UNIVERSITY OF
CENTRAL FLORIDA

Senior Design 1 Documentation

Smart Water Bottle

Prepared by:

Group #5
Self-Sponsored

Neeil Gandhi **(Photonic Science and Engineering)**
Ryan Koons **(Electrical Engineering)**
Dean Pickett **(Electrical Engineering)**
Matthew Woodruff **(Photonic Science and Engineering)**

Submission Date: 07/23/2021

Dr. Samuel Richie & Dr. Aravinda Kar
EEL 4914, Summer 2021
Department of Electrical & Computer Engineering
CREOL, The College of Optics and Photonics
University of Central Florida

Table of Contents

Table of Figures

Table of Tables

1 Executive Summary

According to the CDC, between 30 and 70 percent of global explorers suffer from traveler's sickness. Of those individuals, more than 80 percent fall ill due to bacterial pathogens [1]. One of the most common ways that travelers get exposed to bacteria is through the consumption of biological contaminants in water from underdeveloped sanitation systems. A single sip of improperly treated hydration can expose people to various gruesome diseases including typhoid, cholera, salmonellosis, and a myriad of unpleasant maladies [2]. With sterility in the vogue, virus and disease prevention has prime real estate in the minds of much of the population. And, due to a continuous need for water, pure drinking water is integral to personal wellbeing. This project seeks to design a long-term, portable, and affordable solution for consumers seeking more assurance about their water's quality. A water bottle capable of providing information about the drinkability of its contents would be very useful for a lot of different audiences; It is useful for those that are travelling through foreign nations, but it also applies to those merely going about their daily routine.

It is quite common practice in the industry today, to utilize optical light to sanitize drinking water and make it more potable. Optical sanitization would be a quick and convenient method for anyone to sanitize their drinking water. In addition to this, creating a "smart" product would incorporate additional functionality and information about their drinking water that could be made available to the user. The proposed product is a smart water bottle that can sanitize its contents using UV irradiation and generating a potability report of said contents based on spectral analysis. The sanitization process will start whenever the user initiates the procedure via a button on the bottle. After sanitization is complete, the user will have their own personal opportunity to remove any remaining non-biological contaminants with a standard filter. Water quality analysis will then take place and display a preliminary result (via an LED) on the bottle. The device will also transmit more information about the water quality to the user's personal smartphone device.

It is true that one intent behind this product is to ensure peace-of-mind for international travelers concerned by water-borne pathogens and parasites. However, this product should also appeal to the tech savvy consumer that likes to have "smart" devices with added functionality for ease in daily life. Some of this added functionality will include wireless charging to simplify daily device operation, as well as Bluetooth connectivity and an app to observe more detailed water quality information.

2 Initial Project Description

2.1 Project Narrative and Motivation

From a technology standpoint, several systems will be considered for implementation into the smart water bottle. It will have wireless charging built into the cap to allow for convenient recharges of the device as well as improved water resistance. The optics system will consist of a UV diode for sanitizing the biological contaminants and a Raman spectroscope for determining the content's potability post-sanitization. The bottle will also have LEDs for status and result indications, a Bluetooth module for communications with a paired device, a reed switch to ensure the sanitizer only runs in appropriate conditions, and an embedded microcontroller. All of the electronics will be contained inside of the cap so that users may choose any compatible metal water bottle. The Bluetooth functionality will potentially pair with an app that displays a cumulative water score based on the EPA's regulatory maximums [3]. The intent is to keep the product as simple and coherent as possible for any traveler to use while still allowing the owner to access more detailed data via the app.

Most smart bottles in the market today implement less useful features such as glowing to remind you when to drink, a built-in Bluetooth speaker, and water intake tracking. Some bottles are also UV sanitizing but tend to run on a schedule and do not allow the user to simply press a button to initiate sanitization. Other senior design smart water bottle projects also had water sanitization and smart features such as water intake and a Bluetooth app- however, these projects did not incorporate wireless charging or introduce a CREOL focus on fine-tuning optics technology for the purposes of water quality assurance and monitoring. Previous projects also lack modularity, a feature of significant importance as the proposed device can be used with a variety of vessels.

2.2 Project Components

2.2.1 Water Quality Sensor

This component will perform spectral analysis of the contained fluids. One potential method would consist of a spectrometer that projects the spectrum of the radiation pumped into the sample by the sanitization component onto a CCD or CMOS sensor. Another option would use a photodiode and rotational mechanism to step through a range of wavelengths, recording at intervals. The stainless-steel bottle's reflective surface will serve as a feedback system, amplifying the admittedly insignificant influence of Raman scattering to the point of measurability. The spectrum will then be normalized to account for variations in transmissibility and compared to a spectrum produced by potable water. Significant deviation will trigger a warning LED via the embedded hardware. A copy of the spectrum will then be transmitted to a paired device for more detailed analysis. The spectrum will be produced by projecting a laser beam into the contents with the focus approximately halfway down the bottle when filled entirely.

2.2.2 Sanitization

This component is going to help with the sanitization of the water in the bottle. The components for this are going to be placed in the bottle cap. We are going to combine a couple of LEDs with different peak wavelength, so we have a broad spectrum of light emitted and keep sure we have enough intensity to sanitize the water completely. According to NIST the tunable laser is more efficient in killing the biological contaminants in the water [4]. This light will help kill the biological contaminants present in the water and make it safer to drink the water. This will help prevent from water-borne diseases, like typhoid and cholera, that are caused by the biological contaminants in the water.

2.2.3 Device Power

The device will be powered by a rechargeable battery. It has yet to be fully explored which chemical composition of battery would best suit the project needs, but the two options under investigation are Nickel-Metal Hydride and Lithium-Ion battery technologies. The charged battery will supply power to the rest of the components with DC-DC voltage regulators, specified by the various voltage requirements for each individual component.

Charging the battery is to be achieved by wireless transmission of AC voltage supplied by a standard US 120 Volt wall outlet. The initial intention is to use a wireless power transmitter that will contain the inductive AC power transmission coil to be closely aligned with the top of the water bottle cap while not in use. While the idea to build the power transmitter is being explored, the goal is to create a product that will work with any of the power transmitters currently sold in the market. Many people already have at least one wireless charging dock due to the rise in popularity of wirelessly charged phones and other accessories. The wireless power transmission will be received by inductive coils within the water bottle cap and the AC voltage will be rectified and connected to a regulator in order to charge the battery.

2.2.4 Embedded Hardware

The device hardware will mainly involve sensors, LEDs, a microcontroller, and a Bluetooth module. There will be a water quality sensor, a sanitizing LED controlled by the microcontroller, and a reed switch for determining if the cap is on. Having a reed switch is especially important since UVC light is harmful to human beings and therefore needs to be contained inside the bottle. All of the sensors will be controlled by the microcontroller. It is preferable to use a TI microcontroller due to the familiarity of embedded programming software. The microcontroller also must have enough I/O pins to support the sensors (reed switch included), LEDs, and the Bluetooth module. A microcontroller with a low power mode is strongly desired to improve the battery life of the product, as well as Low level wake up functionality with device peripherals. Due to some of the peripherals, it will also have to be compatible with common communication protocols such as UART and I2C (perhaps also SPI). The microcontroller (even with high-drive current pins) most likely will not be able to drive enough current to the laser, so a MOSFET switching circuit will likely be implemented for enabling/disabling the UVC laser diode. A red/green/yellow LED (preferably in one package) will be used to provide rudimentary water quality info. to the user after sanitization has taken place. It is proposed to include an Integrated Circuit to provide a dot/bar voltage display that shows the charge status of the battery in an array of 10 LEDs [5]. There will be a button to start sanitization, and another for checking water quality. Either one of these buttons can be held down for three seconds to enter Bluetooth pairing mode. From a design standpoint: EAGLE PCB design software will be utilized due to ongoing familiarity, and the boards will most likely be ordered and assembled by PCBWay. A turn-key assembly will likely be utilized due to the current industry-wide chip shortage.

2.2.5 Software

The software will be comprised of embedded firmware for the board itself, and then rudimentary app development. Most likely Code Composer studio will be utilized for creating the firmware for the microcontroller. The firmware must be able to check that the cap is on before activating sanitization, enable/disable sanitization, check the quality of the water, display rudimentary water quality information via LEDs, and wirelessly communicate with the iOS app via Bluetooth. It will require a significant amount of time to learn about the various sensors and how to interface with them (via the microcontroller). The sensors will require implementing standard communication protocol functions such as UART Tx/Rx and I2C commands. A lot of development overhead will also go into implementing Bluetooth wireless protocols to successfully transmit water quality data to the application. It will also be necessary to include code for processing and storing the various water quality statistics. When the device is not readily being used, the microcontroller should also enter low power mode to conserve energy.

The app will receive data via Bluetooth from the device to report more in-depth water quality data to the user. Using a photodiode and servomotor design, this data will consist primarily of a vector of the sample's spectrum. This vector will be produced by incrementally moving the servomotor, sweeping the spectrum over the photodiode. At each increment, the photodiode voltage will be paired with the servomotor position. This information will be used on the bottle-side to perform basic pass/fail tests, and will be transmitted, unmodified, to the app. This vector will contain values for wavelength ranges which can be analyzed for specific excitation responses. The existence of excitation responses indicates the presence of contaminants and said responses will be compared to responses obtained from samples with known contamination levels. Exceeding the accepted maximum values will trigger an alert on the bottle while the app will show the collected spectrum information and produce estimated contaminant concentrations.

3 Requirements

The requirements sections details project requirements chosen based on expected customer demands, regulatory standards, and technological limitations. Table 1 covers the sanitizer and analyzer's black-box performance requirements, and Table 2 outlines a set of physical dimensions which should prove desirable to consumers. Table 3 covers basic features of the sanitizer.

Performance

Maximum Automatic Shutoff Delay for On-board Laser	1 second	
Time for one sanitization cycle	3 minutes	
Sanitization effectivity	Kills 99.9% of contained micro-organisms	
Sensor detects the presence of contaminants in at least these concentrations [3]	Contaminant	Concentration (ppm)
	Arsenic	.010
	Nitrate	10
	Nitrite	1
	Copper	1.3
	Fluoride	4
	Lead	.015

Table 1: Requirements—Performance

Physical Characteristics/Dimensions

Water Bottle Lid	Round 3 1/4” diameter x 5” tall Plastic top to allow for inductive charging
Water Bottle Body	Round 3” diameter x 8” tall
Waterproof	All circuitries should be contained within IP44 certified waterproof housing

Table 2: Requirements—Physical Characteristics/Dimensions

Sanitization

Pump Source	Multiple LED sources to get a broad spectrum. LED with peak wavelength of 265 nm and 275 nm.
Focusing Lenses	Aspherical lenses with collimating tube
Beam Splitter	The beam splitter will help combining the two beams from the LEDs.
Output Coupler	Need a broad spectrum so just will couple the different LEDs with a lower wavelength of 250nm and a higher wavelength of 270nm.
Output lens	Will most likely be a dispersing medium like an LED or a dispersing lens as the light needs to travel in different directions.

Table 3: Requirements—Sanitization

Continuing on, Table 4 outlines basic aspects of the spectrometer component, Table 5 provides general requirements for device power, and Table 6 expounds on embedded hardware requirements.

Water Quality Sensor

Overall Sensitivity	The spectrometer must be capable of reliably identifying the contaminants and concentrations listed in Table 1 in an aqueous solution located in the device's bottle
Overall Size	The spectrometer must have 2" or less in height of the base of the cap (threaded region excluded) dedicated to its components

Table 4: Requirements—Water Quality Sensor

Device Power

Battery Lifespan	At least 10 cycles of the sanitization and sensor modules.
Inductive Charging Speed	Less than 8 hours from drained to charged.
Charging Pad Connection	A standard (US) 120 VAC wall outlet and consist of an inductive transmitter coil, an on/off switch, protective housing, and a permanent magnet (for coil alignment).

Table 5: Requirements—Device Power

Embedded Hardware

Controller	About 10-20 GPIO pins for device operation. Low power mode functionality with current in the tens of microamps.
Wireless Communication	BLE: reduced power consumption. OTA data rate: 1 Mbps. with 6ms latency.
Wired Communication Protocols	UART: Full-duplex serial system. I2C: Device addressing-serial bus
Water Quality Sensor	ADC and/or comparator used to detect photodiode output to the resolution of microvolts.
Bluetooth Pairing	Device will have a button to hold down for 3 seconds and enter device pairing mode.

Table 6: Requirement—Embedded Hardware

The final two tables of this section (Table 7 and Table 8) cover the basic requirements of the wireless communication feature and the app features respectively.

Wireless Communication

Connection Distance	Device should be able to pair with a smartphone from a distance of 10 feet or less.
Pairing Time	The device should take no longer than 30 seconds to pair/connect with a smartphone.

