$H_2\Omega$

Jadyn Lalich, Matthew Peterson, and Lauren Tyler

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — The objective of this project was to develop a water bottle that tracks how much water its user consumes as well as self-sanitizes to keep people healthy. Drinking enough water daily is critical to the human body's overall function and health. Through the use of a user friendly mobile application, users are actively engaged and encouraged to reach their hydration goal.**

Keywords* — Pressure Sensor, Reed Switch, Accelerometer, Microcontroller, LiPo Batteries, Bluetooth, and Ultraviolet C Light-emitting diode (UV-C LED)

I. INTRODUCTION

There are a few water bottles on the market that either track how much water you drink or self-sanitize, but the H-2-Ohm combines these critical objectives into one bottle. The main goals were to be accurate, low cost, waterproof, and energy efficient. Accuracy of determining the water level in the bottle is critical to keeping track of the user's intake. To do this, a pressure sensor takes measurements every time a sip is detected and the bottle is determined to be upright. A reed switch is in charge of detecting when the cap was removed and then placed back on indicating a sip has been taken. Then an accelerometer will take measurements checking if the water bottle is upright.

All of this information is sent to a microcontroller to analyze. After the correct measurement is determined it then gets sent via Bluetooth to the mobile application. The application is user friendly and displays how much water has currently been consumed. This bottle acts just like a regular water bottle so there is no need for the users to do anything other than relax and take a drink just like they would from any other bottle on any other day. They can also rest easy knowing that the bottle and the water contained inside are sanitized by a UV-C LED to make sure that everything stays, safe, clean, and odor free.

This projected was funded by the group. The budget was $150 for one (1) completed water bottle.

II. PROBLEM FORMULATION

The main objective of this project was to research and design a water bottle that tracks water consumption and self-sanitizes.

A. Water Bottle Requirements

Based on realistic knowledge of necessary and available components, a list of requirements was decided on for the bottle.

- The cost shall not exceed $150.
- The battery life of the bottle shall be greater than or equal to one (1) week.
- The wireless connectivity distance shall be greater than or equal to four (4) feet.
- The wireless paring time should be less than or equal to twenty (20) seconds.
- The water sensing accuracy will be accurate to the nearest ounce.
- The bottle size shall be greater than or equal to twelve (12) ounces.
- The sanitation duration will be 120 seconds.

III. $H_2\Omega$ OVERVIEW

This section describes the hardware and software implemented to design this project.

A. Hardware description

After determining what this project was trying to accomplish the next step was to break down the design into five hardware stages. For the sip detection stage a reed switch is used to determine if the cap has been opened and closed. Sanitation stage will be accomplished using a UV-C LED that requires an input voltage of 6V. Water level sensing is done using a pressure sensor and an accelerometer, where the accelerometer makes sure that the bottle is upright before the pressure is read. The communication stage uses Bluetooth to send data to a mobile application. Finally the power stage is supplied by two 3.7V LiPo batteries in series that feed into two LDOs to step them down to 6V for the LED and 3.3V for the microcontroller, Bluetooth, accelerometer, and pressure sensor. Figure 1 below shows the flow of all these stages and how they interconnect.

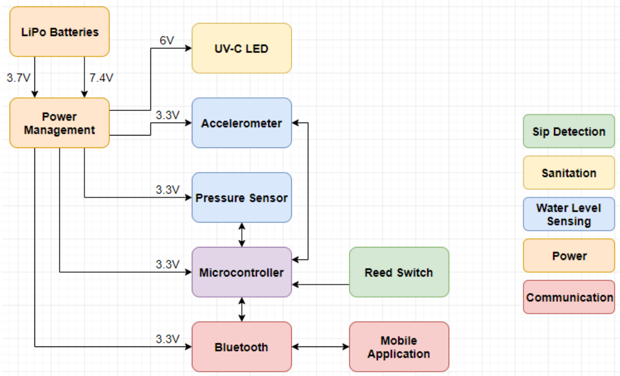

Figure 1: Hardware Block Diagram

B. Software description

For our project, the goal is to have the microcontroller sense when the lid has been opened. Upon opening, if the UV sanitization system is on, it will have its power disconnected for safety of the user. The software then waits for the next time the lid is closed via the 'Lid closed interrupt' at which time the accelerometer will be activated to see if the bottle is upright. This is so the pressure sensor can get an accurate measurement of the water level. The accelerometer will check a few times on a 10 second interval to see if the bottle is upright but the interval will increase to several minutes so as not to waste power. It will continue until either the lid is re-opened, cancelling this task, or it senses that the bottle is upright.

