what the requirements of a printed circuit board must be. When determining the required size of the buoy we must take into consideration all the components that will be mounted on the board itself. The board will be a place for us to mount our Microcontroller, a few passive components, and output pins for our sensors to be connected. Since the PCB will be mounted inside the buoy, it must fit through the hole mounted on the top of the housing, which is 5 inches in diameter. With these things in mind, we can keep the size generally small. The required size of the PCB must be less than 3”x4”. When determining the cost, we are going to be keeping it in a low range being we have a self-funded budget. With the design and printing taken into consideration, we are going to make the overall cost for our PCB under $75.

4.12.2 Design

When designing the printed circuit board, we will first look into what the actual board itself will be holding. The first and main component of the PCB is the microcontroller. The microcontroller is going to be the brains of the whole system and will be located centered of the board. This will make it easier for the traces reach their sensor pinouts. Knowing that most of the sensors will not be mounted on the PCB, we can have the board just host some analog and digital pinouts. The board will have to have two regulated voltages coming from the microcontroller as well. A 3.3 and 5 V output voltage supply are necessary for sending power to the pinouts for each sensor.

In the next page we have drawn a simple schematic to show what the basic and most generalized printed circuit board design. This will make drawing the schematic in a real PCB cad ware simpler and easier to follow. In the next section of the PCB research we will be looking into the different PCB cad wares that are available for use without having to pay excessive amounts of money.

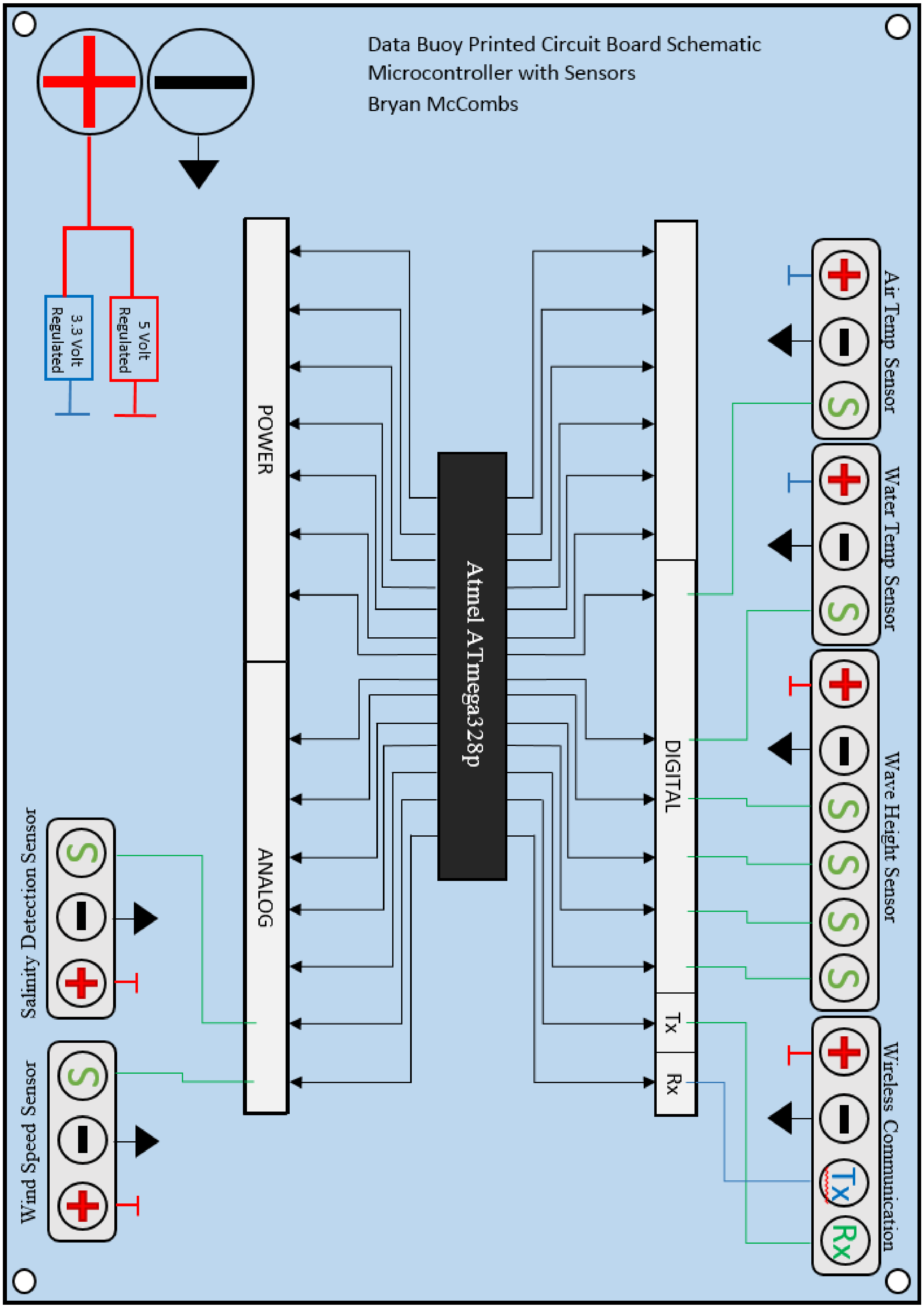


Figure 4.12.2.a. General Printed Circuit Board Schematic Design

From the previous figure we can see the microcontroller centered in the middle of the board with many traces outputting. The traces can be seen to go to the pinouts of the multiple sensors. Each sensor has its voltage supply's, ground's, and signal/data pin's. The voltages can be seen color code to determine if the sensor needs 3.3 V or 5V. The microcontroller shows the different power and ground pins, digital output pins, analog output pins, and transmit and receive pins. This schematic will be used to help draw the final PCB design once we decide which cad ware we will use.

4.12.3 Software

In this section we are going to research the Printed Circuit Board design tool called CadSoft EAGLE. This design software is free to use with certain restraints. Although there are certain things not provided for the free version of the software, it is sufficient for the purpose of our data buoy system. This software is designed to be able to make ease of drawing a detailed schematic using actual component parts down to their sizes and values. The software itself has a component library with practically any component you can think of. Once a layout of the PCB has been designed, like the previous figure, we can use EAGLE to draw the schematic with the specific components needed. Once the schematic is drawn, the software has an auto trace option that will draw the best and most efficient trace lines with no crossing. After finishing the design and laying the trace lines, the file can be sent to whichever circuit board printing company that we choose to use.

4.12.4 PCB Schematic

After taking into consideration the requirements for the PCB, putting together a list of components, and designing a layout of the board, we have created a schematic in the EAGLE software. This schematic is a very general and rough copy of the final board that we will have printed. After adding all the components, we can see from the provided schematic that it will be fairly small in size, which makes for better fit and more room inside the buoy housing for wiring and other components. The first draft schematic in the EAGLE software can be found in the final design section

5.0 Prototyping

While prototyping each component of this data buoy system, we take into consideration how the chosen sensors will be implemented into the system and why the feature is added. Next, we will discuss how the sensor works in line with the other components and features. We will make sure that the component or sensor is designed and implemented so that it will be ready for testing in the next section. Last we look into the physical appearance and configuration of each sensor and component and determine the best suited area on the housing for mounting so that we can monitor and record the most accurate readings.

5.1 Power

The power supply is one of the most important parts of our project. Without a power supply none of the electronic devices utilized in our design would function properly. Given the mobile nature of our project, we are limited to the use of batteries to provide the required power. More specifically, rechargeable batteries will be used in order to save the user the trouble of having to replace batteries once they have been discharged. Several battery chemistry technologies were discussed based on the characteristics that we thought would have a greater impact to our design. Some of these characteristics are the specific energy density, cycle life, overcharge tolerance, self-discharge, nominal cell voltage, maintenance and safety requirements, etc.

5.2 Charging

Charging our lithium polymer battery can be a tricky task. We have to be careful as to not overcharge our battery because these types of batteries do not have a high overcharge tolerance. Also, we must watch out for the charging rate and especially the charging temperature of the battery. The charging of our lithium polymer battery will be done by using Adafruit’s solar lithium polymer charger board. The simplicity of this board provides use with all the safety precautions required when charging or discharging a lithium polymer battery. Figure 5.X.X is a picture of the actual board that will be used to charge our lithium polymer battery. This board should be able to fit inside the buoy housing together with all other components that need be places inside. It is difficult to tell the size of this board just by looking at the picture but we have looked these dimensions up and stated them in a previous section. The board is small enough and should not cause an issues to place inside.

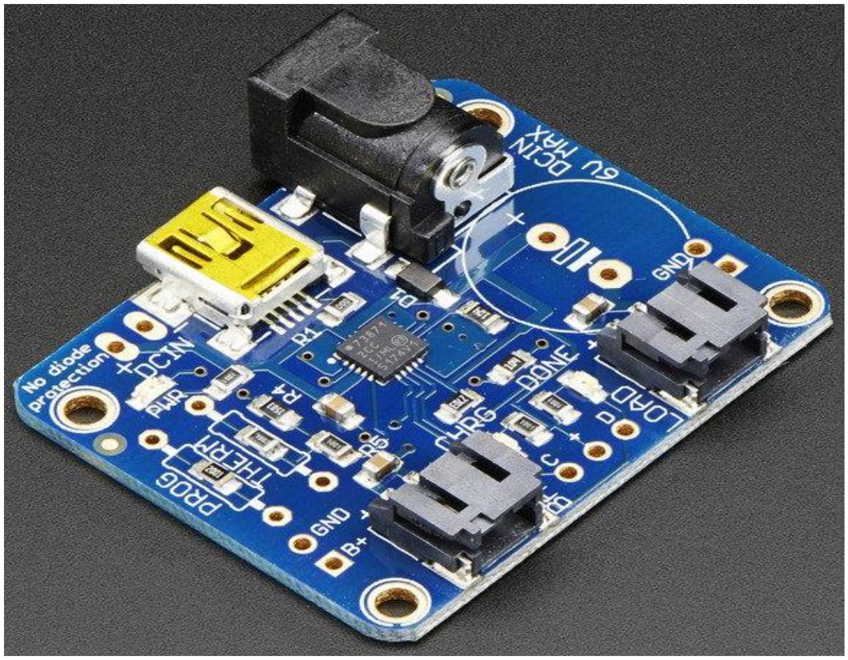


Figure 5.2 Solar lithium polymer charger board

5.3 Microcontroller

The microcontroller, along with the power supply, is one of the most essential components to our project. This component is the “brain” of our entire system. The microcontroller is in charge of reading data from all our sensors and transmitting this data wirelessly to a computer where the user can be able to read this data and have an accurate idea on how the weather and water conditions wherever the data buoy is located are at a given time.

The microcontroller has to be capable of processing signals –analog and digital– from the multiple sensors that will be required to record all the desired measurements. Also, once all these signals have been received, they have to be converted to understandable measurements for the user to read. This data will be transmitted wirelessly to a computer and the microcontroller is in charge of making this happen.

5.4 Salinity Detection

The prototype salinity sensor will consist of a circuit on an optical bench with a water sample in the lab. The laser diode will be aligned and focused into the fiber optic cable which will carry the light to the sample. The light will pass first through a purely distilled water sample until the sensor is seen to be consistently working and aligned. Different levels of salt will be added to determine how large of a fiber array is needed on the output to pick up on the beam divergence for the necessary range of salinity and will be amended as needed at this stage.

The sensor will then be placed inside the sensor housing and attached to the bottom of the buoy housing to be immersed in the water when the buoy is in use. The sensor housing will have to be sealed to the housing and the buoy waterproofed around the hole to prevent leakage.

5.5 Wave Height Sensing

One of our goals for this project is to build a buoy that can measure wave height and transmit the collected data for the user to have. After doing some research, it was clear that the best option to get the job done as accurately and easily as possible was to use an accelerometer. Four different accelerometers were discussed and the best one was chosen based on some key features such as scaling range, output type and communication interface, required supply voltage and current, and price.

Two accelerometers have been chosen for testing. Both are very similar in features such as number of axes, dimensions, weight, operating voltage range, and operating temperature range. They only differ in the scaling range and in the output type. The LIS3DH has a selectable scaling range of ±2g, ±4g, ±8g, or ±16g. On the other hand, the ADXL335 does not have a selectable scaling range. The scaling range for this accelerometer is set at ±3g. These two accelerometers also differ in the output type. This means that one accelerometer has a digital output and the other one has an analog output. More specifically, the LIS3DH accelerometer has a digital output type and the ADXL335 has an analog output type. This output type feature is the main reason why we have decided to test both accelerometers.

Both accelerometers are very small in size and also very lightweight. Given this, finding a location for the accelerometer inside the buoy housing seems to be an easy task. However, it can get tricky because of how an accelerometer measures acceleration. Both accelerometers have 3 axes: x-, y-, and z-axis. We only need to use one of these three axes. The chosen axis has to be aligned correctly in the vertical direction as to measure the acceleration of the buoy as it moves upward/downward. Now, the tricky part is that the buoy will not be strictly moving up and down as it floats in a body of water. As the buoy floats up or down mimicking a wave, the chosen axis might be aligned at an angle if it is fixed inside the buoy housing. This has to be taken into account because the axis chosen to measure the acceleration must be aligned vertically at all moments.