Table 7: Requirements—Wireless Communication

Software/App

Application	iOS platform. Bluetooth pairing and compatibility. Displays the acquired spectra contrasted with a benchmark (pure) spectrum, and singles out the spectra of key contaminants
Interrupt Handling	Reed switch and sanitization. Events occur within 3 second of a button press.
Communication Protocols	I2C, UART, and Bluetooth (most likely 4.0/BLE)
Microcontroller low power mode	Enable low power mode to allow for a sleep current of 10 microamps or less
Water Quality	Transfer spectrometer outputs to microcontroller and evaluate the presence of contaminants as listed in the performance table

Table 8: Requirements—Software/App

4 Initial Prototype Model

The following section will roughly outline the organization of the devices' subsystems. This is a very initial vision of the physical layout and will most likely change later on during the prototype/development process. The very top of the lid (leftmost side in figure below) will contain the power electronics for the device. This involves the wireless receiver, battery chagrining circuit, as well as the battery itself. The next layer down will be the main PCB. This layer will have the microcontroller, 90 degrees status indicating LEDs (facing outward, and most of the embedded hardware. The next layer will consist of all of the components and circuitry for the spectrometer and water quality sensing. The lowest layer will consist of the optical lasers (for sanitization and water quality sensing) as well as sensor windows. Any layers that need to be electrically isolated from one another, can have a thin insulator layer between them.

Currently, the plan is to 3D print a water bottle lid cross-section that could be adhered/attached to a compatible water bottle lid. Modifying an already existing water bottle lid would preserve the original cap's threaded region and base for its durability to UV and reflectivity. Holes could be drilled through for the optical/laser diodes as well as any sensor windows. Similarly, a hole could be made on the sides of the electronic portion of the bottle so that the indicator LEDs are visible from the outside. There would also be a translucent seal placed on top of LED indicator hole to protect the internal electronics. One possible way to stack all of the various sub-systems together would be with pogo board bolts that run through routed holes on each layer. See Figure 1 below for the current envisioning of the device.

Initial Prototype Model

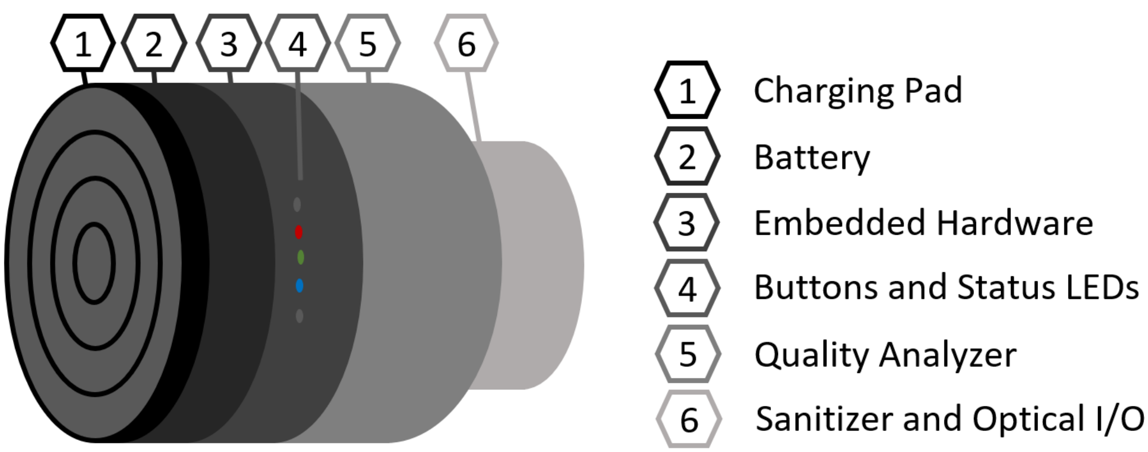


Figure 1: Initial Prototype Model

5 Block Diagrams

The following section will outline the various sub-systems of the device. Each subsystem will have individuals primary responsible for a block diagram and outline how other team members will contribute for each section. The Responsibility Legend seen in Figure 2 below should be followed (unless otherwise stated) order to determine which individual is responsible for a particular block in any of the diagrams in this section. This section also contains preliminary designs for the optical project requirements.

Responsibility Legend

Dean Pickett	Matthew Woodruff
Ryan Koons	Neeil Gandhi

Figure 2: Block Diagram Assignment

5.1 Water Quality Sensor

As shown in Figure 3, Matthew is the primary member responsible for the water quality sensor. Figure 4 details the currently proposed design for such a module.

Water Quality Sensor Block Diagram

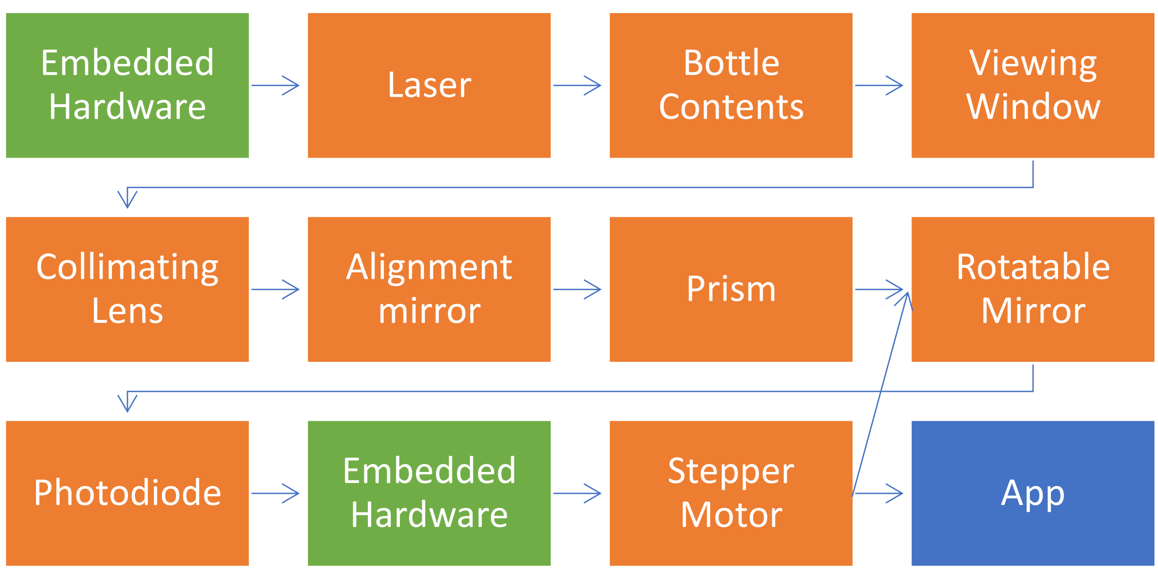


Figure 3: Water Quality Sensor Block Diagram

Compact Circular Spectrometer V3

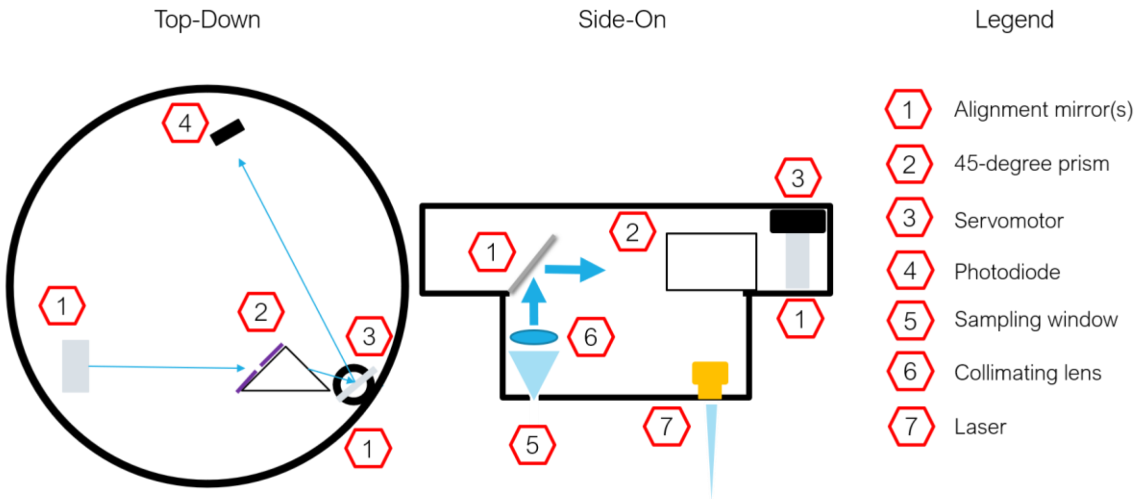


Figure 4: Compact Circular Spectrometer V3

5.2 Sanitization

As can be seen below in Figure 6, Neeil is primarily responsible for the sanitizer. See Figure 6 for a physical layout of such a device.

Sanitization Block Diagram

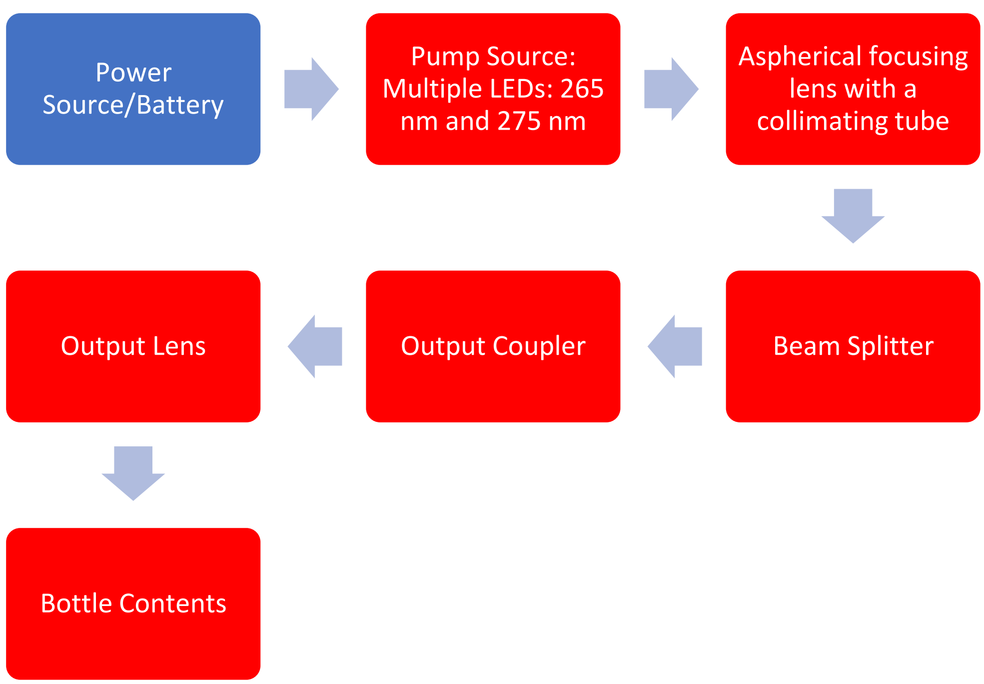


Figure 5: Sanitization Block Diagram

Proposed In-Line Optical Sanitizer Design

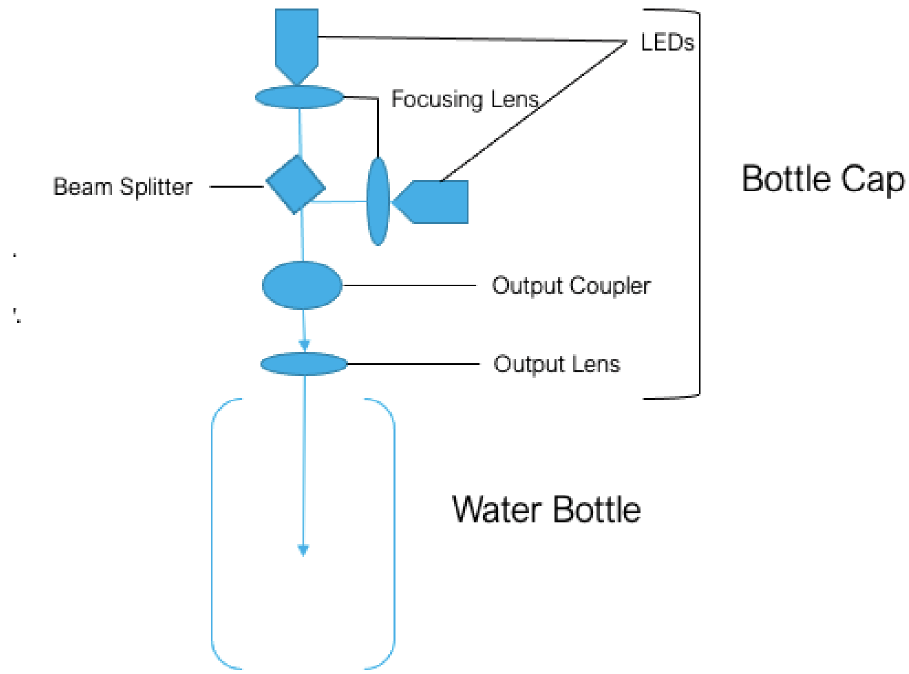


Figure 6: Proposed In-Line Optical Sanitizer Design

5.3 Device Power

Dean will be responsible for most of this section (see Figure 7 below), with some assistance from Ryan.