After a water measurement has been made by the pressure sensor, both the accelerometer and pressure sensor go into low power modes until next time the 'Lid closed interrupt' occurs. The microcontroller then saves this data in its memory and compares it to the previous reading so that it can determine the amount of water consumed. This data is then sent through the Bluetooth module to the connected smart device. If no wireless device is connected, the information is stored on the microcontroller memory and will be sent once the Bluetooth is reconnected.

The UV sanitization system is setup so that every few hours, it will power on the UV-C LED and sanitize the water. As stated earlier, this system will be configured to lose its power connection when the lid is opened so that the user is not exposed to direct UV light. All of this is shown in the Figure 2 flowchart.

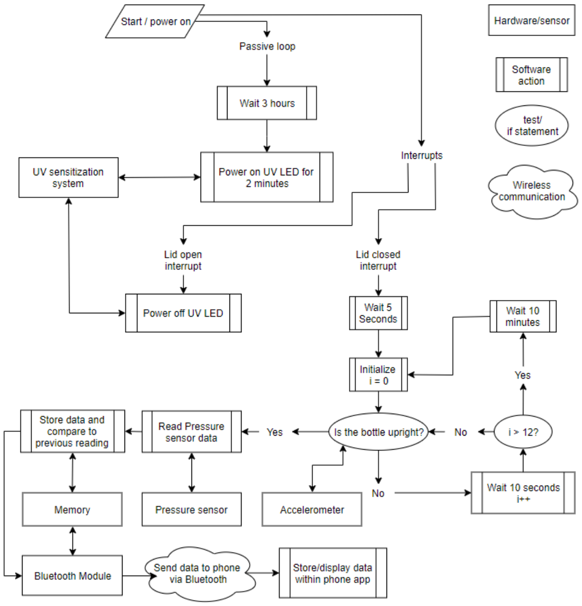

Figure 2: Software Block Diagram

IV. CHOSEN TECHNOLOGIES

Several technologies and methods were researched for all the hardware and software needs. The best technologies for water level sensing, sip detection, sanitation, power, and communication were chosen.

A. Pressure Sensor - MS5837-02BA

To measure the water level in the bottle a pressure sensor was selected as the method of choice. The pressure sensor used for the H-2-Ohm is mounted at the bottom of the bottle so it can read the pressure of the water in a column above it. Utilizing I^2C communication the pressure measurements are sent to the microcontroller. With the pressure reading from the sensor the height of the water can be determined via the following equation:

$$P = \rho g h$$
$$h = \frac{P}{\rho g}$$
$$\rho = desity\ of\ water = 1000 kg/m^3$$
$$g = acceleration\ of\ gravity = 9.81 m/s^2$$
$$P = Pressure\ read\ from\ sensor$$

This utilizes knowns such as the density of water and the gravity of Earth [1]. Which is then used to determine how much water has been consumed based on the difference from the measurement before.

B. Accelerometer – LIS3DH

With the pressure sensor selected as the choice method of water level sensing there was only one flaw that needed to be accounted for and that flaw is measurements taken while the bottle is not in a upright position will be inaccurate. So for example if the bottle is placed in a bag or rolling around in the trunk of a car no readings should take place since the height measurement will be wrong due to how the pressure sensor works. To ensure that the sensor values are only read while the bottle is in an upright position an accelerometer is used.

Once the bottle was built, tests were run to determine what range of x and y values represent when the bottle is upright. After that, the code takes care of not taking measurements unless the accelerometer readings are in that range.

C. Microcontroller – Atmega328P

For this project a microcontroller that is capable of supporting I^2C and UART communication, compatible with the Arduino IDE, is low power, and is small enough to be housed on our small PCB was essential. H-2-Ohm needs to successfully read the amount of water in the bottle from the pressure sensor, take accelerometer readings, sanitize the water with a UV-C LED on a timed interval, detect the reed switch, and relay the data to a mobile application. A microcontroller is the piece that ties all of these tasks together.