Further, both sensors are connected differently with the microcontroller board because one is digital and the other is analog. In the case of the LIS3DH, three power pins, four I2C/SPI pins, and one interrupt pin are available. The three power pins are Vin, 3Vo, and GND. The Vin pin is connected to the microcontroller and is used to power the accelerometer. The maximum voltage for operation is 3.6V for this accelerometer. However, the LIS3DH board comes with a 3.3V voltage regulator embedded on it in case a higher voltage supply is used. The GND pin is the ground pin and it has to be connected to the ground pin of the microcontroller. The other four pins are used for the communication interface. These pins are SCL, SDA, SDO, and CS. Depending on whether I2C or SPI is used, these pins are connected differently to pins on the microcontroller board. In the case of the ADXL335, seven different pins are available. These pins are Vin, 3Vo, GND, Zout, Yout, Xout, and Test. Similar to the LIS3DH, this accelerometer operates with a maximum voltage of 3.6V and it also comes with an embedded 3.3V voltage regulator which is very useful in case a higher voltage is used to power the sensor. The Vin pin is connected to the microcontroller and is used to power the accelerometer. The GND pin is the ground and is connected to the ground pin on the microcontroller board. The Zout, Yout, and Xout pins are the output measurements for acceleration in a corresponding axis. These are connected to an analog pin in the microcontroller board.

Figure 5.5.a: LIS3DH accelerometer board

Figure 5.5.b ADXL335 accelerometer board

Figure X on the left is a picture of the LIS3DH accelerometer board and Figure X on the right is a picture of the ADXL335 accelerometer board. The actual accelerometer sensor is the small chip located in the middle of both boards and that small square chip is what does the actual acceleration measurements that will be used to determine the wave height. All other components and pins are used for the overall functionality of the accelerometers. By looking at the accelerometer boards we can see the directions for the positive x-, y-, and z-axis. The x-axis runs from left to right of the accelerometer board, the y-axis runs from top to bottom of the accelerometer board, and the z-axis runs right through the accelerometer board making a perpendicular angle with the surface of the accelerometer board. One of these three axes will be used to measure the vertical acceleration of the buoy in order to obtain a wave height measurement. The selected axis will have to be aligned vertically at all times in order to obtain a reliable and accurate measurement.

5.6 Temperature Sensing

Now that we have determined that we will be using the Maxim DS18B20 Digital Temperature Sensor (formerly Dallas Semiconductor) for monitoring the air and water temperature for our data buoy system, we can begin prototyping. For this specific feature, our prototyping will involve finding the best location on the data buoy to be fixed, how they are implemented into the microcontroller, and how the decided sensors will work with our system.

First we will mention and start prototyping how the sensor are going to be implemented into our system and microcontroller. From online sources we have found that the sensor is in fact very simple to use. The sensor only has three wires: a supply voltage, a ground, and a signal. For the voltage supply we have determined that it will need 3.3 Volts to give the actual sensor the power needed to be turned on. The signal wire must be connected to the microcontroller via a digital input pin. The last wire will be connected to a ground pin. A 4.7 kΩ resistor will be added between the signal and voltage supply wires for lowering the current provided.

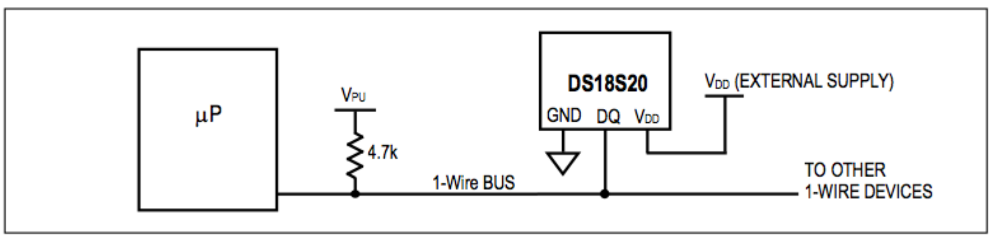


Figure 5.6.a. Schematic of Temperature Sensor Connected to Microcontroller (Reprinted with Permission from OddWire.com)

Once we have the sensors configured to the microcontroller from a bread board, we can check to make sure the component is working properly, which will be discussed later in the testing section of this paper.

Adding water temperature sensors to our data buoy seemed to be a no brainer. This feature can be very useful for anyone trying to check on air and sea conditions. A person will be able to decide if they need to wear a jacket because the air temperature is cold or a tank top because it is hot. Also, they will be able to determine if the water temperature will be too cold for activities such as swimming or if they should bring a bathing suit because it will be perfect for taking a dip. This brings us to our first prototyping decision; where should the temperature sensors be located on the buoy?

We have decided that the buoy housing will have the air temperature sensor fixed to the top side and the water temperature will be located on the bottom. Being that these sensors come in a waterproof form, they will be easy to integrate into our design. A picture of the component itself is shown in the following figure:

Figure 5.6.b. DS18B20 with Waterproof Probe and Cable (Reprinted with Permission from OddWire.com)

To fix the temperature sensor to the buoy housing, we will drill a specific size hole where the metal casing will protrude from the buoy. This will then be sealed for keeping the housing watertight and can be done with a sealant such as silicon or marine cocking. Having the metal tip exposed to the air and water should give us accurate readings of the area needing sampled.

5.7 Wind Speed Sensing

For the wind speed sensing feature of our data buoy, we have decided to use the Adafruit Analog Wind Speed Sensor or Anemometer. When prototyping the wind speed sensor, we will discuss what the reasons are for adding this component to our data buoy system and where it can be mounted for the best accurate results. We will mention how the Adafruit Wind Speed Sensor is implemented into the circuit with the microcontroller as well.

When deciding what our data buoy system needed in terms of features or sensors, we took into consideration air as well as water information. Since we added the air temperature sensor we decided it would be most suitable to add the wind speed sensor to our system as well. This feature is very practical on data buoys that are being used today because it is a key component to weather and water conditions when boating or any other water sports. Knowing the wind speed can help with determining what the weather and water conditions are going to be like at that certain time or day. If the wind is really blowing, you can safely assume that the wave height will be greater than a day when the wind is not blowing at all.

Another advantage for the wind speed sensor in our data buoy system is that it will be monitoring as well as recording the outside wind data. This means that if someone wants to try and determine what the upcoming or near future winds will be like, they can take the data that has been recorded from the past and make an educational guess on future winds. On a typical day of fishing for myself, the first thing I check before going is the wind speed so that I can determine if it will be adequate enough for being in a boat. If the wind is too strong, I will know to take caution and possibly plan another day to make a trip.

When determining where the wind speed sensor will be mounted, we first take a look at what the wind speed sensor looks like and what the actual dimensions are of the component. The unit itself is made of mostly plastic with some metal for connecting wires and a circuit board inside for the analog data measurements. The sensor has a base that is made to be mounted with four screws. The top portion that spins has three arms that stick out with cups on the ends for the best air resistance for the spinning mechanism. From the base a four pin connector is mounted where the external wiring hooks to send and receive the power and data. From the image below we can clearly see the configuration of the Adafruit Wind Speed Sensor.

Figure 5.7.a. Adafruit Wind Speed Sensor Configuration (Reprinted from Adafruit.com)

Now that we know the basic physical setup we can pick the best and most accurate place to mount the wind speed sensor to the buoy housing. Since the wind speed sensor needs room for the arms to spin with the wind, it will need a sufficient amount of space. The top of the buoy will be the best suited area for the sensor to be fixed being that there will be nothing protruding to block the arms from spinning freely as needed. We will have to determine where the solar panel(s) will be mounted as well as the air temperature sensor so that nothing will affect one another in any way.

Now that we know what the sensor looks like and where it will be mounted to the housing of the data buoy system, we can start to integrate it with the microcontroller and implement the internal portion of the prototyping of the wind speed sensor. Form the microcontroller, an input voltage of 7-24 Volts will be supplied to the wind speed sensor. From the wind speed sensor, a signal or data wire will transmit back an analog signal of 0.4-2.0 Volts back to the analog input of the microcontroller where it will make the simple arithmetic conversion from voltage to degrees in Celsius or Fahrenheit. The sensor will be grounded as well. The analog voltage signal is varied as the cups on the sensor turn, thus giving the measurement that is then translated to speed.

5.8 Buoy Housing Assembly

The Buoy Housing will consist of the basic body and bolts to hold components in place, and washers as well as waterproof treatments at each hole to make sure there is no leakage into the housing. The Buoy Housing will be painted on the outside to reduce the internal temperature of the device in direct sunlight. If necessary additions can be made to the housing to increase weight at the bottom to prevent capsizing, and floatation assistance can be added around the outside.

5.9 Wireless Interfacing

Now that we have figured out how all the prototyping for the sensors and hardware will be implemented, we can look into the prototyping for the programming of the wireless interfacing. We will be using the wireless interfacing so that the system allows a user to request information from the data buoy from a computer at a distance. In order to do this, we must have the component connected to the computer through wireless communication. To begin the prototyping, we must take the wireless module and make sure it will power on. The device will need certain commands sent to it to set up some general settings. These settings also determine if the computer is detecting the module. The main setting for the module are, Baud Rate, Device Information, and Device Version.

Once we figure out how we are going to prototype the device, we can then determine where the component itself is going to be situated in the overall buoy system. We have determined that the best spot for the wireless component be inside the buoy housing keeping it away from water or harsh weather conditions. It will be connected directly to the PCB making it need to be in close range as well. We will make sure that there will be no water able to get to the device itself being it is not waterproof and any water that would reach it would damage the component and potentially the whole system.

6.0 Testing

Now that we have determined how the prototyping of the sensors will be accomplished, we will look into testing each component and determining if the outputs are valuable and sufficient for our data buoy system. In the earlier portions of this paper we determined that we found some sensors and components that are best options for the features that will be integrated and implemented into our data buoy system. For testing the sensors, we will be connecting each individual component separately to an Arduino UNO being that it uses the Atmel AtMega328p microcontroller which is the same as the one we chose. Once we have the microcontroller programmed for the certain sensor, we can check that it is giving an output reading on the Arduino Serial Monitor.

After the microcontroller is programmed and the circuit for the sensor is designed and connected, we can test with instruments that are made for the each given feature. After testing each sensor and component we can then put all the code together and integrate all the circuits and sensors into a single breadboard to make sure everything will work as one system. Now that everything is integrated into one temporary system for testing purposes we can make put together the wireless connection system so that we can make sure the microcontroller will communicate with the computer and translate all the data that has been tested and verified from the previous sections of this topic.

Last, after everything has been tested electronically and found to be working properly, we will begin to put together the physical system and mount the components and sensors to the housing. This will temporarily complete the fully functioning device and data buoy system we are looking to accomplish until our Printed Circuit Board is finished being developed and built. Each component and sensor will be added to the housing one at a time, making sure every part added is secured as well as waterproofed adequately.

6.1 Testing Prototypes

During this section of documentation, we will take each component and sensor and test them to make sure they are working as expected and intended. We will build the circuits for each component on a bread board individually and program an Arduino UNO to test the sensor. We chose to use an Arduino UNO because it uses the Atmel atMega328p which is the same microcontroller we will be using in the design of our data buoy system. After testing the circuit and code, we expect to get accurate results as if the data buoy is being used. Once we determine each component is working properly, we will integrate all the sensors into a single system and test again.

Once the sensors have been tested and integrate we will build the physical prototype and make sure each component is mounted and waterproofed into the buoy housing accordingly. The buoy should be complete with all components and sensors once this has been finished. If at any time we determine a component or sensor is not working as intended or has a manufacturer error, we will quickly order a new replacement part or entirely different component based on the determination. We cannot afford to waste any time with faulting parts and or inaccurate readings. The prototyping will be completed once we have all the components mounted and integrated into one system and determine the data buoy is working as intended.

6.1.1 Power Testing

When it comes to testing the power supply, two different components of our project have to be tested. First of all, the chosen battery has to be tested because without a battery supply, our project will not be able to function at all. In order to test our battery, we will be using a digital multi-meter to measure the maximum output voltage and current provided as well the absolute minimum values at which this battery can function. We will be charging and discharging the battery in order to be able to obtain measurements such as charge time and self-discharge. Also, it is important to measure charge and discharge temperatures for the battery given that we have to consider the functionality of all of our components at high temperatures. After our testing has been done, we will be comparing our obtained measurements to the battery ratings in order to verify them.

The second component that has to be tested is the solar panel. Given that our solar panel will be responsible for capturing solar energy in order to charge our battery, some standard tests have to be made. In the following section we will be explaining how our solar panel will be charging our battery.