Device Power Block Diagram

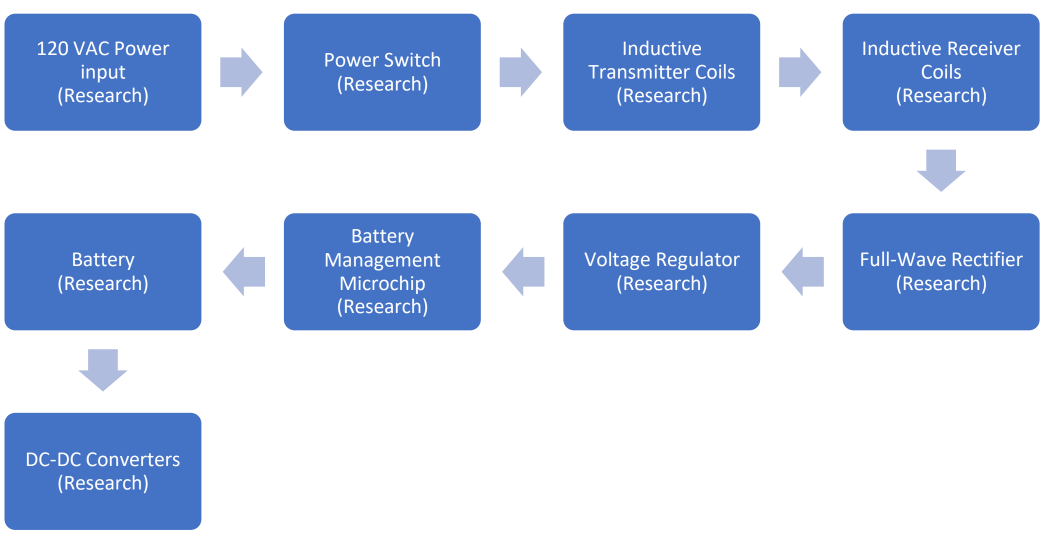


Figure 7: Device Power Block Diagram

5.4 Embedded Hardware

Ryan will mostly be responsible for this section (see Figure 8 below) with some assistance from Dean.

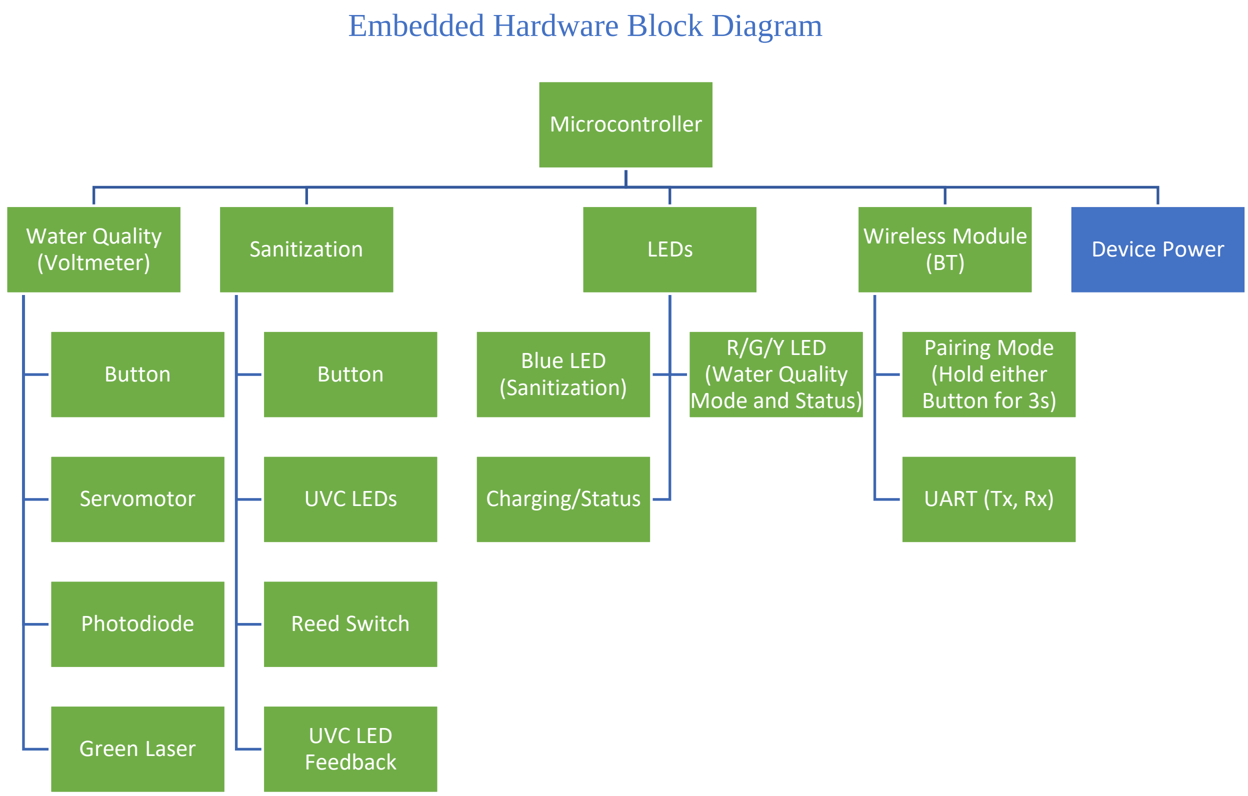


Figure 8: Embedded Hardware Block Diagram

5.5 Software

Ryan and Dean will work together on the software (see Figure 9 below) some assistance from Matthew. For this section blue blocks will represent shared tasks between Dean and Ryan.

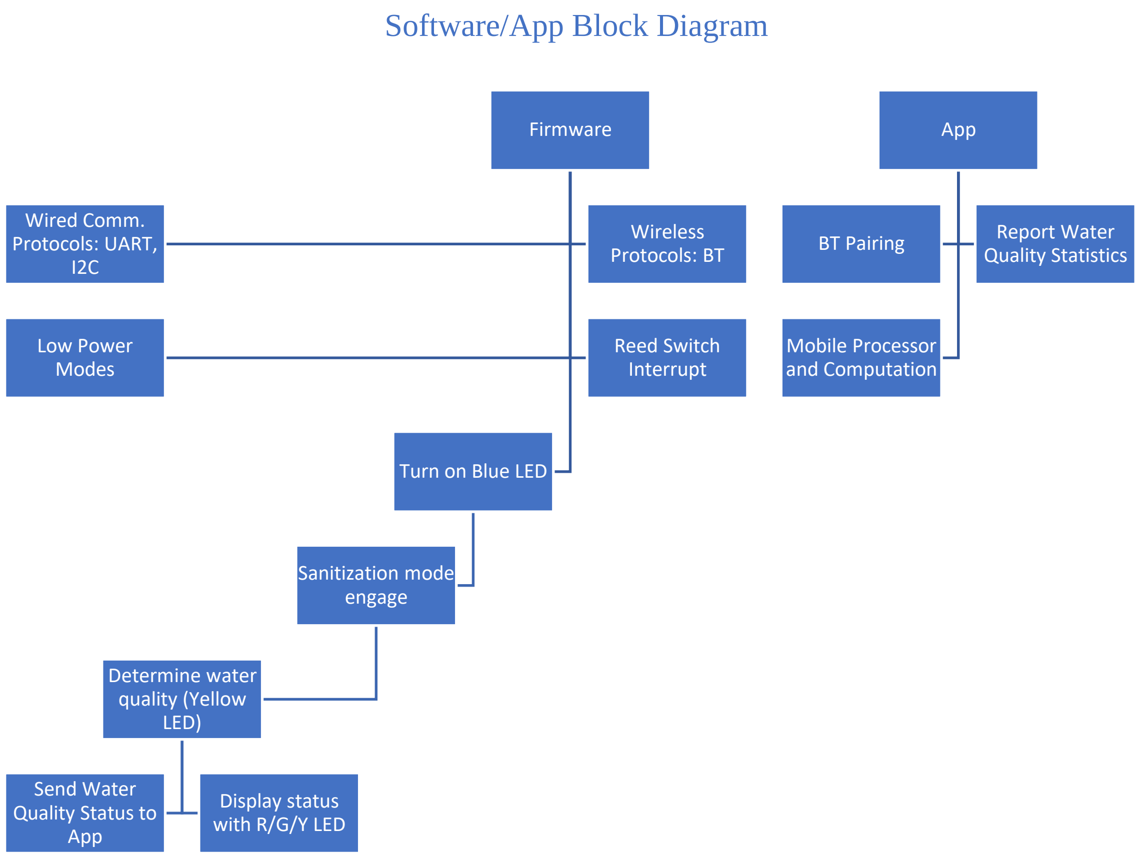


Figure 9: Software Block Diagram

6 Estimated Budget and Financing

Table 9 below outlines the estimated budget for prototype/development, as well as the typical cost per unit. Production cost (per unit) is an important parameter for this product's marketability, and therefore should be factored into all estimated financing.

Estimated Budget

Component	Prototype	Production
Stainless Steel Double Walled (Generic) Bottle and Lid	$15.00	$15.00
Quantity of Five: PCB, solder mask, and Assembly (PCBWay). Two separate board revisions ($100.00 per revision)	$200.00	$20.00
R/G/Y LED	$2.00	$2.00
Buttons (Sanitization, Water Quality, BT Pairing)	$2.00	$2.00
Blue LED	$0.50	$0.50
Reed Switch	$2.00	$2.00
Wireless module	$10.00	$10.00
Bluetooth evaluation board	$54.00	$0.00
UV Sanitizing Diode (includes coupler, source, and lenses)	$30.00	$30.00
MOSFETs	$3.00	$3.00
Raman Spectroscopy Sensor (Water Quality)	$60.00	$50.00
Li-ion/Li-poly charger	$6.00	$6.00
Qi Compliant Charging Receiver	$15.00	$15.00
Rechargeable Battery	$20.00	$20.00
Voltage Regulator Components ($0.60 x3)	$1.80	$1.80
Microcontroller (10-20 I/O)	$7.00	$7.00
SMT Passive Components	$5.00	$5.00
Approximate Total:	**$433.30**	**$189.30**

Table 9: Estimated Budget

7 Initial Project Milestones

This section outlines the project milestones and dates for both senior design sessions. The first table (see Table 10 below) provides a color key for deadline strictness while Table 11 provides an overview of the first semester of senior design and Table 12 estimates general deadlines for the second semester.

Legend

SD1 Hard Deadlines
Tentative deadlines

Table 10: Initial Project Milestones Legend

Senior Design 1 Milestones

Date	**Details**
5/21	Project Conceptual Brainstorming: Optics and Electrical Engineering
5/27	Project Idea Discussion
5/28	Project Selection: Decided on Smart Water Bottle
6/9	Divide & Conquer v1.0: Complete the D&C
6/11	Divide & Conquer v1.0 Due: Submit the completed D&C
6/15	D&C V1.0 Meeting with Dr. Richie
6/23	Divide & Conquer v2.0 Complete: Approximately 25 pages
6/25	Divide & Conquer v2.0 Due: Submit the final D&C
6/28	SD1 Paper: 30-page draft completed
7/2	SD1 Paper: 45-page draft completed
7/5	SD1 Paper: 60-page draft completed
7/9	SD1 v1 60-page paper Due
7/16	SD1 Paper: 80-page draft completed
7/23	SD1 v2 100-page paper Due
7/30	Finishing touches
8/3	Final document due
8/6	Order prototype boards (PCBWay)

Table 11: Senior Design 1 Milestones

Senior Design 2 Milestones

8/23	First day of Fall Semester goals
9/27	September goals -Build and test optical components
10/25	October goals- Work on software
11/24	Thanksgiving Wednesday- Have everything ready and work on testing and final paper and presentation
12/3	Last Day of Class goals – Presentation of the project
12/11	Last day of final exams – Submit the final report

Table 12: Senior Design 2 Milestones

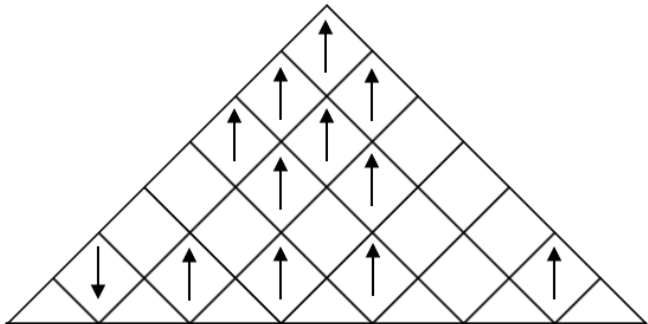
8 Engineering-Marketing Tradeoff

All engineering projects require careful consideration and balance between engineering requirements and marketing requirements. With a focus too large on the engineering requirements, the product may become too expensive or difficult to market. Likewise, a product heavily focused on marketability instead of engineering, may not function correctly or function altogether poorly. Thus, it is important during the early stages of this project's inception to consider all of the various types or requirements and their interaction with one another. The right balance between device requirements will have to be determined such that both types of requirements are sufficiently satisfied. The following diagram below outlines the relationship between engineering and marketing requirements.