Input/Output pins of the microcontroller are utilized for the reed switch and UV-C LED. Where an I^2C communication bus connects the pressure sensor and accelerometer the microcontroller. Lastly UART is used to transfer data from the microcontroller to the Bluetooth and vice versa.

D. UV-C LED – RVXR-280-SB-073105

A UV-C LED is mounted at the bottom of the bottle in order to obtain maximum effectiveness. It sanitizes the inside of the bottle and its contents every three (3) hours for a duration of two (2) minutes. The LED selected has a 120° angle of exposure which helps it reach all of the contents.

Ultraviolet (UV) light is a certain range of wavelengths located on the electromagnetic spectrum. UV light roughly spans from 10nm to 400nm, it is positioned between visible light and X-rays [2]. UV light is broken up into four general categories: vacuum UV, UVC, UVB, and UVA. The subdivision we will be focusing on is UV-C or Ultraviolet C, this has a wavelength range of 100nm to 280nm. UV-C is useful for pathogen reduction in untreated waters [3]. UV-C can be absorbed by proteins and thus, break down cell walls and cause death of organisms.

E. Reed Switch – MK23-80-C-2

The bottle needs to ‘know’ when to take a reading of the current water level by detecting when a sip has been taken. Using a reed switch mounted at the lip of the bottle and magnets placed on the cap, the microcontroller can determine when the cap has been taken off and then placed back on the bottle using a logic pin.

When the I/O pin of the microcontroller connected to the read switch reads LOW the cap is on and when it reads HIGH the cap is off. Also, the bottle needs to deactivate the UV-C LED when the user is taking a drink, otherwise the users’ eyes and skin could be harmed by the light.

This all works based on the science behind the switch. The common form of this switch includes a pair of ferromagnetic metal contacts (reeds) that are overlapping and contained in a glass sealed envelope. When there is no magnetic field being applied the metal contacts are not touching, and the circuit is open. When a magnetic field is applied, in close proximity to the metal reeds, the circuit becomes closed. This electrical component is commonly used for burglar alarms, laptop sensors and proximity sensors.

F. Bluetooth – RN4870-V/RM118

Bluetooth was selected as the method of wireless communication with the mobile application. Through this the user can pair with the bottle quickly and easily as well as from a distance. When the new water consumption measurements have been calculated they are sent via Bluetooth to the mobile application. It was important that this device be able to communicate via UART with the microcontroller to accomplish the task of getting the data to send to the phone.

G. Battery – LiPo

To ensure that H-2-Ohm worked at all, a strong enough power source had to be selected that could one run the entire load, and two last long enough such that the user doesn’t have to charge the batteries an extreme amount of times.

Since there are two main power lines (3.3V and 6V), two 3.7V LiPo batteries are connected in series and use LDOs to step down each stage to the desired voltage. The 3.7V is stepped down to 3.3V and the 7.4V is stepped down to 6V. The batteries are easy to remove so the user can charge them outside of the bottle for safety purposes.

Based on a 16hr day with roughly 5 sips per hour and all the components running at max power, 62.13mAh would be needed to run for the entire day. Which means that the batteries will last over a week before needing to be recharged.

V. DESIGN

Three major designs had to be accomplished for this project. The bottle design which will house all of our components, the hardware schematic design which determines how everything is connected, and the PCB design which incorporates the circuits onto a small board.

A. Bottle Design

A single-walled stainless-steel bottle was selected so the UV-C light will not escape the container and so that the electronics will be protected by the durable shell. Single-walled was important because two small holes were drilled in the bottom of the bottle for the UV-C LED and pressure sensor. Rubber o-rings and food grade sealant material were placed around the sensor edges and UV-C LED to ensure that no water leaks through since the PCB is housed directly below as shown in Figure 3.