6.1.2 Charging

Charging our lithium polymer battery can be a tricky task. We have to be careful as to not overcharge our battery because these types of batteries do not have a high overcharge tolerance. Also, we must watch out for the charging rate and especially the charging temperature of the battery. To do this kind of testing, we will be connecting our solar panel and our battery to different ports of a solar lithium polymer charger. This is a very simple way of testing our charging method. It is done by simply connecting the lithium polymer battery to the BATT port on the solar lithium polymer charger by using a 2-pin JST cable and the solar panel into the DC jack of the solar lithium polymer charger. Once this is done, we will be exposing the solar panel to the sun by placing it outdoors on a sunny day. However, me have to be careful as to not expose our battery to the sun because this can cause overheating of the battery and can affect its performance. We will make sure that our battery is shaded and protected from the sun. The solar lithium polymer charger is equipped with three color indicator LEDs. One LED indicates that the power connection has been established successfully, another LED indicates that the battery is currently being charged, and the third LED indicates when the battery has reached its maximum capacity. Once this process has been finished, we will be testing the battery as we have explained before in order to compare it with its absolute maximum cell voltage rating. By doing so, we will be able to verify the functionality of our solar panel and the solar lithium polymer charger.

6.1.3 Microcontroller Testing

In order to test our chosen microcontroller, Atmel’s ATmega328P, we have decided to use an Arduino UNO board. The Arduino UNO board is equipped with our chosen microcontroller and because of that it is perfect for us to use to for testing purposes. To verify that our microcontroller is capable of being the “brain” of our design, we will use it to test our chosen sensors.

In the following sections, we will be demonstrating how testing was done in chosen sensors such as the LIS3DH used to record acceleration readings that will be then used to calculate our wave height. Also, the Maxim DS18B20 Digital Temperature Sensor which will be used to record valuable data on water and air temperature. Finally, the Adafruit Analog Wind Speed Sensor which is used to measure the wind speed. All testing procedures done on these chosen sensors were done using an Arduino UNO board which has an embedded ATmega328P microcontroller chip. Figure 6.1.3.a is an image of the Arduino UNO board used for testing purposes. Figure 6.1.3.b is an image of the actual microprocessor that is embedded on to the Arduino UNO board.

Figure 6.1.3.a: Arduino UNO board

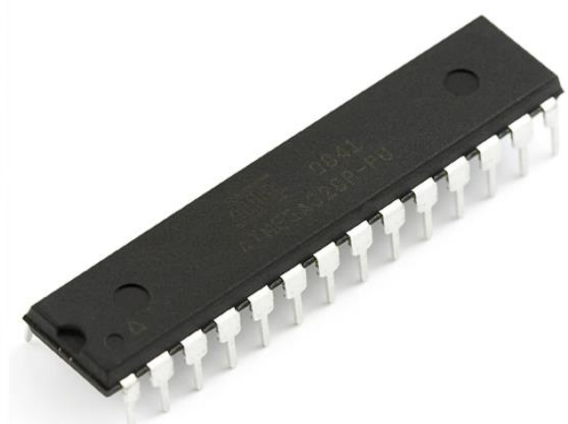


Figure 6.1.3.b: Atmega328P microcontroller

6.1.4 Salinity Detection

The components for the detection must be tested individually before the sensor can be put together and tested for its intended use.

The Laser Diode will be emitted onto a photodetector, at the distance expected for the sensor, and the output power will be checked. Then the same test will be done through water to make sure enough light makes it through the absorption of the water.

The Quad Cell sensor will be tested with the Laser diode once emitted power is documented to test for expected output voltages, as well as the change in voltages for each quadrant given certain beam movement.

The first step in testing for the salinity detection will be to align the beam onto the sensor at the proper angle into the sample to make sure the beam divergence of the salinity change will not cause the beam to deflect off of the sensor. Once the beam is aligned the next step is to establish a baseline for the water with no salt. This experiment will have to be repeated several times to make sure that the same measurements are obtained repeatedly before any salt levels can be detected and to determine any possible variations in data at the start.

The next step is to set a salt level to test for in each sample which will have to be carefully controlled. A table will be created and added to show the amount of water used and the corresponding salt added for each data set obtained to establish our baselines for the device in use.

These tests will be repeatedly several times to determine the consistency of the equipment and determine if changes in environment affect the results as well.

6.1.5 Wave Height Sensor Testing

For the wave height sensor features of this design, we will be testing two accelerometers by placing each individually on a breadboard and wiring its pins to an Arduino UNO. The two accelerometers we chose were the LIS3DH and the ADXL335. These two accelerometers are very similar in features such as number of axes, dimensions, weight, operating voltage range, and operating temperature range. The only two differences between these two are the scaling range and the output type. While the LIS3DH has a selectable range of ±2g, ±4g, ±8g, or ±16g, the ADXL335 has a fixed range of ±3g. The other difference between these two accelerometers is the output type. The LIS3DH accelerometer has a digital output and the ADXL335 has an analog output. Given that we are expecting to require a range between ±2g and ±4g, both accelerometers are equipped to with the necessary range to get the job done. Further testing will be done in order to determine which output type will be best for our purposes.

We will begin testing the LIS3DH accelerometer. The first step is to have the header strip cut into two pieces, one that has three pins and one that has eight pins. The strip with 3 pins is used for the ADC pins and the strip with 8 pins is used for the power and I2C/SPI interface pins. Once this is done, the accelerometer board is placed on top of these strips and soldered were needed.

Now that the pins have been soldered, we can begin wiring the LIS3DH accelerometer board with the Arduino UNO. As shown in Figure 6.1.5.a, the following pins are connected if I2C interface is used:

- Vin pin on accelerometer board connected to the 5V pin on Arduino UNO
- GND pin on accelerometer board connected to GND pin on Arduino UNO
- SCL pin on accelerometer board connected to I2C clock SCL pin on Arduino UNO
- SDA pin on accelerometer board connected to I2C data SDA pin on Arduino UNO

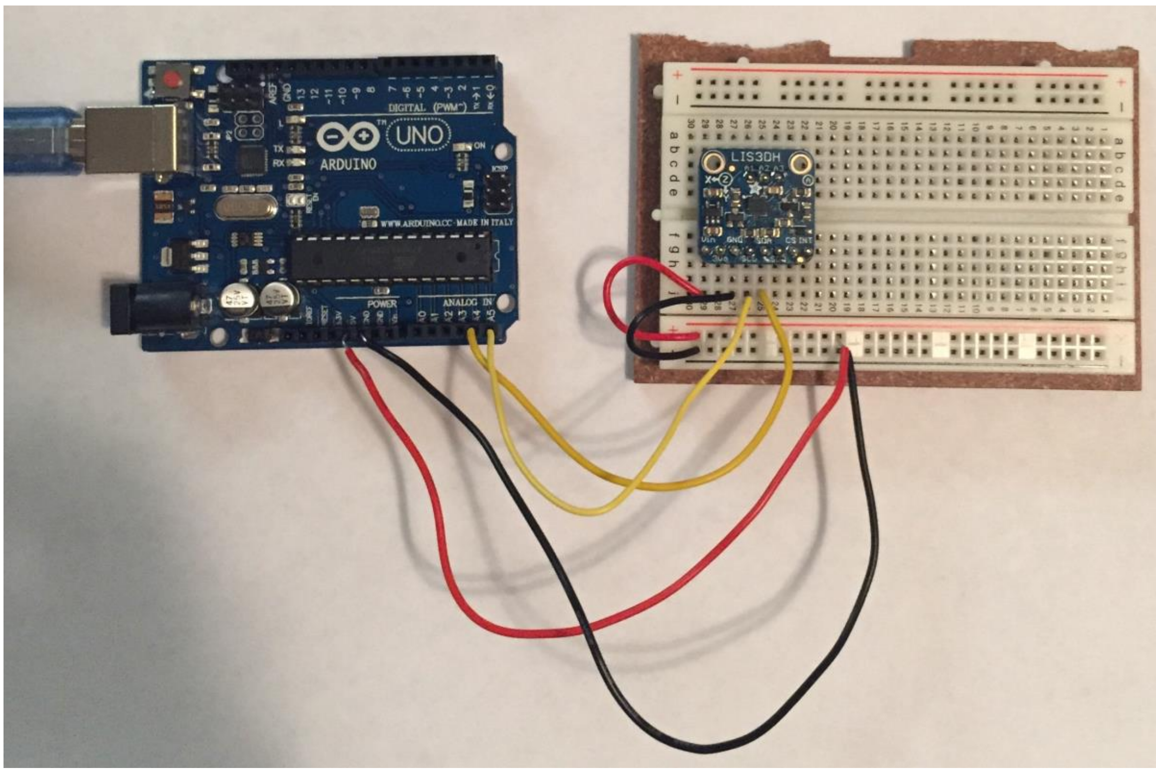


Figure 6.1.5.a: LIS3DH accelerometer board connected to Arduino UNO using I2C interface

As shown in Figure 6.1.5.b, the following pins are connected if SPI interface is used:

- Vin pin on accelerometer board connected to the 5V pin on Arduino UNO
- GND pin on accelerometer board connected to GND pin on Arduino UNO
- SCL pin on accelerometer board connected to Digital #13 pin on Arduino UNO
- SDO pin on accelerometer board connected to Digital #12 pin on Arduino UNO
- SDA pin on accelerometer board connected to Digital #11 pin on Arduino UNO
- CS pin on accelerometer board connected to Digital #10 pin on Arduino UNO

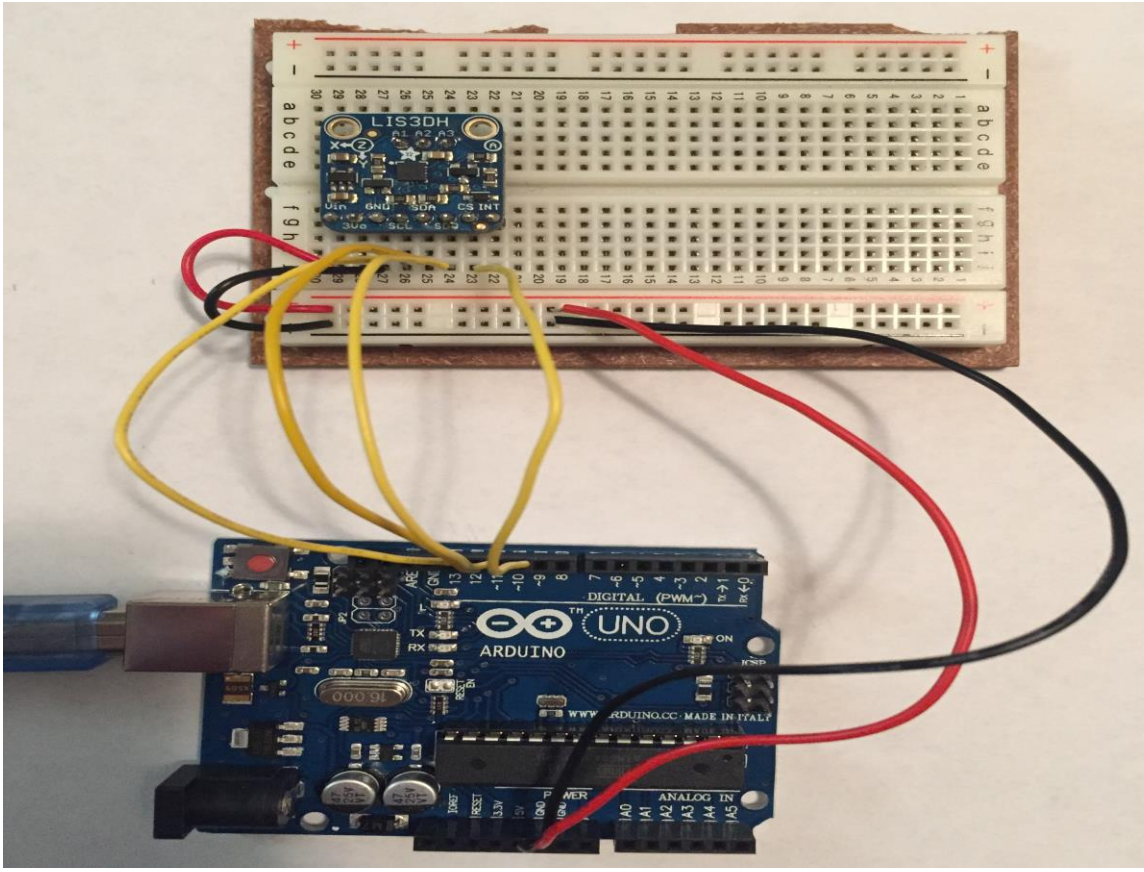


Figure 6.1.5.b: LIS3DH accelerometer board connected to Arduino UNO using SPI interface