Each column of the House of Quality below represents an engineering requirement for the project. Each row represents a marketing requirement for the project. The "Target Specifications" at the bottom of Figure 10 is the value or performance rating related to each engineering requirement. A '+' in the gray section indicates a requirement that should try to be maximized. Whereas a '-' in the gray section means that a requirement should try to be minimized. The up and down arrows represent the positive or negative correlation (respectively) between each type of requirement. Any comparison that has multiple arrows indicates a stronger correlation between the requirements.

The roof of the house of quality illustrates the relationship between the engineering requirements. This is accomplished by providing a cell for each possible pair of columns and placing an indicator to show whether there is a direct, inverse, or no relation between the factors an upwards-oriented arrow indicates that the two columns share a direct relationship--increasing one column will increase the other. To determine the relationship between two requirements, follow each cell up and diagonally left until they intersect at one point. A simple example of this would be that battery capacity and cost are directly related- a large battery will almost universally cost more than a smaller one (this does not imply, however, that increasing the cost will increase the battery capacity). A downwards-oriented arrow indicates that the two columns share an inverse relation- increasing one column will decrease the other. An example of this would be cost and size as decreasing the overall size of a device, while potentially reducing the total amount of materials needed, tends to require more demanding designing and compact components. An empty cell indicates that the two columns have no relationship with each other.

House of Quality



Engineering Requirements / Customer Requirements		Cost	Dimensions	Efficiency	Charge Time	Battery Capacity	Laser Intensity	Sensor Sensitivity
		-	-	+	-	+	+	+
Affordability	-	↑↑	↓	↓	↓	↓	↓	↓
Sterility	+	↓	↓	↓			↑↑	
Accessibility	+	↓	↑		↑	↑	↑	
Safety	+	↓		↑	↓	↓↓	↓↓	↑
Sanitization Rapidity	-	↓		↓				↓
Charging Accessibility		↓	↓↓	↓↓	↓↓			
Data Availability	+	↓	↓	↓				↑↑
Target Specifications		< $1,000	Round 3" x 4"	>30%	<8 hours	4,400 mAh	>1,000 mA	*See Table 1

Figure 10: House of Quality

9 Technology Investigation for Project Development

9.1 Device Power

9.1.1 Inductive Charging

The simplest way to implement a wireless charging system is to use technology that relies on Faraday's Law to create a changing magnetic field in one coil, which in turn induces an AC voltage in a magnetically coupled coil nearby. Reducing the cost of the final device is one of the main engineering and marketing goals of this project, so it is strongly recommended to implement a wireless charging system that does not require a harmonic oscillator in the power transmitter. The electric power that is available through a standard household, single-phase socket already provides an alternating current at 60 Hertz that can be used for simple inductive charging.

There are three physical constraints that need to be addressed in order to implement inductive charging well. All of them involve the coupling of the transmitter and receiver coils [6]. The first constraint for this type of charging is the distance between the two coils being used. In order to maximize the magnetic coupling between the transmitter and the receiver, it is necessary to minimize the distance between concentrically aligned coils. Part of this issue can be taken care of by adding permanent magnets to the casing of the transmitter and receiver so that they are as close as physically possible during the charging process, but it is also possible to implement some simple clip that would keep the power transmitter in place. The other feature within our control would be to minimize the thickness of the casing for the transmitter and receiver. Ideally, this distance between coils would be less than 10 millimeters [6]. Along these same lines, it is necessary for the transmitter and receiver to actually be aligned concentrically in order to minimize the magnetic flux leakage between coils. This problem can also be dealt with either by implementing some sort of clip-on device or by using permanent magnets fixed to the inside of the casing for the water bottle top and the power transmitter.

The third physical constraint for using inductive wireless charging is needing to minimize the surface area of conductive material between the two coils. Eddy currents are created in any conductive surface that has a magnetic flux passing through is perpendicularly. Adding permanent magnets into the centers of the transmitter and receiver coils will inevitably create eddy current losses, which will not only reduce the overall efficiency of the charging system but will also be a considerable source of heat depending on the actual magnets used. These eddy currents could also be induced in the casing of either the bottle cap top or the power transmitter if either were made of some conductive material such as metal [7]. There is no advantage to using metal casing when compared to plastic, so this choice in material would be both cheap and beneficial to increasing the efficiency of the wireless charging. On this topic of eddy current losses, it is still up for consideration as to whether or not it would be best to implement permanent magnets into the charger design. Although we should easily be able to charge a 4,400 mAh battery within eight hours despite the losses in the magnets, it is very important to minimize how much heat is being produced within the water bottle cap since the intention is for the casing to eventually be sealed so as not to allow water ingress into the electronics.

The current standard in the market for inductive charging is called Qi (pronounced "Chee") and this is specified by the Wireless Power Consortium [8]. Having a device that is Qi certified is much more complicated than simply sending and receiving power through electromagnetic induction, because this standard also specifies a communication protocol for data transfer between transmitter and receiver. This communication is used to provide feedback to the power transmitter (or base-station) about information such as the state-of-charge of the battery. Providing a means for this communication is a very important concern for charging a battery, because the worst-case scenario for continuing to push charge into a fully charged battery is starting a fire. There are premade Qi compliant power transmitters and receivers, which run for over $40 for the pair. This is much more expensive than the initial estimates for wireless charging modules but having a Qi compliant transmitter and receiver would make the final product much safer as well as allow for USB compatibility for the power transmitter.

9.1.2 Magnetic Resonance Charging

The newer form of wireless charging technology that is available today relies on the concept of magnetic resonance. This design of wireless power transmission is still possible due to the electromagnetic induction of an AC voltage in a coil, but the main difference is that tuning the transmitter and receiver to oscillate at the same resonant frequency strongly supports the magnetic coupling between the two coils. In other words, the transmitter and receiver do not suffer so greatly from the coils being either misaligned concentrically or separated by distances much greater than those in which simple inductive charging can operate [6]. This feature is not thought to be as useful for the purposes of this project, because the intention is to have a power transmitter that sits on top of the water bottle and easily keep coils aligned and nearby. It is also important to note that even though this technology allows for coils to be further apart and misaligned that it still does not achieve efficiency levels as high as traditional inductive charging due to the flux leakage between coils. Although it is possible to design a circuit to use the power of magnetic resonance while still keeping the coils tightly coupled in space, the added drawbacks for this sort of design are that the oscillating circuits are more complex and that these resonant frequencies are typically much higher than those used in standard inductive charging. Introducing higher frequency power signals has the possibility of creating electromagnetic interference in nearby devices. For the purposes of this project, it seems inadvisable to utilize this more advanced and newer technology.

9.1.3 Battery

Upon investigating various popular vendors online, it has become clear that the choice in battery chemical composition for the needs of this project is much more dependent upon availability than the specific differences in performance. The majority of battery packs that do not exceed three inches in any dimension are either lithium-ion (Li-ion) or lithium-polymer (Li-po) based. The handful of Nickel-Cadmium batteries that were found online were larger and heavier than the lithium-based counterparts of equivalent capacity as well as being more expensive in terms of charge capacity per dollar. Bearing this in mind, the two main aspects of design focus for the battery will be the rated charge capacity and safety circuitry. The options for Li-ion and Li-po battery packs that have been investigated so far typically are rated at 3.7V/4.2V, which means that the nominal rated voltage is 3.7 volts, and the maximum voltage of the battery is 4.2 volts [9]. It is typical that batteries will have a range of voltages that depends on what percentage of charge it has at the time, and this has many consequences in terms of additional circuitry that is needed to properly charge and discharge the units. One such consideration is designing voltage regulators to specifically meet the voltage and power requirements of each instrument in the project. Another factor that is more directly related to the battery itself is the need for special circuitry in ensure safely charging the battery packs. Having batteries with a denser energy content is naturally going to make them more dangerous, which is why so much care must be taken to ensure safe implementation in this project.

Monitoring the voltage and current flowing in or out of a Li-ion/Li-po battery cell is typically handled by protection circuitry that is included in the battery itself [9]. However, even with these safety features installed in the battery packs, it is still highly recommended to use a charger that is specifically designed for Li-ion/Li-po batteries of the specified voltage. This additional circuitry, which will need to be purchased separately, is designed to help control the flow of current into the battery as it is charged. There are many nuances to charging these lithium-based batteries in a way that is safe, fast, and prolongs the life of the battery by not damaging it [10]. The main takeaway from this technology research is that the additional cost of professionally designed and manufactured safety circuitry seems to be well worth the price, especially when considering the consequences of a Li-ion/Li-po battery that has been mishandled.

One final consideration on the topic of battery safety is monitoring the temperature of the cells. The major causes of batteries getting too hot that can be controlled is due to improper charging and discharging [10]. However, the intention of this project is to build a unit that does not allow the flow of air and water into the casing that contains the electronic circuitry and therefore it is not unreasonable to assume that the temperature of the battery pack could reach unacceptable levels due to a lack of proper heat dissipation during operation. A significant production cost of this project is already forecasted to consist of pre-made units to ensure proper wireless charging of the internal battery, but for the sake of safety and completion, it might be worthwhile to also investigate the potential implementation of a temperature monitoring system.

9.1.4 Supply Voltage Regulators

Regardless of what the actual voltage needs are for any of the components that will be included in this project, they will all require a consistent supply voltage that does not change as the battery changes temperature or its level of charge. There are prefabricated options available on the market as well as online tools for assisting in developing voltage regulators that can be implemented in the printed circuit board and both of these options will be explored.

9.1.4.1 LDO Linear Regulators

Low-Dropout (LDO) linear regulators are devices that are built specifically to provide voltage regulation for output voltages that are close in magnitude to the input voltage being provided [11]. Linear regulators do not rely on switching and therefore produce less noise in their output than the typical switching voltage regulators. However, the drawback of a linear regulator is that there is a minimum threshold voltage drop that must exist between the input and output in order to remain stable [11]. This minimum voltage drop that must be established over the internal transistor of a linear regulator is known as the dropout voltage; this can be a liability for a regulator when the input voltage is constantly changing, due to the nature of a battery, and the required output voltage is near the potential range of voltages that the battery provides. Not all linear regulators are designed to minimize the dropout voltage, but this is seen to be a highly desirable trait since it is currently not clear what actual output voltages will need to be maintained.

9.1.4.2 Switching Voltage Regulators

There are several different topologies for switch-mode DC-DC converters that can either be used for stepping up and/or stepping down voltages. One of the main advantages of these switching voltage regulators is the relatively low amount of power that is lost in the switching components, which translates to having a device with a very high efficiency [12]. This higher efficiency is desirable, as it is one of the engineering requirements that has been outlined in the project description. There is a convenient online tool called WEBENCH Power Designer that is made available by Texas Instruments; this was how the switch-mode DC-DC converters that were required in the pre-requisites for this class were created. Using this online tool will allow for easily selecting between different designs based upon the total cost of components, switching frequency, design footprint, and many other variables of interest when designing a regulator to be implemented on a custom PCB.

It is worth specifically mentioning that it is highly desirable to minimize the Electromagnetic Interference (EMI) that will be created from implementing switch-mode voltage regulators due to the fact that we will be using Bluetooth wireless communication on the same board. Designing these types of DC-DC regulators with the added circuitry for filtering switching noise that causes EMI will definitely make this the more expensive option when compared to the linear voltage regulators. It may not even be necessary to include additional filtering circuitry within the power system design due to the difference in frequency of the switching when compared to the frequency of the switching regulators. When some preemptive designs were created on WEBENCH, it was clear that the highest frequency of switching was in the 1-2 MHz range, while other options had switching frequencies closer to 500 kHz. Since the frequency used for Bluetooth communications is 2.4 GHz [13], the difference in these frequencies is at least four orders of magnitude. Since higher order harmonics of a signal occur at each integer multiple of the fundamental frequency, it can be implied that any potential harmonics of the switching regulators that comes close to 2.4 GHz will be negligible due to this difference in frequency magnitudes. Regardless of the potential differences in price, they are not so large as to offset the benefit of having much higher efficiency.