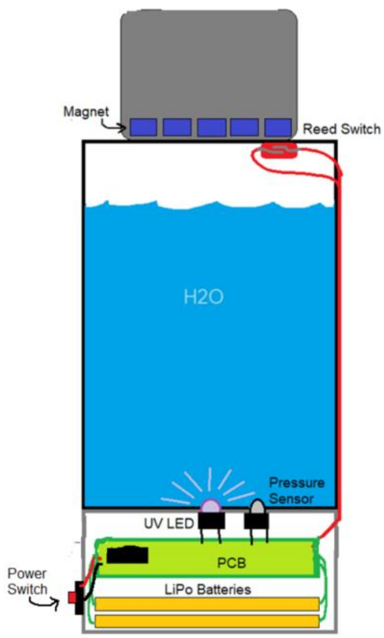

Figure 3: Bottle Design

Female jumper wires are used to attach everything mounted on the bottle to the PCB. The entire electronics portion of H-2-Ohm is housed in a 3D-Printed base section. This bottom section is sealed to the base of the stainless-steel water bottle using an epoxy glue mixture to fuse the plastic base and the metal base together. The actual design can be seen in Figure 4.

Figure 4: Bottle Housing

For experimenting and testing purposes, the side of the electronics housing is completely open/exposed. The actual product would not be in this configuration, this initial product is a prototype and having the ability to test components is essential.

B. Hardware Schematic Design

After breadboard testing was completed a hardware schematic was created to implement the same connections. The accelerometer, microcontroller, and pressure sensor are all connected to the I^2C bus which is pulled high using two 10k resistors. As for the Bluetooth, it connects via UART to the microcontroller.

Everything is powered by the main 3.3V line other than the UV-C LED which uses the 6V line. The reed switch is read via 3.3IO3 and the microcontroller enables/disables the UV-C LED via 3.3IO4.

A few headers are used in this design. The pressure sensor header, UV-C header, and reed switch header will be used to connect those components to the PCB. Where the programming headers and other miscellaneous headers are used when trying to debug the board as well as change the programming (during actual operation none of those headers are need to be used).

Lastly the power management has two headers which connect the ON/OFF switch to the board and then the series 3.7V LiPo battery connectors that go to their respective LDOs to step down the voltage.

C. Printed Circuit Board Design (PCB)

To design the PCB, the schematic components as well as their connections were imported. The design was done in the most compact and efficient way possible to fit under the bottle. The dimensions of the board are 2in x 1.5in and to make the board affordable only two (2) layers were used.

To route the board many vias were needed to connect traces on the two different layers so routing could take place on both the top and bottom. With so many components on such a compact board routing was difficult but not impossible as seen in Figure 5. The microcontroller, Bluetooth, accelerometer, and power management are directly soldered on the board where everything else is connected to the many headers on the PCB.

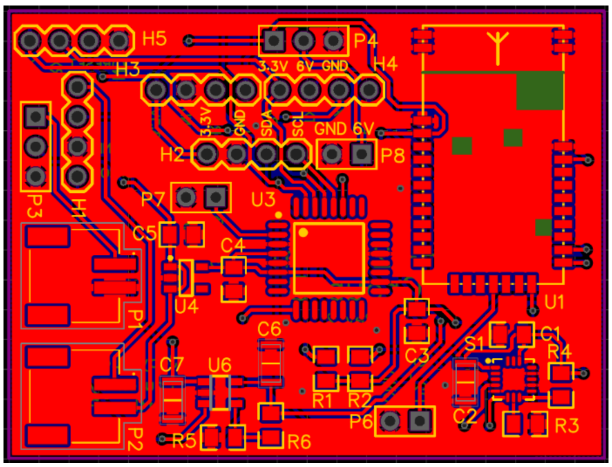

Figure 5: PCB Design

VI. Pressure Sensor PCB

Since the pressure sensor is a surface mount component a small PCB had to be designed to mount it on as well as a 100nF capacitor (C1) close to the VCC pin to stabilize the voltage. Since the selected senor only has four pins, two headers were used (P1 & P2) with two pins each. The pins are VCC, GND, SDA, and SCL which will be connected via wires to the main PCB.

Since both the pressure sensor PCB and the UV-C LED are at the bottom of the bottle, space was limited. The PCB seen in Figure 6 is a 2 layer design with dimensions of 0.5in x 0.5in. Using rubber seals and USDA approved silicon sealant both the UV-C LED and pressure sensor PCB were mounted and checked to ensure that no water leaked out of the base. The PCB size works perfectly with the design and once everything was set tests were run to ensure the silicon sealant did not mess with any of the electrical connections. The bottle was filled with water and the different pressure measurements were analyzed as to find a good way to calibrate our calculations to get the proper water level reading.