Now that we know how the LIS3DH accelerometer board is wired, we begin reading data from the sensor. First, we need to download the Adafruit LIS3DH library and Adafruit Sensor library and add them to our libraries folder. Figure 6.1.5.c shows the code that was used to obtain acceleration recordings using SPI interface and Figure 6.1.5.d shows the acceleration recordings obtained. Notice that we have selected to obtain the acceleration readings from the z-axis only. These recordings were done when the accelerometer was lying flat on a table. As observed in Figure 6.1.5.d, the acceleration recordings for the z-axis are between the values of 9.6 and 9.9 m/s^2. Because the accelerometer is sensitive to gravitational force, which is approximately 9.8 m/s^2, the acceleration readings vary between these numbers. Given this, our accelerometer readings are correct. Also, notice that a range of ±4g was used. However, the range can be easily changed by altering a single line of code.

```
#include <Wire.h>
#include <SPI.h>
#include <Adafruit_LIS3DH.h>
#include <Adafruit_Sensor.h>

#define LIS3DH_CS 10

Adafruit_LIS3DH lis = Adafruit_LIS3DH();

#if defined(ARDUINO_ARCH_SAMD)
   #define Serial SerialUSB
#endif

void setup(void) {
#ifndef ESP8266
  while (!Serial);
#endif

  Serial.begin(9600);
  Serial.println("LIS3DH test!");

  if (! lis.begin(0x18)) {
    Serial.println("Couldnt start");
    while (1);
  }
  Serial.println("LIS3DH found!");

  lis.setRange(LIS3DH_RANGE_4_G);

  Serial.print("Range = "); Serial.print(2 << lis.getRange());
  Serial.println("G");
}

void loop() {
  lis.read();
  Serial.print("  \tZ:  "); Serial.print(lis.z);

  sensors_event_t event;
  lis.getEvent(&event);

  Serial.print(" \tZ: "); Serial.print(event.acceleration.z);
  Serial.println(" m/s^2 ");

  Serial.println();

  delay(2000);
}
```

Figure 6.1.5.c: Code used to test LIS3DH accelerometer board using I2C interface

```
LIS3DH test!
LIS3DH found!
Range = 4G
          Z:  12192          Z: 12.13 m/s^2

          Z:  8160           Z: 9.73 m/s^2

          Z:  8160           Z: 9.67 m/s^2

          Z:  8160           Z: 9.69 m/s^2

          Z:  8192           Z: 9.71 m/s^2

          Z:  8336           Z: 9.98 m/s^2

          Z:  8096           Z: 9.69 m/s^2

          Z:  8128           Z: 9.73 m/s^2

          Z:  8080           Z: 9.85 m/s^2

          Z:  8160           Z: 9.66 m/s^2

          Z:  8160           Z: 9.90 m/s^2
```

Figure 6.1.5.d: LIS3DH test acceleration recordings

6.1.6 Water/Air Temperature Testing

For the Water and Air temperature sensor features of this design, we will be able to code, build, and test fairly simply with just a few extra instruments and components. The water and air temperature sensors we chose are the Maxim DS18B20 Digital Temperature. They will be two separate sensors monitoring and recording data for the temperature in surrounding water and local air. For this we will need to test the sensors in water and out of water to make sure they are reading properly and accurately as intended. We will then compare with instruments made for measuring these same features. Once determined successful, we will look into areas on the buoy housing that will hold the sensors best for accurate readings. If in any case one of the temperature sensors is not working properly, we will replace with one of the extra three spares ordered.

To begin our testing of the temperature sensors, we will start with finding some applicable code that can be found through many open source sites on the internet. The code consists of using two specific libraries, Onewire and DallasTemperature, made for the sensor and their functions as well as some loops for reading bits and making conversions. The specific libraries used are explained previously during section 4.7. Once the code has been compiled correctly we can then compare the data being recorded to that of the actual temperature instrument used for testing.

Beginning with air temperature, we will use an instrument that can accurately measure the air temperature so that we can compare the two readings and determine if our component is working properly. We begin by finding a thermostat in a room where the inside temperature is clearly visible. We then use our DS18B20 digital temperature sensor to record the temperature in the same room. The following figures show the circuit of the DS18B20 connected to the Arduino UNO:

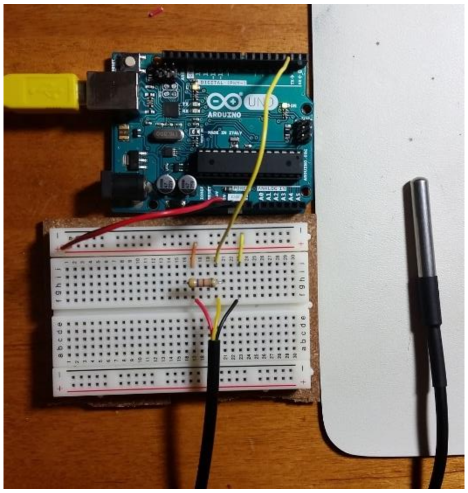
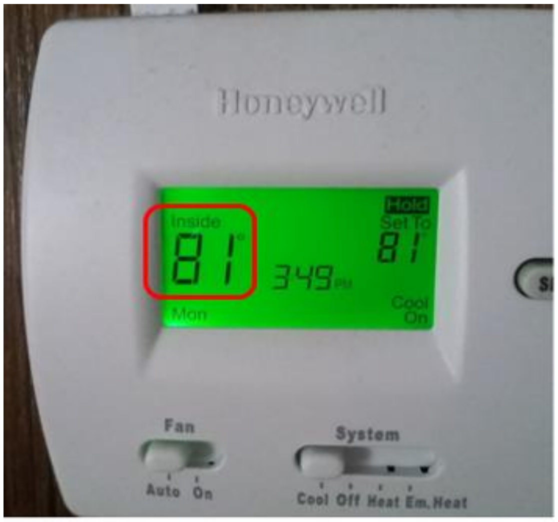


Figure 6.1.6.a. Circuit of DS18B20 Connected to Arduino UNO with Breadboard

Next we compare the readings between the thermostat and our temperature sensor. The figures below show their measured outputs.



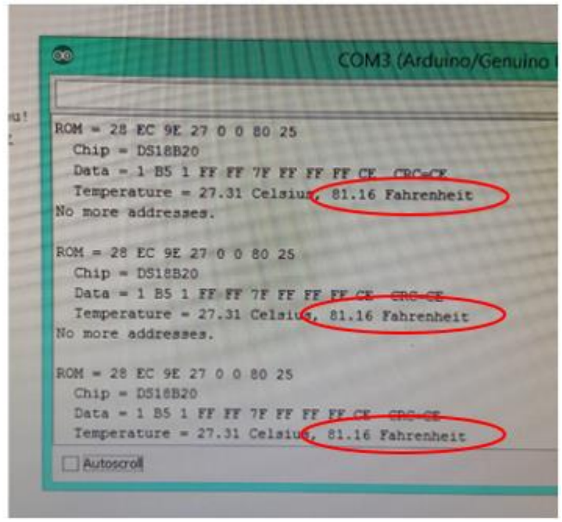


Figure 6.1.6. b. Thermostat Reading Inside Room c. Three DS18B20 Readings Inside the Same Room

This photo shows that the reading of the DS18B20 Digital Temperature Sensor was very close to that of the thermostat. We measured the output three times and got the same measurement each time. We have determined that this sensor will be suitable for the needs of the air temperature feature of the data buoy system. Next we look into testing the water temperature feature and determine if the same sensor will be of same value.

For the water temperature feature we did similar testing. We began by taking a cup of water and microwaving it for a short period of time. Next we used a digital thermometer to record the waters temperature. This is testing the DS18B20's waterproof ability as well. We recorded the data. After we stuck the DS18B20 into the same cup of water only moments later and recorded the data found there as well. The Following figure shows the thermometers reading and the DS18B20 measuring the water temperature:

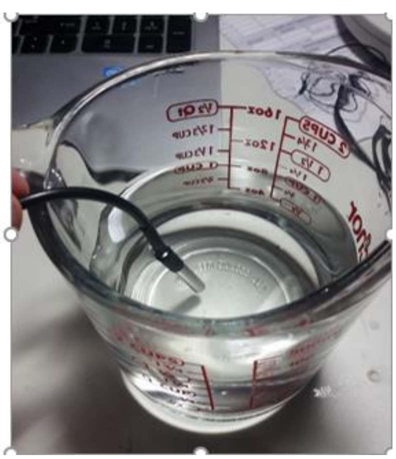
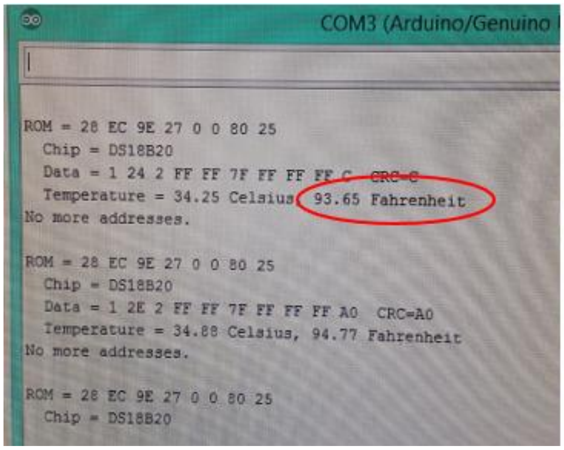


Figure 6.1.6.d. Thermometer Temperature Reading, e. DS18B20 Measuring Water Temperature and Testing Waterproof Design

The DS18B20 digital temperature measurements was then taken and a few samples of data were recorded. The following figures show the temperatures recorded as well as the times they were sampled:



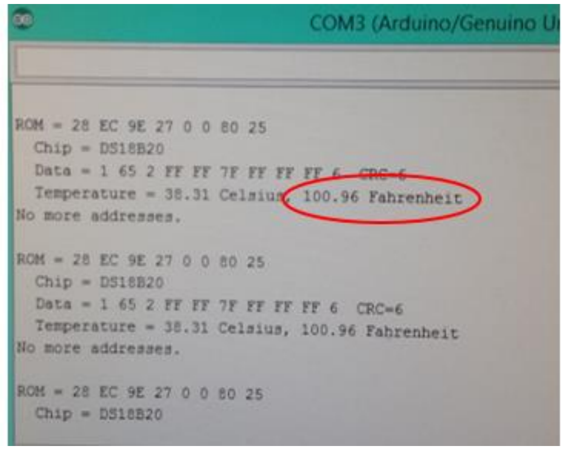


Figure 6.1.6.f. DS18B20 Temperature after 10 Seconds g. After Settling at Around 1 Minute

From the results found using the thermometer and the DS18B20 to measure the water temperature, we can see there is some error between the two. The reading for the thermometer is around 102.1°F and the DS18B20 after settling is 100.96°F. Comparing these two results we have determined that the DS18B20 is suitable for the water temperature feature and will work accurately. The error found seems to be relatively small, around 1.1°F, and was probably a cause of the water cooling between measurements. The waterproof casing for the DS18B20 seemed to work as intended as well.

6.1.7. Wind Speed Testing

For the wind speed feature of this systems design, we will test and make sure the sensor we have chosen is working properly before we move on to putting everything together. We have determined we are going to use the Adafruit Wind Speed Sensor also called an Anemometer. This particular sensor measures wind speed with an analog output voltage. The voltage range is around 0.4-2.0 Volts and is directly proportional to the wind speeds of 0-32.4 m/s in which can be converted to mph. This conversion equation was shown previously in section 4.6. For the testing instrument we used a handheld Anemometer with digital output reading that worked appropriately. The anemometer can be seen in the following figures during the testing of this component.

To begin testing the Adafruit Anemometer we found some open source tutorials that show some circuit designs for our particular sensor. These open sources provided the code needed for using the Arduino UNO Atmel atMega328p, in which we used the same microcontroller in our design as well. The code for the Adafruit Air Speed sensor needed the Onewire library for implementation just as the air and water temperature sensors did. The circuit for the sensor only uses three wires: one for voltage supply of 7-24 Volts, one for signal/data transfer with analog output reading of 0.4-2.0 Volts, and one for connecting to ground. The circuit can be seen in the following figure.

Brown Wire: Voltage Supply (9V)	Blue Wire: Signal/Data	Black Wire: Ground

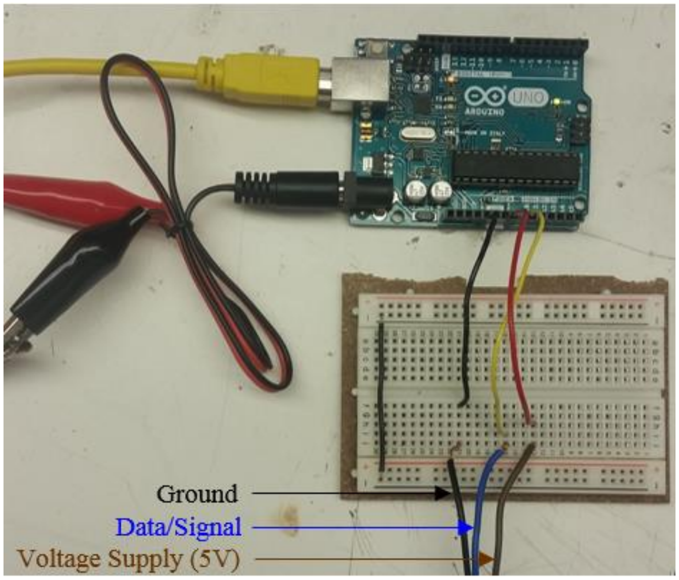


Figure 6.1.7.a. Adafruit Wind Speed Sensor Circuit Using Arduino UNO and Breadboard

From the image we can see the three wires connected to the breadboard and Arduino UNO. Since the Adafruit Wind Speed sensor needs an input of at least 7 Volts and the Arduino only has output supply voltages between 3-5 Volts, we added an external source of 9 Volts to the circuit. Once the circuit was designed and completed, we tested the code to see if there were any errors. The code compiled as intended and we were ready to test our wind speed sensor and compare it with the Anemometer purchased. The following figure shows the wind speed output readings from the Adafruit Wind Speed Sensor as well as the output readings from Anemometer used for testing.