Not only does a higher efficiency mean longer battery life of the device, but it also means less heat being generated within the device. None of the members of this team have any formal experience in properly designing a system by calculating the maximum heat output, the heat dissipation, and then comparing these figures to the maximum safe temperature of the electronic components. Therefore, it seems to be prudent that every possible effort be made to minimize the heat losses during power transmission and supply because the bottle cap will be sealed, so as to prevent the ingress of moisture. Including a Lithium battery within this confined space will mean that every effort must be made to ensure that it does not get too hot either during charging or discharging.

9.1.4.3 Current Regulating Diodes

This project already appears to be requiring a lot of various regulated voltages. It is also clear that the microcontroller must be regulated to 3.0 V for device operation; however, there may be other options for regulating device components. One option would be implementing current regulating diodes (CRDs) for the optical components. They are also known as current limiting diodes or constant current diodes. This regulating circuit is typically composed of a JFET and is rated for a certain amount of current. Current regulating diodes that match the desired V-I graph for the optical diodes could be purchased and implemented into this project. This would require fewer regulators on the device, which overall would decrease the cost of the project. As long as the supply voltage is high enough to support the voltage drop of each optical component (with some leeway), this would essentially create miniature regulators for each optical diode to achieve their own regulated current and forward voltage. However, they technically place a maximal current on the node; in other words, a current below the rating could still occur based on the voltage being supplied to the diode.

9.2 Sanitization

How do you feel when you can see, taste, or smell a contaminant? It is bad, right? What do you think of the water that smells, looks, and tastes just fine? Is it good enough to drink? The answer to this is -- not necessarily. The water can be contaminated without having an odd flavor.

Microbial and organic substances cannot generally be recognized by human senses and can lead to severe health issues down the road- for microbes in particular, the affect is generally gradual enough as to allow for significant transmission prior to diagnosis. Water may contain contaminants from pesticide or compost application. These chemicals from pesticides and manures in “water may increase cancer risk and reproductive problems, and can impair eye, liver, kidney, and other body functions” [14].

The goal of this bottle will be to sanitize the water in the bottle by killing the microbiological contaminants present in it. “The transmission of diseases such as typhoid and paratyphoid fevers, cholera, salmonellosis, and shigellosis can be controlled with treatments that substantially reduce the total number of viable microorganisms in the water" and that is what we are looking for here [2].

The goal of the sanitization component is to disinfect the tap water by elimination of microorganisms which are responsible for various waterborne diseases. There are various kinds of methods for the sanitization of water, the most common and widely used are chlorination, boiling and UVC light.

Choosing the method for the disinfection of water depends on various factors. These include:

- how effective it is in eliminating the microorganisms (bacteria, protozoa, viruses, and helminths).
- how reliable and accurate of the process and the way it can be controlled and monitored.
- whether the method leaves some residuals behind and how would that affect the disinfection process and be taken care of.
- how purified the water gets after the process is complete; and
- how accessible the technology is for the public water supplies [2].

Now we will look into different sanitization methods used in the industry and compare and chose the most feasible for our project.

9.2.1 Sanitization Methods

There are different kinds of methods used throughout the water irrigation industry for cleaning and purifying the water. The most common and widely used methods are boiling and chemical treatments. Ultraviolet light industry is rapidly growing in today's world. The methods depend a lot on what scale the industry and purifying is happening and then the methods are considered. Some big industries use one, two or all the three methods for sanitizing the water. In the following section we will discuss more about each water treatments and compare them to get the best method for our design project.

9.2.1.1 Boiling System

This is one of the most common method used in a lot of areas throughout the world let it be how remote it is. In this method the source does not depend as it is ensured that even regardless of the source the water kills the pathogens on boiling. Boiling time of water depends on the altitude. Higher altitudes require longer time and lower altitudes require shorter time. "Water should be at a rolling boil for 1 minute and at altitudes greater than 6,564 feet, boil water for 3 minutes" [15]. This method has great advantages such as the source does not matter and requires a short amount of time to boil the water. But, for this project this method to kill the contaminants will not be ideal. As our product is a water bottle it would be difficult to incorporate the electronic system as it will require high voltage and hence will need a power source outside where it can be plugged to. The temperature is going to rise due to boiling and hence the material the bottle will be made of will have to be taken care of as the material should not disintegrate or mold at high temperatures. The heating will also result in the water to be too hot to drink and might also heat up the outer casing of the bottle for the user to hold it. We can add a cooling system to cool the water after the boiling process is completed but it will require more electronics. Even though these problems can be overcome it would not be the most suitable for the water bottle.

9.2.1.2 Chemical Treatments

There are several chemicals that can be used to disinfect the water. The most commonly used chemicals are chlorine, iodine, bromine, and oxidizing agents. Chlorination is a widely used process in big factories, but in a project like this it has various drawbacks as it leaves residuals behind, which can be dangerous when consumed. Other chemical like Iodine has a better efficiency but it is tough to store Iodine as it deteriorates in sunlight. It also leaves a bad metallic taste behind. Bromine has its drawbacks in storing and transporting too as it highly reactive. Bromine's major drawback is its reactivity with ammonia or the other amines that affect its treatment of disinfecting the water. Oxidizing agents have great efficacy but due to its high cost and desired pH concentration it is not really viable for us to use it. All these being chemicals they can react and act differently with different foreign viruses and hence will not give the accuracy we need. This method is easily accessible and does not require any electrical components, but they have various drawbacks such as pH of water, temperature of water, it leaves residuals, leaves an after taste, and takes a longer time for disinfecting the water.

9.2.1.3 Ultraviolet Light

Ultraviolet (UV) light means "beyond violet" coming from the Latin word "ultra". Naturally, UV light comes from the sun. UV light spectrum lies between visible light and x-rays. The discovery of UV light was done in 1801 by a German physicist Johann Wilhelm Ritter. He observed in his experiment that the light rays beyond violet light darkened silver slats [16]. Ultraviolet light has an electromagnetic spectrum ranging from 10 nm to 400 nm with corresponding frequency ranging from 750 THz to 30 PHz as shown in Figure 11 (see below). These wavelengths are too short for the human eyes to see.

Ultraviolet light is broken into four categories:

- Ultraviolet A (UVA) – also known as Long Wave/ Near UV as it has longer wavelength and is closer to the violet light. The wavelength ranges from 315 nm to 400 nm. This light is not blocked by the ozone layer.

- Ultraviolet B (UVB) – this is also known as the medium wave UV. It has a wavelength between 280 nm and 315 nm. This light not completely blocked by the ozone layer but most of it is absorbed by it.

- Ultraviolet C (UVC) – known as short wave UV or commonly referred to as germicidal. This light is completely absorbed by the ozone layer. UVC has wavelength between 100 nm to 280 nm.

- Vacuum Ultraviolet – this has a spectrum from 10 nm to 100 nm. They are absorbed by the nitrogen in the atmosphere and hence cannot penetrate through.

Electromagnetic Spectrum showing the wavelengths of sub parts of UV Light

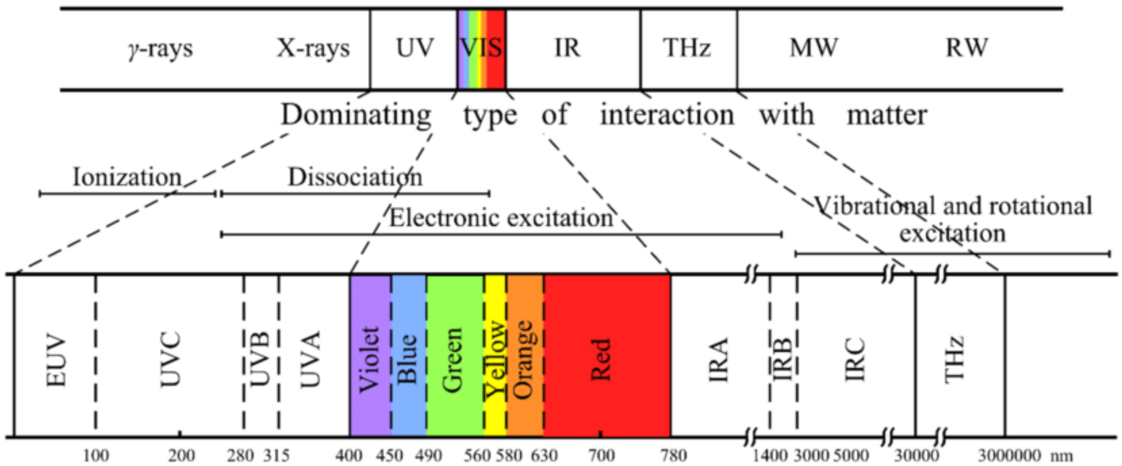


Figure 11: Electromagnetic Spectrum Figure taken from [17]

Pathogens like bacteria have genetic information coded like DNA and RNA similar to that of the cells in our body. We all know the steps for the central dogma of biology: DNA to mRNA to Proteins. If the process is interrupted at any time the cell dies. “DNA is like a blueprint; a little alteration can greatly affect the intended structure and cause a collapse in the entire cell” [18]. All viruses contain either DNA or RNA.

Sufficient intensity of any radiation has the possibility to kill. For example, when there is an atom bomb blast, the gamma radiation has the capability to vaporize an entire person within a few hundred feet but occasional photons of gamma radiation, which are experienced by all the passengers traveling by air causes no damage [19].

Now we know that occasional photons of gamma radiation can damage the DNA and RNA. When the damage done is minimal, cells have the ability to self-repair and recover. “The persistent radiation- induced changes in DNA/RNA may express themselves as cancer” [19]. When the radiation is strong then it completely damages the DNA/RNA, and the cell dies.

Similar to gamma radiation, UV light with appropriate wavelength and intensity can destroy the cells and viruses. DNA and RNA are particularly more sensitive to UVC light. The efficiency of destroying the microorganisms depends mainly on the distance, wavelength, intensity, and duration of exposure. “A standard UVC 270 nm LED fixture can kill most microbes within a six-inch radius in 10 seconds” [18]. DNA and RNA breakdown occurs due to ablation of the fatty acid molecules comprising the source code of life, as the high energy photons break off pieces. This molecular rearrangement of the biological components in the microorganisms causes inactivation of organisms functioning and hence kills it without leaving any residuals. The common pathogens found in ground water are Giardia, Cryptosporidium, and bacterial pathogens like E. Coli. UV radiations effectively inactivate these pathogens by interfering in their genetic information.

9.2.1.4 Methods Comparison

It is clear after looking into all three sanitization methods from Table 13 (see below) that ultraviolet light is the best approach to go forward with, due to its everyday use and portability. Boiling system and the chemical treatments are not feasible methods for our product due to its lack of portability and everyday use as our audience is travelers and hikers. UV light requires more safety precautions due to its ability to harm humans with potential exposure but that is just a minor issue and can be taken care of. Seeing the table and detailing everything we feel it will be easier and a better approach for our project to go with UV light.

Sanitization Methods Comparison

	Boiling	**Chemical Treatment**	**Ultraviolet Light**
Effectiveness	Very High	Very High	99.9% pure
Time	1-3 min	15-20 min	10 sec – 2min
Cost	<$25	<$20	<$30
Residual Left	None	Yes	None
Electric Components feasibility	Not feasible due to heating and cooling	No electronic components required	Feasible
Implementation	Hard	Hard	Medium
Power	High	Low	Medium
Safety	Medium- can get too heated	Medium – because of the residuals left	Medium – UV light can be harmful for humans
User friendly	Low – water gets too heated	Low – User needs to keep refilling the chemicals	High – no user interaction needed
Portability	Hard	Medium	Easy

Table 13: Sanitization Methods Comparison

9.2.2 Safety and Precautions of Ultraviolet Light

We all know and have heard about UV light not being good for our health in the long run and it is completely true. Most of the UV light is blocked by the ozone layer keeping us safe but now as the industry has started building UVC lights for its usefulness for killing microorganisms we need to be careful and take the necessary precautions. Following are the risks caused by UVC radiation and they all depend on the wavelength, duration of exposure and dose:

- It affects the skin and eye and can cause painful eye injuries and skin burns.
- Some UVC lamps generate ozone. Ozone inhalation can be irritating to the airway" [20]. These ozone gases can cause several health complications like coughing, chest pain, lung damage, throat irritations, asthma, and more.
- As we know it interrupts the DNA and RNA in the cells and same could happen in human cells and cause many health issues. If UV light is not exposed correctly, it might just modify the virus and hence causing mutations of the virus that might be difficult to kill later.
- "High-exposure to radiation promotes the formation of cancerous tumors that can prove fatal if not detected in time" [18].

Therefore, proper safety precautions are very necessary when using a UVC light. Do not come in direct exposure to it and never look right into the path of it.

For these safety concerns we will have a sensor in the bottle that will not let the UVC light radiate when the cap is open for the safety purpose of the user. UVC light will only work when the bottle cap is sealed.

9.2.3 Different types of UV Sources

Now as our group has decided that the most feasible method for our project is UV light, lets dive deeper into it. UV light has many different sources, and we need to decide on the best one for our project.