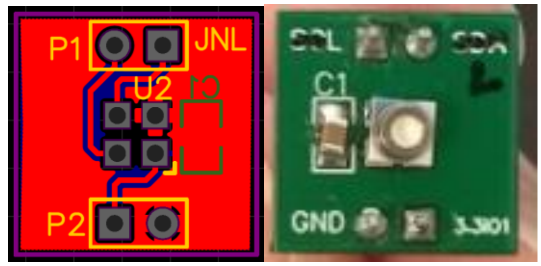

Figure 6: Pressure Sensor PCB Design

VII. MOBLIE APPLICATION

The H-2-Ohm application was built to be a simple design while displaying all the information on the user's water intake that they might want to see. The app was built using MIT's App Inventor, a free development tool. All data is stored locally on the device and the last reading is also stored on the H-2-Ohm bottle. The data stored with the application is transmitted over a Bluetooth low energy module using a UART transparent connection which allows data to be transmitted between the bottle and the app with minimal intermediate steps. The app uses the date as a storage tag which makes it easy to access later if we decide to implement features that allow the user to see more than the past week of data. In Figure 7 on the home screen (left), pressing the refill button will calibrate the bottle after refilling it. Pressing the gear icon in the top left displays the settings which allows the user to set goals, adjust notification settings, and connect over Bluetooth to the H-2-Ohm bottle. Also in Figure 7, the Bluetooth screen (right) shows the options which allow the user to connect to the bottle.

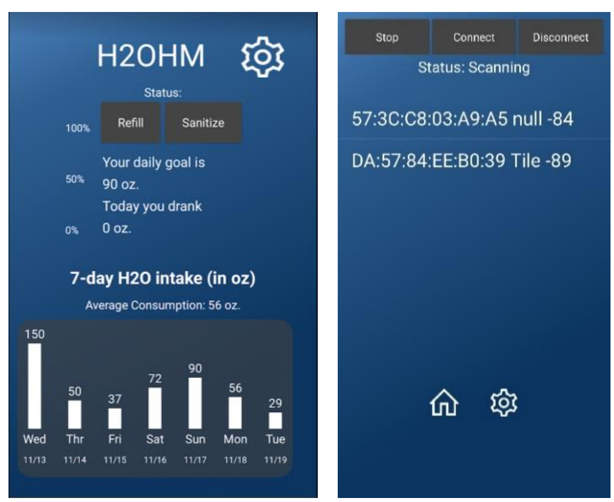

Figure 7: Mobile Application

VIII. Bluetooth Programming

The Bluetooth module is pre-programmed to run in UART transparent mode which in the RN4870 has built in services and characteristic UUIDs to allow data to be sent directly from the micro-controller to the mobile app.

The module was also configured to run at 9600 baud rate to match the speed of the microcontroller.

A few functions require the microcontroller to send commands to the Bluetooth module rather than the mobile app. This is possible using the "$$$" command which lets the Bluetooth module know that the following commands (i.e. pairing or severing the Bluetooth connection) are to be received and interpreted rather than sent over UART transparent to the app. Then after all the commands are done sending to the module, the MCU sends the "---" command, telling the module to return to UART transparent mode.

The Bluetooth module is also designed to send updates over UART to the microcontroller whenever something happens. Whether it be that a new device is connected, paired, disconnected, or otherwise. These updates always use '%' as a delimiter. The app was designed to use the same delimiter when sending data to the microcontroller. Therefore, any data received by the microcontroller that does not use this delimiter is ignored as it does not come from the app or the Bluetooth module

IX. Using The $H_2\Omega$

Starting with the very basics on pairing the bottle with a mobile phone, using the app, filling the bottle, ensuring proper measurements are taking place, and maintaining the bottle will all be explained in detail. It is important that all of these steps and procedures are followed for best results while using the H-2-Ohm.

A. Mobile Phone Settings

To start out the day using the H-2-Ohm, it is important to first power on the bottle using the switch located at the base of the bottle. Then on the user's mobile device they should go to their settings and enable (turn on) Bluetooth and enable (turn on) their location.