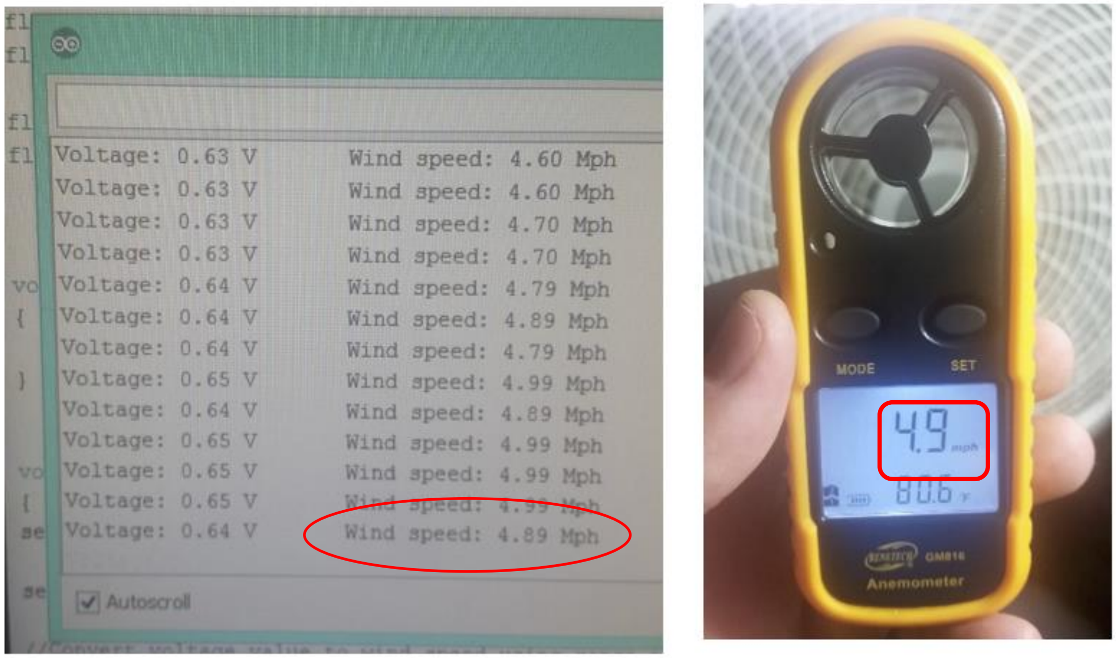


Figure 6.1.7.b. Adafruit Wind Speed Sensor Output, c. Compared Anemometer Output

We measured the wind speed created from a fan, for 5 random samples, with each instrument and compared our results. We took five random reading so that we could effectively compare the output readings in case there were any outliers found. The readings for each instrument can be seen in the table below.

Table 6.1.7.a. Compared Data for 5 Random Samples of Wind Speed

Random Testing #	Adafruit Wind Speed Sensor	Anemometer for Testing
1	1.78 mph	1.9 mph
2	3.26 mph	3.4 mph
3	2.34 mph	2.7 mph
4	4.89 mph	4.9 mph
5	5.67 mph	6.8 mph

After testing the Adafruit Wind Speed Sensor to the Digital Anemometer we have found the results to be similar but not exact. Finding a way to test the wind speed sensor exactly is relatively tough. Being that the sensor is analog and the it is not a constant input, we had to take random samples to make sure that the sensor was reading similar results. For the case of our data buoy system, we will be calculating averages over short periods of time instead of instantaneous readings. This will make for better data recorded being that there are always wind gusts and dead wind periods that will occur.

6.1.8 Wireless Communication Testing

For the testing of our wireless component we will be determining if it is going to work as intended as well as if the component chosen is suitable for our data buoy system. We will be connecting an Arduino UNO to a computer wirelessly via the HC-06 and make sure that is responding to the computers commands as it should. Throughout this section we will go step-by-step into the wireless testing and compare the conclusion with our testing results. To begin, we will connect the HC-06 to the Arduino UNO on a bread board from a circuit found in an open source online site. We will supply the correct voltage that is stated in the section 4.9.2. From here we will determine if the HC-06 is powering up as it should.

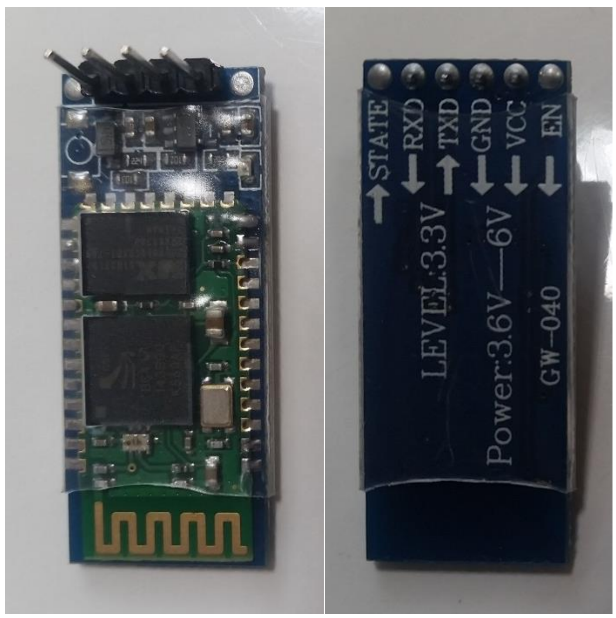


Figure 6.1.8.a. HC-06 Front and Back View Showing Pinout Configuration

We were able to connect to the HC-06 and detect it through the computer. We set the AT commands checking the version and baud rate settings. Next we will check to see if the computer is able to receive text data wirelessly from the Arduino through the HC-06. We use some open source code found online. The code lets you type a number into a Tera Term which is a terminal software. The number that is put into Tera Term will be printed in the serial monitor in the Arduino IDE. If the number is printed correctly this means the HC-06 is working as it should.

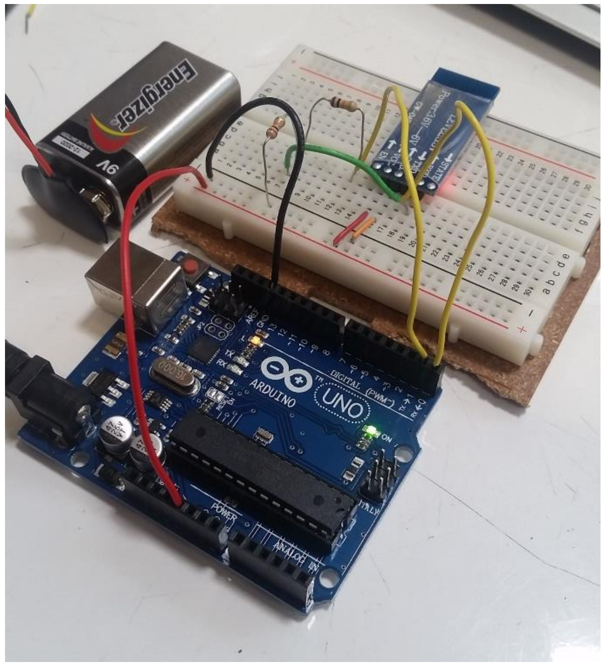


Figure 6.1.8.b. HC-06 Connected to Arduino for Testing

In Figure 6.1.8.b, we can see the HC-06 connected to the Arduino UNO through a breadboard. The figure shows how the pins from the HC-06 are connected to the pins on the Arduino. The Arduino was powered by a 9 V battery wirelessly connected to the computer with Bluetooth signal. After connecting we did multiple tests to show that the HC-06 was working properly and found success. We have determined that the HC-06 will in fact be a suitable wireless connection device for our data buoy.

6.1.9 Buoy Housing Testing

The buoy housing will have to be tested for a number of factors before the prototype is constructed to make sure the housing can hold up to open environment conditions such as the ocean and sunlight.

The first factor to consider is buoyancy of the housing as the housing is not the only weight to be considered and once the components are added it still has to stay afloat. This testing will determine if more floatation components need to be added or if the housing needs to be redesigned completely. The test for this will be weighing the device and see how far it sinks into the water. Next we would add set weight to see how much further the device sinks to determine if the additional components weight would overwhelm the housing.

The next factor is heat resistance of the housing against warping as the device will most likely be in direct sunlight throughout the day if it is in use on a dock or in the ocean. Any warping of the device could cause damage to the components attached as well as affecting the integrity and causing the device to shrink. Leaving the housing out in direct sunlight for several days would allow us to determine any possible problems in this area.

Internal Temperature is a definite factor to keep in mind for such a device that will be in direct sunlight. While the device will be in water to allow for heat to be dispersed, if the device’s internal temperature reaches too high the components inside could melt or fail. The device will be placed in water out in the day time with an internal thermometer as well as possible extra electronic pieces to see how they hold up inside during the test. Any damage to the electronics could call for a change in the design.

This device is intended to be placed in water and left there for use. As such the device should be resistant to the corrosive effects of the salt in the water as well as other possible chemical interactions that could occur in the open environment. The best way to test for this is to simply leave the buoy in the ocean for a set period and visually inspect the device for any signs of wear or damage.

Waterproofing the housing is very important for a device intended to go anywhere near the ocean as the waves will cause it to be submerged. This device will be sealed and repeatedly submerged for an extended period, then observed to see if any bubbles rise from the device indicating a leak.

6.1.10 Prototype Waterproof Testing

The prototype will have to be proven to be waterproof when it is all put together. While the housing has been tested prior to this test, all of the components being set in the device could change the results. First the device will be placed in the water to test buoyancy and make sure the device is not slowly sinking. Next the device will have water sprayed over the top to check for leaks in the solar panels or wiring into the device. Finally, the device will be submerged and the output will be checked after each submersion that all components and the housing have maintained their integrity.

6.2. Materials Needed

Air and Water Temperature Sensor Testing

- Maxim DS18B20 Digital Temperature with Waterproof Design
- Arduino UNO
- Computer
- Digital Thermostat (Air Test Only)
- Digital Thermometer (Water Test Only)
- Cup of Water (Water Test Only)
- Jumper Wires
- Breadboard
-

Wind Speed Sensor Testing

- Adafruit Analog Wind Speed Sensor (Anemometer)
- Arduino UNO
- Computer
- Woopower Digital Anemometer GM816
- 9 Volt DC Power Supply
- Fan
- Jumper Wires
- Breadboard

Wave Height Sensor Testing

- Adafruit LIS3DH Accelerometer
- Adafruit ADXL335 Accelerometer
- Arduino UNO
- Computer
- 5V DC Power Supply
- Jumper Wires
- Breadboard

Wireless Communication Testing

- HC-06 Wireless Bluetooth Serial RF Transceiver
- Arduino UNO
- Arduino IDE Serial Monitor
- Tera Term
- Breadboard
- 9V DC Power Supply
- Computer

Salinity Sensor Testing

- Listed Components
- Tub of Pure Water
- Table Salt
- Measuring cup
- Breadboard
- Voltmeter
- Power supply
- Protractor
- Ocean Water

6.3 Test Results

After testing the Maxim DS18B20 Digital Temperature Sensor for use as our air and water temperature features on our buoy, we have determined that they are very well suited for the job. They both had accurate and similar readings compared to the instruments we used to test them against. When measuring the air, we found exact results which was expected since the sensor was sitting in the room for hours before taking a reading. This means that it had been able to settle on a certain temperature for a while which made the output readings very accurate.

When we tested the water temperature we found a very similar result, though it was off by a small percentage. The water tested was able to cool off since the surrounding room temperature was less the sampling water temperature. Being that our water temperature sensor was used to measure a decent amount of time after the thermometer, it had more than enough time to drop 1°F. With this noted, we still believe that the measurements of both the air and water temperatures were accurate and precise enough for the feature we are looking to have in our data buoy system. We can conclude from these tests that the DS18B20 will be a valuable addition to the system.

During the water temperature testing portion, we have also concluded that the DS18B20 is waterproof as needed and that it should work as intended on the bottom half of the buoy housing where it will be mounted.

From the wind speed testing we gathered enough data to show that the Adafruit Analog Wind Speed Sensor can measure wind speed as accurately as we are needing to achieve. From Table 6.1.7.a, we sampled many different speeds randomly while measuring with both devices at the same time, to make the measurements more precise, neglecting any error that may occur. We have concluded that after testing the Adafruit Analog Wind Speed Sensor, it will be another valuable addition to our data buoy system. As for being water resistant, I believe it will work as intended on the top portion of the buoy housing where only small amounts of water will be splashed.