All UVC lamps are not the same. Some lamps may emit specific wavelengths of UV light (254 nm or 222nm), and some may emit broad spectrum of UV wavelengths. These lamps may be combined with emitting visible and infrared radiations too. The emitted wavelengths play a vital role in the effectiveness of the lamps and the health and safety risks associated with it. "There is some evidence that excimer lamps, with peak wavelength of 222-nm may cause less damage to the skin, eyes, and DNA than the 254 nm wavelength, but long-term safety data is lacking" [20]. Following are the different kinds of lamps used to emit UV light:

- Low-pressure mercury lamp: These are the most common lamps which are used in most of the water irrigation systems to disinfect the pathogens. Its main emission is at a specific wavelength of 254 nm. They can also produce other wavelengths, but this is the main one for UVC disinfection.

- Excimer lamp or Far-UVC lamp: These lamps are close to the infrared radiation and has a peak wavelength at 222 nm.

- Pulsed xenon lamps: These lamps are mainly used in hospital settings to disinfect the air in the operating rooms or other spaces. They emit short pulse of broad spectrum which includes UV infrared and visible. These lamps can cause many health hazards and for safety purposes are mainly used when there are no humans in the desired space.

- Light-emitting Diodes (LEDs): Light-emitting diodes emitting UV light are booming in the industry and are becoming very common in the industry and commercial use. They are getting commonly available and hence the demand is increasing. LEDs emits very narrow band of spectrum. There are multiple LEDs available in the market right now with different peak wavelengths like 265 nm, 273 nm, 280 nm, and others.

Comparison between UV LED and UV Lamp

Originally most of the disinfection systems for water to kill the microorganisms used UV lamps. UV lamps had a single wavelength light emitting at 254nm. Due to the increase in the technology most of the companies were looking into polychromatic lamps (meaning the light is emitted in multiple wavelengths) due to its better efficiency in killing the pathogens. Therefore, after referring to Table 14 (see below), the most feasible out of these options due to our design's portability, size, energy, and cost constraint would be using UV LEDs.

LED Comparison

UV LED	**Section**	**UV Lamp**
New, Light, Simple, Compact, Small	Technology	Old, Bulky, Heavy, Complex, Large
10,000 – 50,000 Hour	Lifetime	2,000-10,000 Hour
Low	Energy Consumption	High
Zero	Warm-up time	Slow
No mercury, No Ozone	Environmentally Friendly	Mercury used, Ozone generation
Low	Heat Generation	High
Single UV Band, Customizable	Emission Wavelength	Multiple Peaks
None	Heavy Metal	Mercury (20 - 200 mg)

Table 14: Comparison between UV LED and UV Lamp. Referenced from 21.

9.2.4 LED

LED is a semiconductor device most commonly constructed out of a single P-N junction. It emits light when current is passed through it. The basic principle on which LED is works is that of a semiconductor. LEDs are made of p-type semiconductor placed on the top and a n-type semiconductor placed on the bottom. The lack and excess of electrons in the materials creates junction between them where electrons are passed when there is a current flow. When forward bias voltage is applied, holes and electrons move. Electrons from the n-type material are pushed to p-type material and holes from p-type material are pushed to the n-type material. At the depletion region or the junction between the two materials these holes and electrons combine and hence produce quantum energy due to the radiative recombination. This quantum energy can produce other energy like heat too, but with proper schematic of the semiconductor one can design it in a way to emit certain wavelength of light. Therefore, the material, quantum wells and dislocation density play a vital role in the semiconductors to emit specific wavelengths.

LEDs are available in many different wavelengths. They were originally used in infrared spectrum and then into visible light starting with red and green and then to blue and white. To produce LEDs in UV light was very difficult in historic times, but now we can see the tremendous growth in it due to the newer technology.

"The first UV-C LED devices were developed primarily in Japan, Korea and the US, as extensions to LED devices emitting in the blue and then UV-A and UV-B wavelength region" [22]. UV-LED are made from III-V semiconductors materials. The ratios of atoms from both the materials matters a lot. They are precisely chosen to fulfil the needs. The atomic number plays role in the energy bandgap and this in turn determines the wavelength emitted by the material. By alloying the material, we can alter and change the bandgap and hence the wavelength, which is very beneficial in the semiconductor industry. The commercial LEDs present in the market are mostly made of sapphire substrate with buffer layers of aluminum nitride (AlN) or aluminum gallium nitride (AlGaN). These films have more defects. "Any defect in the crystal structure is a point where electrical carriers can recombine in a way that does not produce the desired wavelength and, thus, the efficiency is reduced" [23]. Growing and using Aluminum Nitride substrate is defect free and hence more reliable for UV LEDs. The shortest wavelength that was emitted using aluminum nitride substrate with very little dopant was 210 nm. Therefore, most commercial UV LEDs are made of AlN and AlGaN with some dopants. The efficiency decreases as the emitted wavelength decreases due to the material lability. These LEDs are very reliant and therefore benefits our project. "A test of 170 UV-C-LEDs fabricated over the course of six months showed a median degradation of less than 4% after 1,000 hours of continuous operation at 100 mA," this shows the durability too, which is very important for our design [23]. To choose the correct LEDs is a tough task for our project design as we want it to be cost efficient. Most of the UV LEDs are expensive as they are very difficult to make. AlGaN is also tough to make and hence there are only a few reliable and efficient semiconductor industries making these materials efficiently. The special ceramic packing, quartz window and the chip used makes it more complicated and expensive. The chip's structure makes it difficult to optimize it. Therefore, selecting the LEDs to be implemented in our project requires a lot of research and finding.

9.2.5 Technology in the Industry

Water is everything for human beings. Water keeps us hydrated, which results in a better metabolism and better skin. Recently with the decrease in the amount of natural clean drinkable water, water purification industry is at its rise to provide clean and safe drinking water for every individual all around the world. Also, with the covid-19 pandemic, UV-C light technology has become more popular and boomed as it kills the viruses and bacteria.

There are different technologies being implemented everywhere for sanitizing the water and air. All these technologies are great and have been proven to be very efficient. Figure 12 shows some of the technologies used in the water disinfecting industry to purify the water. The first one shows a filter and purifier attached to the plumbing, so the water passes through it and gets cleaned and is then transferred to the output source using chlorination and filters. The second image showing Pearlaqua is a UV-C LED purifier which is small and compact so it can fit in anywhere and can have multiple use. It has different features like intensity monitoring and temperature independent, which are useful to know about when sanitizing water using UV-C. It could be placed between any two pipes and can clean the water as it is passed through it. The third image shows Acuva, which is a large container with multiple UV-C LEDs. It is also connected to the pipes and water is allowed to let flow through it to purify it. Major big industries have everything combined like chemical treatment, boiling and UV sanitization to get purified water. They work on large samples and processes and hence need that, but for our households and daily use these small systems have proven to be very reliable and efficient for healthy drinking water.

Commercially Available Sanitizers

Figure 12. Showing some of the examples of water disinfectant in the industry. Figure taken from [22].

Main differences for our household and everyday use of water filters and sanitizers from the industry are implementation and use of different filters and UVC intensity and efficiency. There can be purifiers with multiple LEDs or single LEDs. Both are very efficient but have their pros and cons. Figure 13 shows a LED array sanitizer used in industrial settings. This device, produced by the Universal Science company, is called 'SpecialRED.' It has a strip of LEDs on the side and a slot in which a clear bottle can be placed in the handle or other things you want to disinfect, and it will illuminate the light and sanitize them. Other devices consist of multiple UV LEDs, and we can just place a bottle on the top and it can sanitize the water from different angles. The purpose of using multiple LEDs is for greater intensity of light, which results in more efficiency. Also, if one LED fails and does not output enough intensity there are other LEDs to take care of it and still purify the water. The disadvantages are it uses a lot of power and hence more electronics and higher voltages battery to supply enough current and voltage, requires a bigger space therefore not handy and easily portable and carried to different places and gets more expensive as it uses multiple LEDs, which are not cheap as discussed in the LED section.

Commercial LED Sanitizer

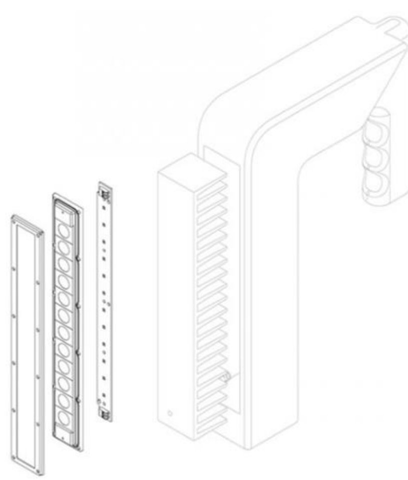


Figure 13: SpecialRED Sanitizer. Figure taken from [24].

Figure 14 shows a single semiconductor LED in different packaging styles. Part (a) is TO packaging which was used in the earlier days before the semiconductor (SMD) packaging was invented, (b) shows typical semiconductor packaging LED, which is mostly used in today's industry and (c) shows how the SMD LED can be incorporated on a PCB board on the circuit to be used. These LEDs can be placed anywhere in the system and does not require many electronics or optics and hence it is easily compatible to any use. They are very small and easily accessible. They are comparatively cheap as they are single piece. Mostly all the UV-C water bottles in the industry right now use the PCB LED and these bottles have proven to be very efficient with their effectiveness in sanitizing the water. There are some disadvantages to it like if the LED does not produce enough intensity, then the water is not sanitized completely and could leave to false interpretation, therefore the system needs to take that into account too. Choosing the LED for the system needs different analysis and we need to check the specifications and the datasheet correctly as the LED needs to have enough output power to purify the water, with appropriate wavelength, as not all the LEDs can have the required specifications needed for our specific design.

Commercially Available LEDs

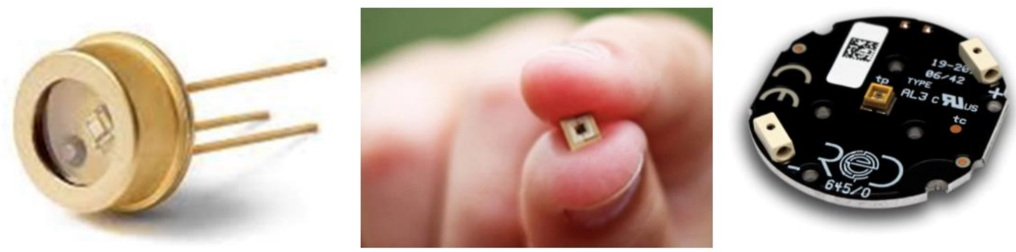


Figure 14. Single LED.

(a) LED with TO packaging, (b) Semiconductor packaging LED. Figure (a) and (b) taken from [22] (b) LED placed on the PCB board to be used in the system. Figure taken from [24]

Therefore, after the analysis, our group chose to go with using 2 single LEDs as it is cost efficient, small to be fit anywhere like the bottle cap, easy to implement and does not require a lot of electrical power to function them and hence keeping the system small and compact. This would fit perfectly with our design and the requirements for the project.

9.2.6 Tunable UV spectrum and it's efficiency

Most of the UV light emitted by the UV lamps and LEDs used for water purification are single wavelength. After increasing demand of UV and technology, the interest in UV lamps increased for polychromatic lamps, meaning the wavelength is emitted in different wavelengths. In 2011 as a part of Water Research Foundation, National Institute of Standards and Technology (NIST) was asked to help build a tunable UV laser for irradiating water samples [**Error! Reference source not found.**]. NIST build a tunable laser of wavelength ranging from 210 nm to 300 nm. The foundation did not want to use lamps as they have broad bandwidths, which causes difficulty in testing the specific wavelengths efficiency on different pathogens. Even using the filters did not help that much and still unwanted frequencies were passed through and acted on the sample. Therefore, NIST made a tunable laser with very narrow bandwidth of about 1 nm so it could be tuned with precision to check the efficiency of each wavelength in the spectrum. "By tuning the laser wavelength and controlling the dose exposure, an action spectrum or germicidal effectiveness curve is developed to quantify the efficiency of micro-organism inactivation" [26]. The action spectrum is defined as "a plot of a relative biological or chemical photoresponse (= Δy) per number of incident (prior to absorption) photons, vs wavelength" [27]. Figure 15 shows the DNA damage and the infectivity caused at different wavelengths in the UV spectrum. From this we can see that at wavelengths below 240 nm there is a higher absorption of UV by the proteins and at 260 nm the loss of viral infectivity is lower than the DNA damage and at 254 nm, 270 nm, 280 nm, and 290 nm there is no difference in the infectivity and DNA damage. Therefore, it is evident from the figure that wavelengths below 240 nm have a greater difference between the genome damage and rate of inactivation, whereas it is relatively similar at wavelengths above 240 nm. Comparing rate of inactivation at 210 nm and 254 nm we can see that it is almost 16 times greater at 210 nm and then it decreases rapidly. Most of the biological pathogens follow this trend, with a little difference amongst them. This shows that DNA damage is not the only thing responsible for inactivation of viruses in water using a range of wavelengths. The low wavelengths show an important role in inactivating the pathogens and hence can be tailored in a way to combine with other wavelengths to get the best result in purifying water in the industry.