B. Using the App

Once the app is open, the user should navigate to the settings page and select "Bluetooth Setup". Once on the Bluetooth page select to "scan" for nearby devices. Depending on the phone and environment, it may take a few seconds for the mobile phone to find the H-2-Ohm Bluetooth but the wait should be no longer than 20 seconds. When the H-2-Ohm is found select it from the list of devices and select "connect". If a connection was made the status will say "connected". As for the other feature on the settings page, this is where the user will go to set their daily water goal. On the home screen, the daily water statics appear and this is where the user will spend most of their time viewing their daily progress and goal. This is the page the app defaults to when opened.

C. Filling the Bottle and Drinking

For daily first-time use, it is important that the device has been turned on and paired before you fill the bottle. That way, as soon as the user places the cap on and sets the bottle down (preferably on a hard surface), the starting measurements will take place when the user press the "refill". Other than that, the user can use the bottle just like any other throughout the rest of the day. It is important that the cap be placed back on the bottle if the user would like to see the measurements of the sips they just took right then and there. Measurements only take place after the cap has been placed back on the bottle, so if the user finishes the contents and does not replace the cap before refilling, whatever they drank after the last measurement will not be accounted for. A simple way to keep this from happening is to replace the cap and allow time for a measurement to take place before filling the bottle again. If the user forgets to press the "refill" button after refilling the bottle the app will notify them that they should press it since water was added in the bottle.

D. Maintaining the Bottle

After the user is done using the bottle for the day, the power should be turned off to save the battery and the remaining water should be poured out. Store the bottle on a drying rack or in a cabinet without the lid on. The lid can be washed by hand with soap and warm water. It is very important that the bottle itself is NOT washed since it self sanitizes and has electronic components. If the user feels their bottle is particularly dirty a wet wipe should be used to wipe the bottle clean. Please make sure the bottle has no liquid in it and can dry overnight. This will help the overall user experience and the lifespan of the H-2-Ohm.

X. Testing

Since the H-2-Ohm is so modular each component was tested individually before combining them as a whole. Most components required the microcontroller so it was used in almost all of the tests. The main components that needed to be tested were the pressure sensor, accelerometer, UV-C LED, reed switch, Bluetooth, and power stage. After all of the components were tested individually the process of putting everything together on a breadboard helped with schematic design and the end game PCB layout.

Final tests were run when the PCB was completed and everything was mounted to the bottle.

A. Pressure Sensor Testing

Some basic test were run to see if the pressures sensor breakout board worked as well as communicating with the microcontroller via I^2C. Using a breadboard setup, the pressure readings were read off of the serial terminal of the Arduino IDE confirming that the device indeed worked as expected.

After that was confirmed, the pressure sensor was mounted to the bottle and water measurements were gathered. The bottle was filled and then using a measuring cup water was poured out ounce by ounce to see how the spread of the data looked. Using the average difference between the spread this value is used in the final code to determine how many ounces are poured out. Then with the final code the same test was run where water was poured out into a measuring cup to confirm accuracy.

B. Accelerometer Testing

Determining whether the bottle is upright is the only objective the accelerometer is concerned with so the tests run on this component did not need to be in depth or too complex. The code was loaded onto the microcontroller and the accelerometer was run. The tests took measurements of the board sitting upright and then slowly tilt it in every direction. With this calibration taking place a small range of values was noted and the final code uses that range to test if the bottle is upright.

C. UV-C LED Testing

The initial UV-C LED testing was done by setting up the LED in a box as to not hurt our eyes. LEDs are diodes so the cathode (-) was plugged into the negative voltage supply terminal and the anode (+) was plugged into the positive voltage supply terminal for the component to work. According to the datasheet, the typical forward voltage for this UV-C LED is 6.5V, this value was set in the power supply which made the power supply automatically switch to constant current mode. We want to power the LED with a constant current instead of a constant voltage because the turn-on voltage varies between each manufactured LED.

This particular LED that we tested only requires 5.55V forward voltage at its nominal current (100mA), which was almost 1V lower than what was expected. It's important to note that the UV-C LED acts as a constant current load, this means that it will draw a constant amount of current but the voltage going to the load will vary. So a 6V rail will work desirably to power the LED.