7.0 Final Design

Now that we have done extensive research on many different components for our data buoy features, we can put together the final design of the system. Since we chose a specific component for each feature, we will be able to go through all the parts and pieces that we have gathered. During this section we will also show the schematic of the overall design and how each of the components will be implemented and integrated to make a single working unit. While connecting all the components, we will describe the final functionality and what the intentions of use are. After the final design is completed we will go through multiple times to make sure the buoy system is working properly and as intended.

7.1 Gathered Components

Temperature Sensors

- Maxim DS18B20 Digital Temperature Sensor (Air)
- Maxim DS18B20 Digital Temperature Sensor (Water)

From all the research, comparisons, and pros and cons found; we have decided on which temperature sensor we will be using for the data buoy system. The DS18B20 has been determined the most suitable temperature sensor that can be used in our project. The decision fell on the digital architecture and the option of getting it in water proof form. With its programmable digital architecture, it will handle measuring temperature with the needed amount of precision and accuracy. This sensor will be used for both, the water and air temperatures. Since the DS18B20 has the option to come in a water proof form, it will handle the wet environment of the data buoy system. With this sensor integrated into our system we will have the capabilities of water and air temperature sensing.

Wind Speed Sensor

- Adafruit Analog Wind Speed Sensor (Anemometer)

After looking deep into many wind speed sensors and doing extensive research, we have decided on the most suitable and accurate component for our data buoy system. We have determined that we will be implementing, as well as doing further research, on the Adafruit Analog Wind Speed Sensor. After comparing and contrasting with the other devices available, we believe this one is the best applicant for our system. With very many Pros and only one Con, we look forward to using this component in our data buoy. The previous two sections of research gave us all the information needed to make our decision.

Wave Height Sensor

- Adafruit LIS3DH Accelerometer

All four previously discussed accelerometers are very similar when it comes to power consumption, number of axes, supply voltage needed for operation, operating temperature range, weight, and size. However, the price does vary for each accelerometer option. Given that all the digital output accelerometers are very similar; we have chosen the LIS3DH because of its affordability. Also, we have decided to test the ADXL335 accelerometer because it is the only one that has an analog output and it has characteristics that are very similar to the LIS3DH. These two accelerometers will be tested and compared to see which one will benefit our project the most.

Salinity Detection

- Red Dot Laser Diode Module from lightinthebox.com
- Laser Diode Mirror from Edmund Optics (32x32mm)
- Quad-Cell Detector from Mouser Electronics
- Sensor Housing from The Container Store

The components listed were chosen from the options given in the research section. These parts were the cheapest possible components capable of obtaining the needed information for the salinity detection. The sensor will be tested once assembled to make sure no parts need to be replaced to obtain proper data.

Battery

- Lithium Ion Polymer battery

From the various battery technologies that exist, we have chosen to use a lithium polymer battery for many reasons. Lithium polymer batteries have the highest nominal cell voltage compared to other battery chemistries. These batteries come in small packets and are very lightweight, which is a plus because we do not want our data buoy to be heavy. Further, lithium polymer batteries have a high cycle life and an acceptable self-discharge. More importantly, batteries with this battery chemistry do not require maintenance.

Solar Panel

- Adafruit Solar Panel

From the four solar panel options discussed in previous sections, we have chosen Adafruit's solar panel. One of the main reason we have chosen this solar panel is for its reliability. This solar panel's characteristics fit our requirements perfectly. In addition, many tests have been done previously with this solar panel and lithium polymer batteries. Also, the solar lithium polymer charger works perfectly with this solar panel.

Housing

- Walmart 10 QT Drain Contain

We have chosen this container because it is very practical and fit our housing requirements. It is big enough to fit all other components on its interior and small enough to be easily transported. It has a cap on top which can be taken off if any components need to be replaced or added. The top of this buoy housing is capable of accommodating the solar panel easily. The material from which it is made of has good buoyancy.

Microcontroller

- Atmel ATmega328P

All three microcontrollers seem to fulfill our requirements for our design. Being that all three options are designed to operate at low power. These low power features allow us to get the highest battery life possible which is very useful. Price wise, all three microcontrollers are affordable and do not vary much so this should not be an issue. After analyzing all the key characteristics for each microcontroller and comparing them with each other, we have decided that the ATmega328P microcontroller might be the best suited for our design given that it fulfills all of our requirements and we have plenty of experience working with it on an Arduino UNO board. Most of our sensors are compatible with the Arduino UNO board, which is equipped with this microcontroller, and a wide variety of resources are available. This will help us a lot with the programming of the microcontroller to read data recordings from our different sensors implemented in the data buoy.

Wireless Communication

- HC-06 Wireless Bluetooth Serial RF Transceiver

From the Wireless Module Comparison section, we have come to a decision that we will be using Bluetooth to communicate with the user. Once we made that decision we also determined that the HC-06 Wireless Bluetooth Serial RF Transceiver was the best suited module for our data buoy system. Although the other two modules researched had slightly better range of use, there were many other features that made or decision more clearly. We found that price was more important of a factor than the range of use. Also, the HC-06 has a very simple connection and setup procedure making implementation quick and easy. The last reason we chose to use the HC-06 is because its size is very small and even with its own built in antennae.

7.2 Printed Circuit Board (PCB)

During this design course we are expected to build a self-designed Printed Circuit Board or PCB. The PCB is a board that hold the electrical parts and electronically connects the components using conductive tracks, pads, and other features etched from copper sheets on a non-conductive board. From the Final Design section of this documentation, we can follow the schematic in drawing the PCB design in the Eagle Cad Software. Once the PCB is designed and drawn, we can then have a company print our overall board. Once the board is printed and we have received it, we can solder all the components in their respective positions. In the following subsections we will discuss, explain, and evaluate our Printed Circuit Board design and drawing.

After researching what the requirements for our PCB are going to be, we have come up with a final layout. From combining all the component that will need to be mounted on the board to making sure the power supply values are regulated correctly, our schematic shows a simple yet working system. In the following section we will see a capture of the final schematic done in the EAGLE PCB tool software.

7.2.1 PCB Design

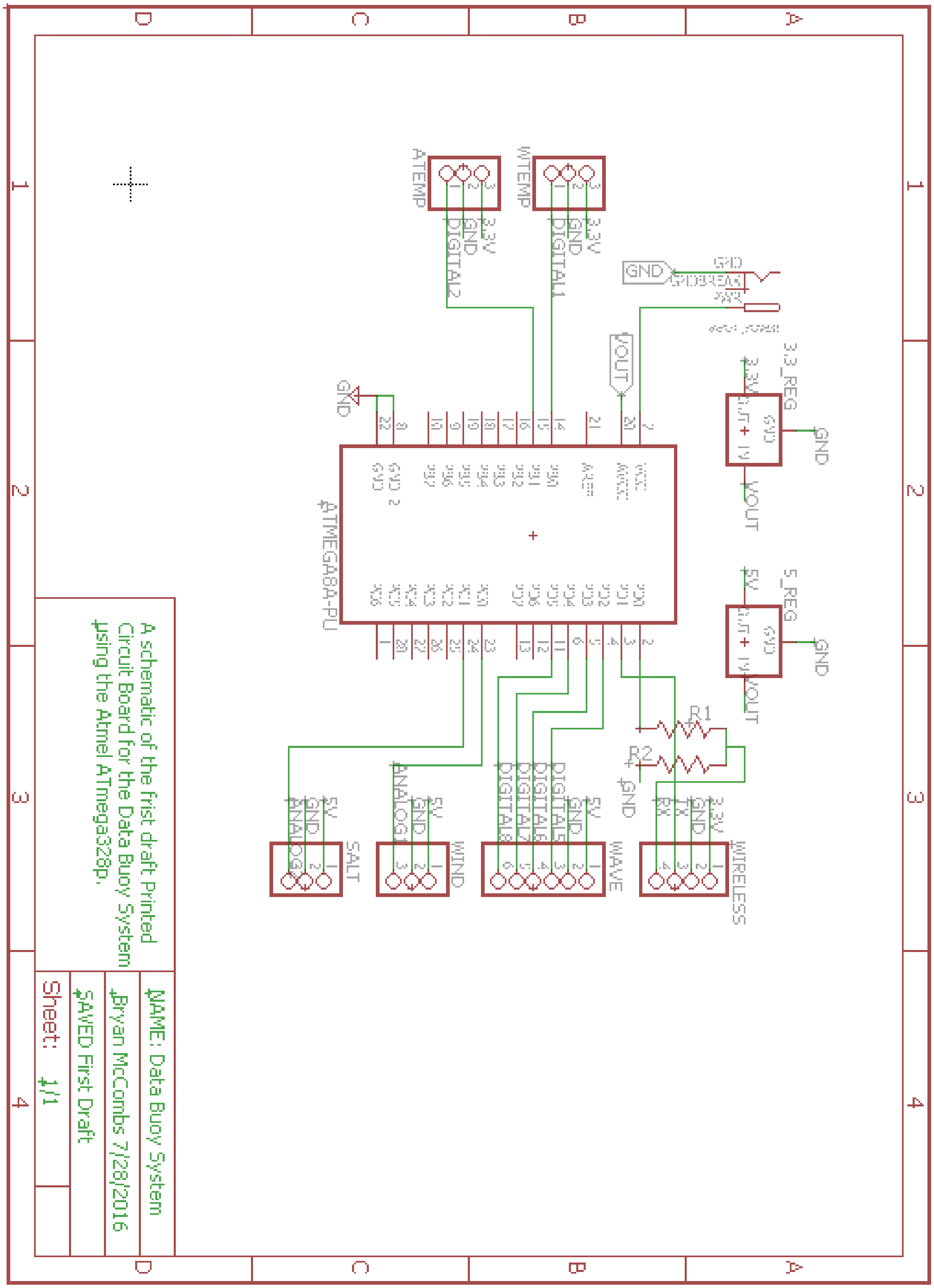


Figure 7.2.1.a EAGLE Data Buoy PCB Schematic

8.0 Administration

The purpose of the course Senior Design I is to prepare a final schematic and choose components to build a final working device in Senior Design II to present for graduation. The goal is to work with a team of other engineers to design and build something that applies to each team member's field of study and has real world application. The milestone chart in this section is designed to keep team members on track for the completion deadline. There are several major deadlines that must be completed as the semester goes on.

The first deadline is the testing of all the individual components separately, then with the microcontroller and PCB. Once all of the components test as working on the breadboard they will start to be placed into the device body until a working prototype is completed. The prototype must then be tested for demonstration and real world applications. The final step is simply to test the device repeatedly to ensure no problems occur with repeated testing to ensure a working device for the final presentation.

Each deadline on the milestone chart is set to have the device finished well ahead of the final deadline for when problems occur and a component has to be rebuilt or redesigned. This is to ensure on time delivery of the project and to allow for everyday life obstacles that could interfere with project completion.

If the project is completed ahead of schedule and working well there are other possible features that may be implemented for the device. The schedule is flexible enough that if it is decided that more needs to be added to the device, then the additional time needed is available.

8.1 Milestones

All project deadlines are subject to change due to issues and troubleshooting as required by the design team. Dates are set early to ensure on time completion of the project is still possible in the event of the arrival of major issues or component breakage during project completion and testing.

Summer 2016

Milestone	Estimated Completion Date
Project Idea Decision	6/3/16
Initial Theoretical Research	6/17/16
Parts and Components Research	7/1/16
Table of Contents Submission	7/1/16
Components Chosen	7/20/16
All Components Ordered	7/27/16
Report Documentation Half	7/6/16
Report Documentation Full	7/27/16
Components Checked	8/1/16

Fall 2016

Milestone	Estimated Completion Date
Initial Testing Stage – Buoy	9/9/16
Initial Testing Stage – PCB	9/16/16
Initial Testing Stage – Sensors	9/9/16
Initial Testing Stage – Power System	9/16/16
Design Finalization	10/7/16
Prototype Built	10/14/16
Complete Design Testing	10/28/16
Repeat Testing	11/11/16

Tables 8.1 a and b. Milestones for the Summer and Fall Design Semesters Respectively

The Milestones listed for this project begin with choosing a project that is both complicated enough to show a strong understanding of the given engineering disciplines but not so difficult as to make the goals listed impossible to complete in the allotted time of two semesters. The group then must research the ideas behind the project to have a strong understanding of how the device will function. Then all of the components will be researched and compared to similar parts to make sure the proper piece is chosen and purchased. The group will then have a meeting to check through all components and begin ordering parts. During this time period the group will be creating a 120-page documentation for the project that will have deadlines at half and full completion that must be met by all team members equally. After all the parts have arrived the team will meet in a lab and visually check each component for possible problems before proceeding to the testing and prototyping phase in the second semester.

The second semester will proceed with the testing of all individual components in a lab setting. Each component will be exposed to the expected tolerances of the device as well as pushed slightly past limits to check for any possible failures. The buoy housing will be tested in lab for any faults before being exposed to the environment as listed in the testing section. All components will pass their testing before the prototype is assembled. Once the Prototype is assembled a new set of tests will be implemented to make sure the device will work under expected conditions. The testing will then be repeated in an attempt to work out any bugs or issues that may occur with the device initially.