Spectral Sensitivity of Adenoviruses

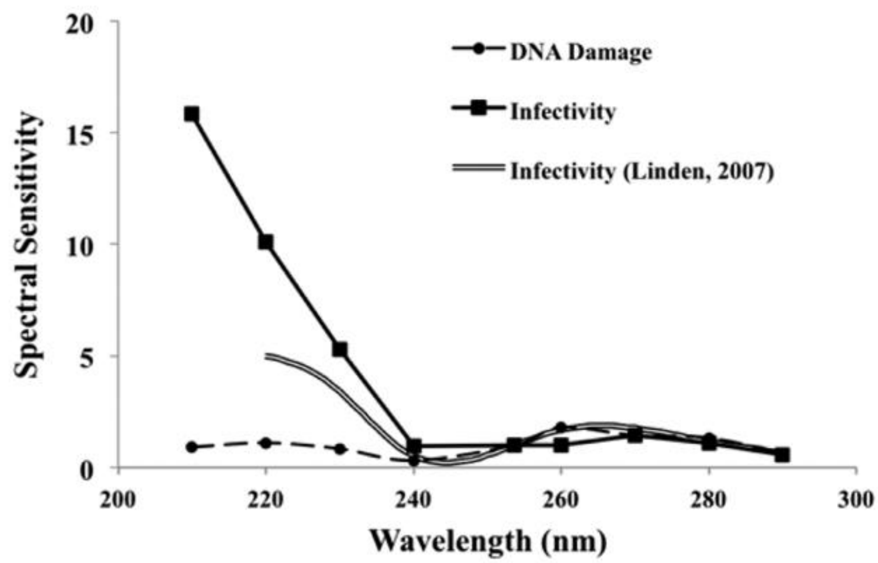


Figure 15. Spectral sensitivity of adenovirus 2 and its DNA damage. Figure from [28].

9.2.7 Selecting the LEDs

From all the knowledge discussed earlier the best option to pick LEDs are picking a semiconductor LED made from materials AlGaN, AlN or InGaN and choosing a couple of LEDs with different wavelength so we get a broad spectrum. In choosing the LED, we had to keep in consideration the size, cost, and energy constraint. The LED also needs to produce a good amount of intensity at the given wavelength to purify the water effectively.

9.2.7.1 M275D3

This LED is from Thorlabs, which is a very trusted company and therefore the assurance of the product would be great, and the product will be very reliable. The peak wavelength of this LED is 275 nm. The LED comes with a PCB and hence easy to use and implement in the product. The PCB is metal core and made in a way that it will seek in the heat emitted by the LED and hence protect the system and the product. The LED requires 12 V forward voltage and 300 mA maximum current. The output power generated by this is typically 68.3 mW. The emitter size is 2.7 mm x 3.3 mm, and the PCB thickness is 1.6 mm. The cost of this product is $139.67.

9.2.7.2 SU CULCN1.VC OSLON® UV 3636

This product is from Osram. This is also a well-known and well trusted company. The LED is made from AlGaN based flip chip and ceramic packaging. This is only the LED and does not have a PCB board with it, so it is difficult to implement it as it is very tiny and hence difficult to solder and implement it. The size of the LED is 3.5 mm x 3.5 mm x 1.71 mm. The peak wavelength of this LED is 275 nm. Voltage and current required by this LED are typically 5.7 V and 100 mA. The power generated is 10 - 13 mW. The cost for this LED is $6.27.

9.2.7.3 3535 LED Diode

MPN: lAf1186826
Manufacturer Part Number: gfW8379622
This product is from eBay made from a company in China. There are no details on the company and hence it does not seem that reliable, but it has got good reviews and rating, so it probably is kind of trustworthy. The peak wavelength is 265 nm. The voltage and current required is 5-6 V and 100 mA. The power generated is 2-5 mW. This product comes with a copper CREE PCB and hence easy to implement and test the product by just soldering wires at positive and negative ends and connecting to a power source with appropriate voltage and current. The of this LED with the PCB is 20 mm in the diameter. The cost of this LED is $4.89. They also have a similar one for 275 nm on eBay which costs $4.30.

9.2.7.4 UVC 250-260NM

This LED is from Shenzhen Trillion Auspicious Lighting Co., Ltd. This is a new company and they do not have those many reviews yet. The peak wavelength is 256.4 nm. The LED requires 24 V voltage and has a power consumption of 3 W. The radiant flux is 120 – 200 mW. The cost of this LED is $18.50.

9.2.7.5 DS-UV254P3Z-3535AC-S-06

This LED is from Zhejiang Danse Electronic Technology Co., Ltd This is also a new company. The peak wavelength is 255 nm. The LED requires a voltage of 5-7 V and current of 100 mA. The give out power of 10 -12 mW. This does not come with PCB and hence tough to implement and get the LED working as it is very tiny and has difficulties in soldering or finding the perfect mount for it. This LED is made of AlGaInP material. the cost of this LED is $3.85

9.2.7.6 HOUKEM-02W-255

This LED is from Dongguan Houke Electronic Co., Ltd. They are a new company from China too. This LED is made from InGaN. The peak wavelength is 255 nm. This requires 5.5 – 6 V voltage and 20 mA current. Typical output optical power is 3.5 mW. Not really sure if this comes with a PCB or not, but most likely it does not. The cost of this LED is $2.00.

After going through all these possibilities, we narrowed down to only 2 LEDs which match our product design and requirements. We had to look for size, material, cost, power, and electrical requirements. Due to these constraints the 3535 LED Diode from eBay have proven to be the best match as they are cheap, easy to implement, small and does not require a lot of power to function it. The last two are great options too but they do not come with PCB and the shipping cost on Alibaba for it is around $20 which makes the product expensive, and we do not even know if it will work perfectly in that small UV range. I am still looking more into it and talking to the dealers for less shipping cost and warranty so we can get one of them which is 255 nm. We want a broad UV spectrum and that would definitely help.

9.2.8 LED Design

Referring to Figure 6 design we need to test, implement, and modify the schematic for the optical path ray. Using a beam splitter to combine beams is only one of the ways to combine the two LEDs with different wavelengths. The other methods are using a dichroic mirror or diffraction gratings instead of the beam splitter. The reason to try these out is that due to the very small bandwidth difference between the two LEDs, optical components like beam splitter, diffraction gratings, and dichroic mirror will not work efficiently to combine the beams. Cannot be sure without testing it out. The phase matching is very difficult in LEDs and when we use optical components to combine beams, they should have a big difference in the bandwidths so one of them passes through and the other one reflects, but here the bandwidth difference being negligible it is very difficult to do so. If somehow, we manage to do it, it will be great. The other option is to just have two LEDs next to each other and have a broad beam projected. It should be broad enough so there is some overlapping and then project that out. The last option is to just drill two holes in the cap and without any interference let the LEDs project out the UV spectrum and sanitize the water. This would also be good as we will not be using any mirrors or lenses and hence no power will be lost giving greater efficiency.

9.2.9 LED Feedback Mechanism

It is important to know if the LED is providing enough power and sanitizing the water to be safe to drink. The LEDs might lose the power, get burnt out or just die and the user might not know about it and would just trust the system and drink the water. For this reason, we need a feedback mechanism which will tell us if the LED is working and hence giving us the confirmation that the water after sanitization process will be clean and safe to drink.

One way to get a feedback mechanism is by radiometry. Radiometry works on the principle of converting radiant energy to heat or electric current energy which can be used to test the system. There are several detectors which can be used in this based in the wavelengths, field of views, sensitivities, and other characteristics. To get the correct detector and have it calibrated to your needs one needs to define the exposure dose and spectral irradiance. There are different ways to measure the spectral irradiance like directly by using a spectroradiometer, by luminous flux and by human eye or by calculation from the radiance data. Important factors we need to take care of in choosing the detector which may affect the process are precision, sensitivity, accurate calibration, dynamic range – the upper and lower limit of the irradiance, field pf view, cost, speed of response, and portability.

In markets today there are several detectors available for UV wavelengths such as pyroelectrics, photodiodes, photomultipliers, thermopiles, and photoresistors. These systems work on converting the photon energy to current or voltage and then measuring that system to get the results. Photomultipliers, vacuum photodiodes, pyroelectric detectors, and photoresistors convert radiant energy to current whereas solid- state photodiodes convert it to voltage and thermopiles convert heat energy to voltage. Photomultipliers have great sensitivity but require very high voltage and are very fragile. They also lose sensitivity with time due to the alteration of the photochemical used in photocathode material. Vacuum photodiodes are similar to the photomultipliers but are not as fragile as the photomultipliers and hence better for portability and ruggedness. Solid-state photodiodes depend a lot on the material used to make them. Everything depends on the band gap energy of the material. They are sensitive to temperature increase, and they have two different modes: photoconductive and photovoltaic based on forward biased or reverse biased voltage supply. Photoconductive (reverse- biased) mode is more sensitive. Pyroelectric detectors are great and have very high precision but are very expensive due to the materials used to make them. Thermopiles are not sensitive detectors and depend on temperature for accurate measurement. Photoresistors are not as sensitive as any of the above detectors discussed but are cheap. portable and easy to implement and use.

For our system the best two options of all of them are a solid-state photodiode or a photoresistor. We need to use filters to capture the radiant energy for both. Due to our portability and size constraint these two would be of great choice. Photodiodes are more sensitive but also more expensive. For our system we do not need a more sensitive one. We just need to know if the LED is working or no and a photoresistor can let us know. Photoresistors are slower but time is not a constraint. Therefore, analyzing all the factors the best fit for our project is go with the photoresistor as our feedback to check if the LEDs are working properly or not.

9.3 Trace Element Detection

For the purposes of this document, "water quality" is analogous with the Prescence (or lack thereof) of hazardous trace elements.

9.3.1 Methodology

Water quality can be analyzed by a variety of chemical, optical, and electrical means. For the purposes of this project chemical methods are an inherently poor choice due to their disposability- a water bottle that requires monthly refills of obscure compounds and potentially toxic solutes is a poor choice for a hydration vessel. This leaves optical, electrical, and opto-electrical methods as potentially feasible methods.

9.3.1.1 Infrared Spectrometry

Infrared spectrometry identifies materials by introducing a known, broad, infrared spectrum to the sample. The returned spectrum, having transmitted through the sample, is then compared to the original spectrum [29]. By identifying the absorption spectrum of the sample, the chemical composition can be deduced to a high degree of accuracy (less than a ppm for gases) [29]. Infrared spectrometry has a shortcoming with aqueous solutions, however, as water is highly absorptive in the infrared regime. This quality makes infrared spectrometry a non-ideal solution for a spectrometer dedicated to the detection of chemicals dissolved in water.

9.3.1.2 Raman Spectroscopy

Raman spectroscopy appears to be the most relevant technology for the project as it is a non-destructive, contactless optical method capable of detecting minute concentrations of dissolved materials. This methodology exploits a non-linear effect of light knows as Raman Scattering, which involves a small portion of incident light shifting from the characteristic wavelength due to interactions with the vibration of atomic bonds [29]. This method, while first discovered using the broad spectrum of sunlight, needs its pump source to be monochromatic to obtain usable data [29]. This method is non-destructive but is observed in the UV range, which necessitates special care for imaging sensors to avoid ablation [29]. While most usages involve near-infrared pumps, this method can use pump sources in the visible range to reduce absorption by aqueous solutions [30]. Infrared lasers (and thus infrared spectrometry) are fundamentally incapable of functioning in a large aqueous sample due to the high absorption coefficient for said frequencies.

9.3.1.3 Fluorescence Spectrometry

Fluorescence spectrometry involves the stimulation of a sample with UV light, whose frequency-shifted response is shifted by contained fluorescent materials (primarily present in micro-organisms) [31]. This system, while experimentally demonstrated to be effective at determining the quantity and type of micro-organism present in a sample, is not of particular importance for this project’s design, as our product will destroy living contaminants. This methodology could potentially be useful for demonstration purposes, however, to provide a rapid means for showing the efficacy of the sanitization process.

9.3.1.4 Inductively Coupled Plasma

Inductively Coupled Plasma (ICP) is a method by which a sample is stabilized to a desired PH using bases or acids before introduction to a plasma source generated by electromagnetic inductors [32]. The process results in of several thousand Kelvin while drawing multiple kilowatt-hours of electricity and destroying the sample- despite its effective and reliable spectral output, the system is not low-cost, low-power, simple, nor compact [32].