D. Reed Switch Testing

To test the reed switch, two wires were attached to either side and then the switch was placed on the bottle near the magnets placed on the cap. When the switch was within a certain distance of the magnet the switch would close and read a small resistance on the multimeter and when the cap was moved away the switch would open and the resistance on the multimeter was that of an open circuit.

With the initial test out of the way the switch was connected to the microcontroller and tested to see if the I/O pin could determine when the signal was HIGH or LOW.

E. Bluetooth Testing

When testing to make sure that the Bluetooth was working correctly, we wanted to make sure that the Bluetooth module responded appropriately to the microcontroller instructions, stayed connected within its range, connected to paired devices within a reasonable amount of time, and correctly transmitted data to the paired device.

The first thing tested was to make sure that the Bluetooth module paired correctly with an Android phone with no major issues. While, making sure this very basic function worked correctly, we also confirmed the time it takes to pair with a smart phone and then connect to it, is short and does not take too long as this could severely harm the user's overall experience with the H-2-Ohm module.

Lastly, we made sure that the data sent from the microcontroller for transmission was able to be seen by the mobile device being used as well as from the mobile device to the microcontroller. This was tested using several different data values including extreme values that have a higher chance of being miss-interpreted during its transmission to the user's mobile device.

F. Power Stage Testing

The two LiPo batteries were tested to ensure that they were producing the expected 3.7V as well as to make sure that in their series configuration the second node has a voltage of 7.4V. All of this was tested with a multimeter and after that was confirmed both voltage regulators were tested to see that they were giving their desired output of a steady 3.3V and a steady 6V. These tests were not complex but they were essential to ensuring that the proper voltages were attained.

XI. CONCLUSION

Starting out with a simple idea of how to improve people's health by improving their water intake has led this team down a long path to perfecting an innovative and interactive solution. Creating the H-2-Ohm was not done overnight and took extensive research in Senior Design 1 and then the design/creating the physical project was truly put to the test in Senior Design 2.

Overall, this design process has been a huge learning experience for everyone on the team. For the most part, the first pieces of the project consisted of research, concept designs, minimal testing, and explanations of what we planned to do in the near to come future. Then that future hit us fast as we made sure to keep up with the aggressive Senior Design 2 schedule to complete the project. Working well with a team is an important skill to have when it comes to being successful in any professional setting. Engineering is about using your knowledge to simplify complex ideas and issues, into something that can work successfully in the real world. H-2-Ohm was an original idea developed by our team, the steps that we took to make it become a real product were priceless experiences.

ACKNOWLEDGEMENT

The group would like to thank Chung Yong Chan for his help and very determined methods to instill what it means to be an engineer and to learn from your mistakes. The group would also like to thank UCF for more than four magical years and for our gray hair. To date pursuing these engineering degrees have been one of the most difficult experiences of every team members lives.

REFERENCES

[1] The Engineering ToolBox, "Hydrostatic Pressure - Depth and Hydrostatic Pressure," [Online]. Available: https://www.engineeringtoolbox.com/hydrostatic-pressure-water-d_1632.html. [Accessed 02 2019].

[2] Wikipedia, "Ultraviolet," 17 February 2019. [Online]. Available: https://en.wikipedia.org/wiki/Ultraviolet.

[3] I. Crystal IS, "KLARAN," 2019. [Online]. Available: https://www.klaran.com/klaran-university/about-uvc.

Jadyn Lalich will graduate and receive her Bachelor's of Science Degree in Electrical Engineering in December of 2019. She will begin her career at L3Harris Technologies in Palm Bay. In her free time she enjoys traveling, hiking, and spending time with family and friends.

Lauren Tyler will graduate and receive her Bachelor's of Science degree in Electrical Engineering in December of 2019. She will then further her education at UCF through pursuit of a Master's of Science degree in Electrical Engineering, projected to graduate December of 2020. She has been a member of the Air Force National Guard since 2012 and plans to stay in the military while maintaining an engineering career. She enjoys skiing, hiking, and tending to her plants.

Matthew Peterson comes from a family of 11 kids and has a background in Home IT. He will graduate in December of 2019 with a Bachelor's degree in Computer Engineering at UCF and he is currently pursuing several job opportunities. In his free time, he enjoys long naps and hanging out with friends as well as playing and creating video games.