In the event of a component failing a test the test will be repeated and the component reevaluated if needed. A possible replacement component may have to be chosen and ordered at that time. Additionally, weather conditions are a possible limitation to testing deadlines as sunlight is needed, as well as moderately large waves at the beach which are both completely out of the groups' control.

8.2 Cost and Budget

Being that the water and air data buoy system we are design is supposed to be a low cost build, our cost and budgets were monitored closely and wisely. We made sure that each component of our design was inexpensive while still being suitable for the overall project.

8.2.1 Total Cost

After determining each device and component of our air and water data buoy system, we have researched costs and started to purchase most of them. From the table below we will be able to understand and analyze where our purchases have been made. We will also be able to calculate the overall cost of the data buoy system.

Name	Quantity	Cost	Total
Microcontroller	1	$5	$5
Air Temperature Sensor	1	$4	$4
Water Temperature Sensor	1	$4	$4
Accelerometer	1	$8	$8
Anemometer	1	$45	$45
Laser Diode	1	$3	$3
Laser Diode Mirror	1	$25	$25
Quad Cell Sensor	1	$45	$45
Solar Panel	2	$30	$60
Battery	2	$10	$20
Buoy Housing	1	$7	$7
Wireless Communication	1	$5	$5
Miscellaneous			$50
Total			~$280

Table 8.2.1. Total Costs and Quantities

All costs are estimated for a single built buoy system. Components will be purchased in multiple quantities in case of failure or malfunction.

8.2.2 Cost Responsibilities

This data buoy system will be fully funded by the group members involved in the design project. At this moment there will be no sponsors. This is subject to change if and when an adequate sponsor is found. It is important to note that several of each component will need to be bought in the event of breakage and another is needed. This should ensure that the part is available to the group in the event that part is discontinued by the seller. The group will have the use of a fiber optic lab for splicing and fusing of fibers so tools will not need to be purchased or rented for this purpose. In addition, an optical bench is available for use in the lab to test the components as needed and will incur no additional cost for this project.

8.2.3 Work Responsibilities

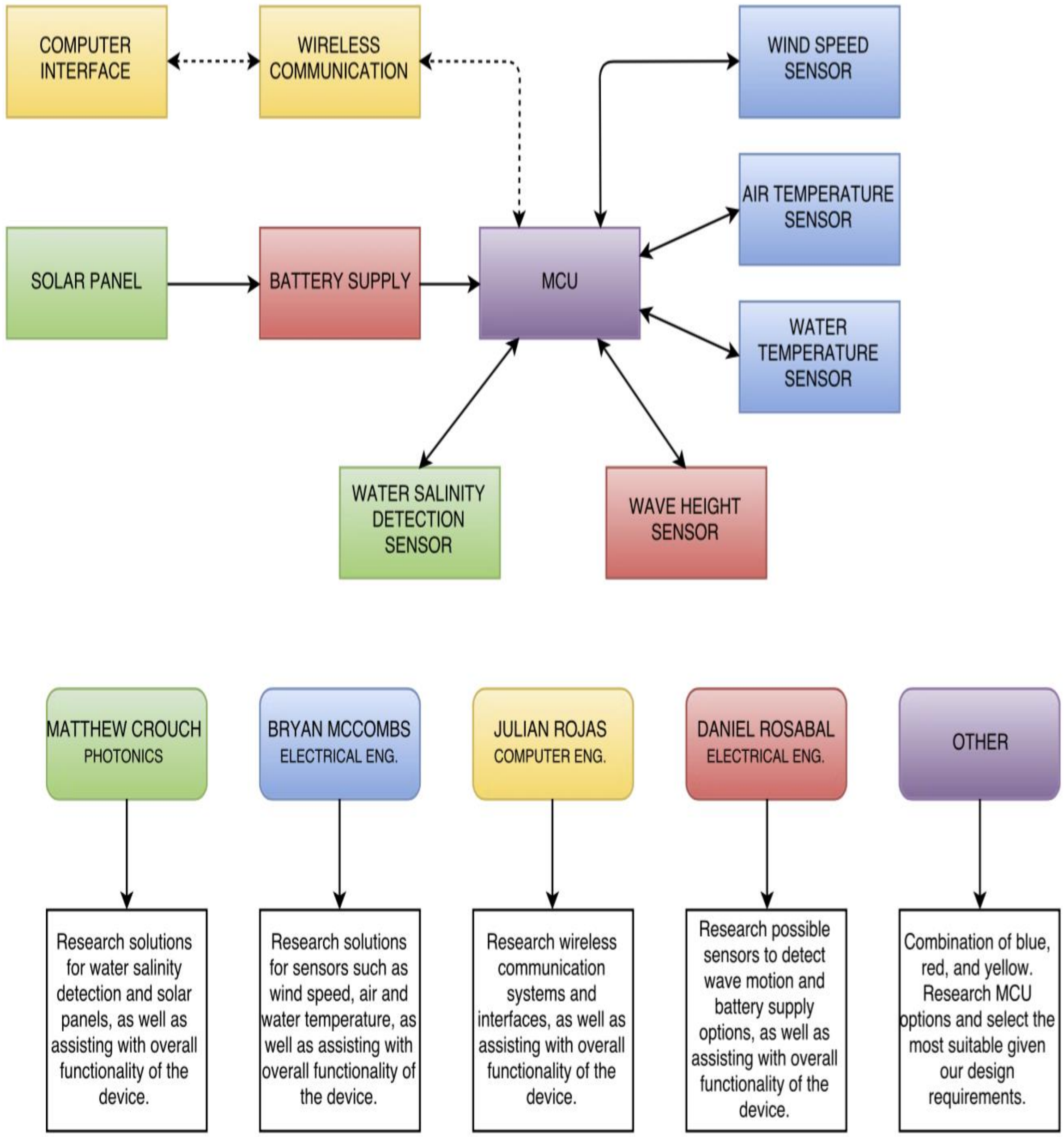


Figure 8.2.3: Work responsibility distribution block diagram

8.3 Marketing Trade-off Matrix

↑ Positive ↓ Negative ↑↑ Strong Positive ↓↓ Strong Negative

Table 8.3. Marketing Trade-off Matrix

	Low Power	Functionality	Setup Time	Dimension	Performance
Usability	↑	↓	↑↑	↑	↑
Accuracy	↑	↑	↓	↑	↑↑
Cost	↑↑	↓↓	↑	↑	↓
Durability		↑	↓	↑	↑

This diagram is to show how the device compares to comparable devices currently in use in the market today. The number of arrows indicates how strongly the device compares in the given column versus the corresponding row. For instance, the first row indicates that this device in terms of Usability is lower power, less functionality, much shorter setup time, smaller dimensions and better performance when comparing to comparable devices. The lower functionality is the trade-off for the increase in every other category in terms of marketing the device.

9.0 Conclusion

Throughout our extensive research and design in this documentation of our data buoy system, we have gained greater knowledge of field training and hands on experience. To begin our design project, we decided that we must find a project that fits the requirements and the descriptions of our Senior Design 1 course. After brainstorming through about 20 different ideas, we finally came up with one that seemed to meet the standards needed. Since we have Electrical, Computer and Photonics engineering majors on our design team, we had to find a project that had enough diversity for each person. The Data Buoy Design seemed very logical being that it needs all three sections of our team's various majors. We feel that a data buoy design can challenge us to use extensive knowledge learned throughout each of our courses take during our time at UCF.

Once we decided the Data Buoy Design task, we began by researching similar systems being used today. Most devices and systems found were particularly large and made for commercial use. This gave us even more motivation to take the challenge and add more features that ca be valuable to the average water sports enthusiast. Being that we have to keep the budget low, the design must be in our ability to be completed within a minimal amount of time.

Now that we had a design motivation and a team to work together, we were able to start the writing of our 120-page documentation. We determined what the functionalities of our buoy would have and the split the writing from there. We had our computer guy writing about the programming and the wireless communications, our Photonics guy writing about the Salinity Detection and Solar Power, and our two Electrical guys writing about the wave height, wind speed, water and air temperature, battery, and the microcontroller. Divided this gave each person around 30 pages each to write about.

To being the writing we went through and crated a table of contents as well as a requirements portion so that we had something to work off of. The starting of the writing was the main struggle because so much research had to be done before any deep and real writing could be done. After each person did their research we kept a weekly monitoring on how much we would get done. This kept us on track with our overall design and writing.

The salinity sensor was the most photonics intensive part of this device and a unique sensor that is not currently in use by any device found in this field. The use of light to obtain the given information from the sample allows us to do more than a device that is purely electronic. The choice of the design for the device was based on the minimization of components required, ease of calibration, reduction of cost, and sensitivity needed. The laser diodes from lightinthebox.com were chosen for being in the right wavelength chosen and being exceptionally cheap in price compared to the other options. The quadrant-cell sensor was large enough to obtain the data that is anticipated to be needed and should work well based on calculations of the group. The constructed sensor should be sufficient to obtain the necessary information from the water samples tested in the coming months.

Once we decided on the certain products, components, and sensors, we were able to start ordering, gathering, and testing them. After receiving all the components, we tested each one to make sure they were working as intended. From our testing section, we can see that all testing that was able to be done at this time seemed to work properly and no major complications arose.

Although our design documentation has come to an end, our research and overall design will still be continued through Senior Design 2. For future development we plan on starting early on building the data buoy and combining all the components and sensors. We want to make sure we are ahead of schedule and don’t fall behind at any time during our project timespan.

Looking through each component of our design, we have found that the salinity detection and wave height sensors will be the most complicated to implement. The salinity detection is going to be fully designed by our photonics and electrical members and programmed by our computer member. The difficulty found in the salinity detection will be in the design, building, and programming of the sensor itself. The difficulty for the wave height sensor, or accelerometer, is going to be in the programming. We are determined to make these two sensors an addition to our project, so we will spend a sufficient amount of time on the implementation and integration.

During Senior Design 1, we learned a few personal skills, as well. First we learned that we must always keep track of time and to get the major things out of the way first and fill in the little things later. This means to never wait until the last minute to do anything. During our design we had a few cases of Murphy’s Law and found ourselves making extra time to resolve the issues. Also, we learned that organization is a key component to making a design, project, or any group activity go as smoothly as possible. Another lesson learned is communication. Without communication there is no organization, especially when it comes to group work.

Leaving this design documentation, we are relieved to say that we are excited to continue and finish this Data Buoy System. Throughout the semester we ran into some complexities, although we always found our way through them. Now that we are finished with documentation we will spend the rest of our Senior Design course, as well as the rest of our time at UCF building and completing the data buoy.

Appendix A References

1-Wire Temperature Sensor | DS18B20 | Stainless Steel | Heavy Duty | Rugged Assembly | |Rated for Automotive Use | 1-Wire Temperature Probe. (2016, June 28). Retrieved June 10, 2016, from http://www.tempsensing.com/sensors/1wire-temperature-sensor-8ssp_3_rgd_3m_3w_b20.php

A Beginner's Guide to Accelerometers. Retrieved July 2, 2016, from https://learn.sparkfun.com/tutorials/accelerometer-basics

Albaladejo, C; Soto, F; Torres, R; Sánchez, P; López, J.A. A Low-Cost Sensor Buoy System for Monitoring Shallow Marine Environments. (2012, July 16). Retrieved June 29, 2016, from http://www.mdpi.com/1424-8220/12/7/9613

Analog Devices "Small, Low Power, 3-Axis ±3g Accelerometer" ADXL335 datasheet. [Revised 2010].

Arduino - One Wire Digital Temperature Sensor - DS18B20. (2016). Retrieved June 30, 2016, from http://www.hobbytronics.co.uk/ds18b20-arduino

Arduino 1-Wire Address Finder. (n.d.). Retrieved June 30, 2016, from https://www.hacktronics.com/Tutorials/arduino-1-wire-address-finder.html

Arduino 1-Wire Tutorial. (n.d.). Retrieved June 30, 2016, from https://www.hacktronics.com/Tutorials/arduino-1-wire-tutorial.html

Battery Definitions and What They Mean. (2016, June 06). Retrieved June 30, 2016, from http://batteryuniversity.com/learn/article/battery_definitions

Bender, L. C., Guinasso, N. L., Walpert, J. N., & Howden, S. D. (2010). A Comparison of Methods for Determining Significant Wave Heights—Applied to a 3-m Discus Buoy during Hurricane Katrina. J. Atmos. Oceanic Technol. Journal of Atmospheric and Oceanic Technology, 27(6), 1012-1028. doi:10.1175/2010jtecho724.1

Brick-Temperature-DS18B20. (n.d.). Retrieved July 5, 2016, from https://arduino-info.wikispaces.com/Brick-Temperature-DS18B20?responseToken=2a24fec4a9becd0fffd0d2a8d14d1499

Burg, J. Getting wind speed from an anemometer. (2014, November 13). Retrieved June 27, 2016, from http://www.hackerscapes.com/2014/11/anemometer/

Charging Lithium-Ion. (2016, May 17). Retrieved July 8, 2016, from http://batteryuniversity.com/learn/article/charging_lithium_ion_batteries

Charging with Solar and Turbine. (2016, June 17). Retrieved July 8, 2016, from
http://batteryuniversity.com/learn/article/charging_with_solar_and_turbine

Comparison of the DS18B20 and DS18S20 1-Wire® Digital Thermometers. (2009, March 18). Retrieved June 30, 2016, from https://www.maximintegrated.com/en/app-notes/index.mvp/id/4377

Comparison Table of Secondary Batteries. (2016, April 11). Retrieved June 30, 2016, from
http://batteryuniversity.com/learn/article/secondary_batteries

Data Buoy Types. (n.d.). Retrieved June 23, 2016, from http://www.jcommops.org/dbcp/platforms/types.html
Patil, S. S., & Shaligram, A. D. (2013). Refractometric Fiber Optic Sensor for Detecting Salinity of Water. *JST Journal of Sensor Technology*, *03*(03), 70-74. doi:10.4236/jst.2013.33012

DS18B20 Waterproof Temperature Sensor. (n.d.). Retrieved July 3, 2016, from http://www.oddwires.com/ds18b20-waterproof-temperature-sensor/

Hayes, J. Understanding Wavelengths In Fiber Optics. (n.d.). Retrieved July 4, 2016, from http://www.thefoa.org/tech/wavelength.htm

HC-06 Bluetooth module datasheet and configuration with Arduino. (2014, May 25). Retrieved June 19, 2016, from http://42bots.com/tutorials/hc-06-bluetooth-module-datasheet-and-configuration-with-arduino/

HC-06 Bluetooth RX & TX voltage (3v3 || 5v). (n.d.). Retrieved June 10, 2016, from http://arduino.stackexchange.com/questions/10472/hc-06-bluetooth-rx-tx-voltage-3v3-5v

Horsey, J. Arduino Wind Speed Meter Anemometer Project - Geeky Gadgets. (2016, March 30). Retrieved July 7, 2016, from http://www.geeky-gadgets.com/arduino-wind-speed-meter-anemometer-project-30032016/

How are spectral wave data derived from buoy motion measurements? (2011, November 28). Retrieved June 10, 2016, from
http://www.ndbc.noaa.gov/wave.shtml

How to Charge and When to Charge. (2016, February 25). Retrieved July 8, 2016, from
http://batteryuniversity.com/learn/article/how_to_charge_when_to_charge_table

How to pick the best temperature sensor for your Arduino project. (2015, February 16). Retrieved June 30, 2016, from https://www.intorobotics.com/pick-best-temperature-sensor-arduino-project/comment-page-1/

K, Toni. Accelerometer Basics. Retrieved July 2, 2016, from https://learn.sparkfun.com/tutorials/accelerometer-basics

Lawson, B. International Standards and Testing Applicable to Batteries. Retrieved July 20, 2016, from http://www.mpoweruk.com/standards.htm#general

Locke, S. (2008, October 20). How does solar power work? Retrieved June 30, 2016, from http://www.scientificamerican.com/article/how-does-solar-power-work/

Microchip “Stand-Alone System Load Sharing and Li-ion / Li-Polymer Battery Charge Management Controller” MCP73871 datasheet. [Revised 2009].
Measuring Temperature. (n.d.). Retrieved June 30, 2016, from https://www.facstaff.bucknell.edu/mastascu/eLessonsHTML/Sensors/TempR.html

One Wire CRC. (2010, January 12). Retrieved June 30, 2016, from https://developer.mbed.org/users/snatch59/notebook/onewirecrc/

One Wire Digital Temperature. DS18B20 Arduino. (2011, July 8). Retrieved June 30, 2015, from http://bildr.org/2011/07/ds18b20-arduino/

Optical Cable Corporation - Water Tolerence vs. Waterproof. (n.d.). Retrieved July 2, 2016, from http://www.occfiber.com/main/index.php?p=31

OSHA STD 01-12-002. (1987, 09 21). Retrieved 26 July, 2016, from Occupational Safety & Health Administration: https://www.osha.gov/pls/oshaweb/owadisp.show_document?p_table=DIRECTIVES&p_id=1703

Raymond, N. Open Source Ocean Data Buoy Project. (2014, May 14). Retrieved June 14, 2016, from https://opensourceoceanweatherbuoy.wordpress.com/

Sorrel, C. Inside the Nobel Prize: How a CCD Works. (2009, October 7). Retrieved June 19, 2016, from http://www.wired.com/2009/10/ccd-inventors-awarded-nobel-prize-40-years-on/

Sensor tutorials. (2012, April 24). Retrieved June 30, 2016, from http://www.ladyada.net/learn/sensors/temp36.html

STMicroelectronics “MEMS digital output motion sensor ultra-low power high performance 3-axes nano accelerometer” LIS3DH datasheet. [Revised May 2010].

TMP36 Temperature Sensor. (2015, November 19). Retrieved June 30, 2016, from https://learn.adafruit.com/tmp36-temperature-sensor/using-a-temp-sensor

TMP36 Temperature Sensor Example Projects. (2015, June 04). Retrieved June 30, 2016, from https://learn.adafruit.com/tmp36-temperature-sensor/example-projects

Tweaking4All.com - How to measure temperature with your Arduino and a DS18B20. (2014, March 22). Retrieved July 12, 2016, from http://www.tweaking4all.com/hardware/arduino/arduino-ds18b20-temperature-sensor/

Types of Lithium-Ion. (2016, July 05). Retrieved July 7, 2016, from http://batteryuniversity.com/learn/article/types_of_lithium_ion

USB, DC & Solar Lipoly Charger. (2012, August 3). Retrieved June 29, 2016, from https://learn.adafruit.com/usb-dc-and-solar-lipoly-charger

Weather Station Wirelessly Connected to Wunderground. (n.d.). Retrieved July 2, 2016, from https://learn.sparkfun.com/tutorials/weather-station-wirelessly-connected-to-wunderground

Woodford, C. (2016, June 8). Accelerometers. Retrieved July 2, 2016, from http://www.explainthatstuff.com/accelerometers.html

Woodford, C. How does fiber optics work? (2016, July 16). Retrieved June 5, 2016, from http://www.explainthatstuff.com/fiberoptics.html

Woodford, C. Semiconductor laser diodes | How they work | Who invented them? (2016, March 16). Retrieved June 13, 2016, from http://www.explainthatstuff.com/semiconductorlaserdiodes.html

Yang, H; Qiao, X; Lim, K. Optical Fiber Sensing of Salinity and Liquid Level. (2014, September 1). Retrieved June 17, 2016, from http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6832571

Appendix B Copyright Permissions

BM

Bryan McCombs
Mon 1:56 PM
BatteryU@cadex.com

Reply all

Sent Items

Hello BatteryU,

My name is Bryan McCombs, and I am a Senior Design student for the College of Engineering and Computer Science at the University of Central Florida. Our team has used many of the sources on your site in regards to our design project and they have been such a great help. We were wondering if we could have the permission to use some of the images/tables that are provided throughout your website to help with visual aids in some of our documentation. We intend to provide credit for anything used. Please email back regarding.

Thank you for your time!

Bryan McCombs

B

BatteryU <BatteryU@cadex.com>
Today 2:36 PM

Hi Bryan,

Yes, you may use the material as requested. Please cite sources where appropriate.

Regards,

John Bradshaw - Marketing Communications Manager
Cadex Electronics Inc. | www.cadex.com
Vancouver | Minneapolis | Frankfurt
Tel: +1 604 231-7777 x319 | Toll Free: 1-800 565-5228

Follow us on Twitter: twitter.com/cadexelectronic
Join us on Facebook: facebook.com/cadexelectronics
Add us on Google+: plus.google.com/+Cadex

BM

Bryan McCombs

Today 10:47 AM

abl@berkeley.edu

Reply all

Sent Items

Hello Allison,

My name is Bryan McCombs and I am a Senior Design student for the College of Engineering and Computer Science at the University of Central Florida. During our design we have used many of the sources provided through the Hackerscapes website and they have been very helpful. My team and I were wondering if its possible for us to use some of the images and resources provided throughout the site in some of our documentations. We intend to fully reference the source anywhere the material is used. Please email back.

Thank You!

Bryan McCombs

AL

Allison Lassiter <allison.lassiter@gmail.com>

Today 9:26 PM

Hi Bryan,

No problem at all! Glad you found the site useful.

Out of curiosity, what is the course that you're taking? Is there a place I could see your/your classmates final projects online?

Best of luck,

Allison

Enquiry Bryan

Support <support@hobbytronics.co.uk>
Today 4:20 AM
Bryan McCombs

Reply all

Hi Bryan

No problem at all. Use any images you like, and good luck with your studies.

Mike
Hobbytronics Ltd

Bryan wrote:
> I am a Senior Design student for College of Engineering and Computer Science at the University of Central Florida. We have bought and used some of the components found on your site. We were wondering if we could have the permission in adding some of your images provided on your site into our design documentation. We intend to mention your name for any sources used. Please email back.
>
> Thank you,
>
> Bryan

Tim Goodman <tim.goodman@kitronik.co.uk>
Wed 9:19 AM
Bryan McCombs

Hi Brian,

I am happy for you to use images if we are given credit.

I hope all goes well in your course.

Thanks,

Tim Goodman

Head of Sales and Marketing

Kitronik Ltd

Full Name: Bryan McCombs

Email Address: BryMcCombs@knights.ucf.edu

Hello OddWire,

My name is Bryan McCombs and I am a Senior Design student for the College of Engineering and Computer Science at the University of Central Florida. My design team has used some of the components found on your site and we would like to ask permission to use some of the images and information provide throughout in our documentation. We intend to mention your name anywhere a source is used. Please email back.

p.s. your site has been very helpful with tutorials for our design as well! T
Thank you!

Bryan McCombs

oddWires
http://www.oddwires.com/

---

ian@oddwires.com
Today 12:57 PM
Bryan McCombs

Reply all

Hi Bryan, no problem. And if you have links to any of your projects we'd like to be able to reference them. Thanks, Ian.

Ian Archbell
oddWires
2570 San Ramon Valley Blvd Suite A103
San Ramon
CA 94583
Tel: (925) 314 5540
Email: ian@oddwires.com

Bryan McCombs
Today 12:23 PM
Office@intorobotics.com

Reply all

Sent Items

Hello IntoRobotics,

My name is Bryan McCombs, and I am a Senior Design student for the College of Engineering and Computer Science at the University of Central Florida. Our team has used multiple tutorials on your site regarding our design project and they have been such a great help. We were wondering if we could have the permission to use some of the images that are provided throughout your website to help with visual aids in some of our documentation. Please email back regarding.

Thank you for your time!

Bryan McCombs

Permission Request to Use Images From Site

calinezul@gmail.com on behalf of into robotics <Office@intorobotics.com>
Today 1:29 PM

Hello,

You can use everything as long as you mention the source.

Best regards,
Dragos

Email:office@intorobotics.com
Blog: www.intorobotics.com
Mobile: 0040-0720535359

A

adafruit@gmail.com on behalf of Adafruit Industries <support@adafruit.com>
Mon 7/18
Daniel Rosabal

totally ok!

On Mon, Jul 18, 2016 at 5:46 PM, Daniel Rosabal <support@adafruit.com> wrote:

> security token :
> contactname : Daniel Rosabal
> email address : daniel.rosabal@knights.ucf.edu
> contact us 2 section : billing
> message text : Good afternoon,
>
> My name is Daniel Rosabal and I am a senior at UCF in Orlando, FL. I am currently involved in a group project for our senior design class in which we are building a data buoy that will collect data for its user's convenience. One of the measurements we are trying to obtain is wave height and we have purchased two accelerometers (LIS3DH and ADXL335) to get the job done. I wanted to ask for your permission to include schematic diagrams and designs for these accelerometers in our report. Thanks in advance!
>
> Daniel Rosabal
> useragent string : Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko) Version/9.1.1 Safari/601.6.17
> Client IP: 50.89.240.16

A

adafruit@gmail.com on behalf of Adafruit Industries <support@adafruit.com>
Tue 3:19 PM
Daniel Rosabal

all good, feel free to!

On Tue, Jul 26, 2016 at 3:12 PM, Daniel Rosabal <support@adafruit.com> wrote:

> security token :
> contactname : Daniel Rosabal
> email address : daniel.rosabal@knights.ucf.edu
> contact us 2 section : billing
> message text : Good afternoon,
>
> My name is Daniel Rosabal and I am a senior at UCF in Orlando, FL. I am currently involved in a group project for our senior design class in which we are building a data buoy that will collect data for its user's convenience. One of requirements is for our system to be solar powered. Being so, we are using one of your solar panels, lithium polymer batteries, and the solar lithium polymer charger. I wanted to ask for your permission to include schematic diagrams and designs for these components in our report. Thanks in advance!
> useragent string : Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko) Version/9.1.1 Safari/601.6.17
> Client IP: 132.170.212.13