9.3.1.5 Impedance Analysis

Impedance analysis detects and identifies the presence of metallic ions by registering the impedance among an array of electrodes of differing compositions [33]. The impedances are collectively compared to known, contaminated scores for similarity analysis- a similar response indicating the presence of the contaminant in question [33]. While impedance analysis is a reliable means of detecting dissolved heavy metals, the method requires submerged electrodes which would be an undesirable departure from the project's modularity (as the electrodes would need to be in the bottle rather than in the cap to ensure contact with the sample) while also introducing new surfaces that require sanitization and cleaning. Under ideal circumstance these electrodes would undoubtedly add a slight, metallic flavor to the bottle contents- a process which would be accelerated by a UV sanitizer.

9.3.1.6 Summary of Methodologies

The methodology most applicable to this application would be Raman spectroscopy. ICP requires extremely high temperatures and currents while impedance requires submerged probes that are likely to introduce undesirable flavors. Fluorescence spectrometry, while unreliable for detecting the project's scope of contaminants, is potentially useful for demonstrating the effectiveness of the sanitization module during demonstrations (see Table 15 below).

Summary of Methodologies

	Infrared	Raman	Fluorescence	ICP	Impedance
Senses	Atoms	Atoms	Micro-organisms	Atoms	Atoms
Rapidity	Rapid	Rapid	Rapid	Slow	Rapid
Cost	Mid-range	Mid-range	Low-cost	High cost	Mid-range
Destructive	No	No	No	Yes	Partially
Feasibility	Unfeasible	Feasible	Feasible for demonstrations	Unfeasible	Feasible but undesirable

Table 15: Summary of Methodologies

9.3.2 Spectral Isolators

While the function of this component is the same for both diffraction gratings and prisms, the differing implementation results in key differences for the overall system.

9.3.2.1 Prism

Prisms are the most basic means for separating incident light into its spectral components. Light of differing wavelengths is refracted at different angles in the prism due to the wavelength-dependence of the material's refractive index. While prisms made from ice, plastic, candy, and resin are all usable in a high-school science demonstration, it is worth noting that many materials used to produce prisms absorb UV radiation and/or are outright destroyed by it, while tailor-made glass (such as THORLAB's UV fused silica) is somewhat expensive [34]. This limitation is particularly of a concern for Raman spectroscopes, which typically measure their outputs in the UV region.

9.3.2.2 Diffraction Grating

Diffraction gratings use grooves in a reflective or transparent sheet/panel to separate light into its spectral components [35]. While lighter and potentially smaller than prisms, diffraction gratings experience worse efficiency rates than prisms [34][35]. Additionally, the materials used to construct more economical gratings are vulnerable to ablation in intense UV.

9.3.2.3 Summary of Isolators

Neither isolator is a clear winner, and the spectrometer design has switched between both components several times during the design process. The current design, however, uses a prism to isolate the spectral components of the sample, as the pump laser has been changed to a visible wavelength making absorption in the UV schema a non-issue, the prism's correspondingly low cost for the visible regime (see Table 15 below) gives it a lead.

Summary of Isolators

	Prism	Diffraction Grating
Cost	Low	High-Medium
Compactness	Moderate	Moderate
Efficiency	Moderate	Moderate-Low

Table 16: Summary of Isolators

9.3.3 Water Quality Sensor

A sensor is required to convert the spectral bands into data, which is accomplished by shining the spectrally separated beam onto an array of receptors- or by projecting the beam onto a single receptor and shifting the spectrum across it. If the beam is separated in a horizontal manner, then the resultant image on the sensor will vary in wavelength based on its horizontal location.

9.3.3.1 CCD

Charge-coupled devices (CCDs) consist of an array of metal oxide semiconductors (MOS) in parallel to an array of illumination-shielded semiconductors [36]. CCDs operate by collecting optical energy and passing said energy as an electric charge into their respective semiconductor cell, which is shifted toward a readout cell every clock cycle. This results in a constant stream of individual cells outputting charge, which can be formed into an image.

9.3.3.2 CMOS

Complementary metal-oxide semiconductors (CMOS) are similar to their CCD predecessors but contain readout cells on every pixel [36]. This added complexity is beneficial for some purposes but until relatively recently was a substantial engineering hurdle resulting in increased cost, although manufacturing techniques have improved significantly. Essentially all commercially available CMOS (and CCD) sensors are packaged with a microchip that converts these readouts into a composite video signal, which effectively expresses information on color, intensity, and sound (from an onboard microphone, if applicable) for each pixel in a clever and complex manner [37]. This complexity, unfortunately, makes device integration with microcontrollers a titanic feat worthy of its own senior design project.

9.3.3.3 Photodiodes

Photodiodes are produced in single diodes or arrays of diodes. Photodiodes the simplest of the researched technologies, consisting of a PN junction or an array of PN junctions constituting an array of pixels, all of which are continuously active [36][38]. These devices are attractive for spectrometers due to their large pixels and narrow gaps, reducing the quantity of light (and thus information) lost. Photodiode arrays are industrially available which are tailored to specific spectral ranges (such as UV or IR), which is generally lacking in commercially available CCD and CMOS chips [43].

9.3.3.4 Phototransistors

Phototransistors, like photodiodes, are semiconductor junctions which become more conductive under illumination [38]. Unlike photodiodes, however, phototransistors use a PIN junction to achieve higher responsivity (sensitivity) at the cost of a poorer frequency response [38]. This higher responsivity is also coupled with a loss in linearity, however- and while not important for a simple switching mechanism, this makes precise measurements with phototransistors a challenging feat [38].

9.3.3.5 Summary of Sensors

While a CMOS sensor was originally slated for the project due to its commercial availability and high resolution, the device's output signal is beyond the realm of feasibility for a cost-efficient microcontroller and would cause a significant increase in software complexity as well. While phototransistors are highly appealing due to their high sensitivity and low cost (see Table 17 below), this sensitivity is marred by non-linearity, resulting in less precise measurements of intensity. Photodiodes, thus, are the most feasible component for the project's spectrometer.

Summary of Sensors

	CCD	CMOS	Photodiodes	Phototransistors
Cost	Low-Medium	Low-Medium	Low-High	Low
Compactness	Medium-High	Medium-High	Low-High	High
Sensitivity	Medium	Medium	Medium	High
Ablation	Possible	Possible	Unlikely	Unlikely

Table 17: Summary of Sensors

9.3.4 Rotational Mechanism

As the second version utilizes a rotational mechanism to sweep the spectrum over the sensor, a servomotor or other precise rotator is required, and the placement of said device must also be investigated.

9.3.4.1 DC Motors

Direct current motors are an excellent way to convert electricity to rotational movement, providing consistent torque regardless of resistance from the load. These motors, however, are utterly useless for applications requiring strict angular precision, as they move in a continuous fashion rather than incrementally. While a solution could be attained by using pulses of consistent duration and current as an increment, and reducing the output with a high gear ratio, these remain excessively imprecise for optical applications.

9.3.4.2 Stepper Motors

Stepper motors "step" their axles by a set angular value in response to electronic input. These devices are capable of stepping in the clockwise and counterclockwise orientations based on the input signal but cannot rotate a partial step. This in turn means that the internal gearing dictates the angular resolution of the component, with the step angle as the smallest distinct value. For commercially available stepper motors of a size implementable in this project, the angular resolution tends to be 5.625° or 18°, corresponding to 64 and 20 teeth in the internal gearing, respectively [39][40]. It is apparent that these angular resolutions are insufficient for the spectrometer, but this can be remedied by using external gearing [41]. While external gearing adds unwanted mechanical complexity to the design, it does allow for the resolution to be tailored to fit the project, as a modified gear ratio allows for higher angular resolutions.

9.3.4.3 Servomotors

Servomotors are, most directly, a more advanced version of a stepper motor. These devices are directed to specific positions using a PWM signal duration to dictate the exact angle [42]. This is generally accomplished by having a header duration (to signify that a signal is being sent) followed by a duration actually representing the position desired (for a servo with 180 degree rotation this could involve 1 or 2 millisecond signal durations for -90° and 90° respectively, with angles between determined by partial milliseconds [42]. This is especially advantageous in the event of an unexpected power failure, as servomotors can simply be reset to their starting position while stepper motors require more creative means of returning to a known position. Worth noting, however, is that the "starting position" is prone to change in the proposed spectrometer, as variations in pressure exerted from the bottle's threading may adjust the alignment of the system, and temperature changes will affect the characteristic wavelength of the laser diode as well. The angular resolution of the device is not advertised on commercial datasheets for the hobby retailers investigated, and while the angular resolution can be decent it may be necessary to implement a gearbox with a servomotor as well. These devices accomplish these functions using clever but imperfect mechanisms (namely, the angular value is determined by measuring the resistance offered by a potentiometer, which serves as the final axle), leading to potential issues regarding accuracy.

9.3.4.4 Summary of Rotational Mechanisms

Based on the technological investigation (summarized below in Table 18), I believe that a hobby servomotor is currently the most desirable option due to the simplicity of implementation involved, decent baseline resolution, and power failure recovery. Physical prototyping may result in increased appeal for stepper motors due to their higher precision, although usage of a stepper motor must be coupled with a gearbox to increase angular resolution (a shortcoming which may exist for servomotor implementation as well).

Summary of Rotational Mechanisms

	DC Motor	Stepper Motor	Servomotor
Cost	Low	Low	Low
Precision	Low	Excellent	Decent
Resolution	Low	Medium	Decent
Feasibility	Unfeasible	Feasible with a gearbox	Feasible, may require a gearbox

Table 18: Summary of Rotational Mechanisms

9.3.4.5 Summary of Analysis

Of the technologies available and feasible for the project, the most appealing design appears to consist of a Raman spectroscope with a prism as a spectral isolator, a photodiode for sensing, and a servomotor for sweeping the spectrum across the photodiode. Prototyping may result in amendments to this, however, as certain technologies with high paper appeal (such as servomotors) may lack the advertised capabilities in practice. Additionally, the rotational mechanism may require a gearbox to increase angular resolution.

9.4 Safety Switch Detection Device

It will be of utmost importance to ensure that any lasers for either quality sensing or sanitization cannot be activated accidentally while the cap is detached from the bottle. Having a safety switch hardwired into the circuitry that can detect whether or not the cap is secured onto the bottle will prevent the user from shining any harmful rays towards someone's eyes or skin. It is greatly desired to implement a power safety switch that does not rely on physical pressure, due to the possibility of an inadvertent tripping of the switch due to pressure from either a user's fingers or some surface that the bottle cap may be resting upon. Therefore, it is suggested that a device will be chosen that relies upon the detection of a constant magnetic field. Attaching a magnet to the water bottle towards the top will allow the safety switch to close while the lid is properly secured. It would certainly be possible for an intentional override of this safety mechanism by using another magnet but designing a product that is preventative of malicious abuse is outside of the scope of this project.

9.4.1 Reed Switch

One device that relies on the presence of a magnetic field in order to engage is the Reed switch. The advantages of using this type of component are that they do not need added circuit components to be used and they do not constantly draw power [44]. However, it is important to note that a Reed switch consists of reeds enclosed in a glass container and this device is sensitive to intense and abrupt acceleration [44]. There will be many sensitive optical components within the project device that will not do well if the cap is dropped, so it seems to be a reasonable assumption that a Reed switch would not be the only component that is sensitive to sudden impact. It is also worth noting that a Reed switch would not be damaged by the introduction of moisture into its surroundings [44]. Although the intention is to properly seal the electronic components within the casing of the water bottle cap from the humidity of the bottle or its surroundings, it is an added bonus to have one more component that is resistant to water vapor.

9.4.2 Hall Effect Switch

A Hall effect switch is a semiconductor device that can deliver a triggering output signal in the presence of magnetic fields [44]. Although these devices do need a constant supply of current for operation, it is definitely advantageous that Hall effect switches are solid-state devices without any moving elements [44]. This difference in device construction would make the safety switch much more resistant to damage from sudden impact caused by the dropping of the water bottle cap. However, it has already been established that dropping the project device would already be extremely destructive to other vital components and therefore this physical robustness seems to be a fairly unnecessary advantage.

9.4.3 Safety Switch Type Selection

Overall, the lower complexity and power requirements of the Reed switches are the biggest selling points for this type of technology. Some of the main constraints for this project are the limited amount of energy afforded by whichever battery will fit in the cap and the limited amount of space in which all of the components must fit together. Choosing a safety switch device that will maximize the product battery life while minimizing the complexity of the printed circuit board will definitely help to create a more competitive alternative to the existing products of similar nature.

9.4.4 Reed Switch Selection

Investigating the available options for Reed switches has brought up a discussion about the best type of connection to incorporate the switch onto the PCB while also positioning it appropriately near the bottom of the cap casing. There are different magnetic sensitivities available for the Reed switches on the market but having a more sensitive switch that is closer to the top of the cap would increase the chance that the magnetic flux from the wireless charging could accidentally actuate the Reed switch. Rather than using a through-hole or surface mount device, there are sensors in the market that come with wire leads installed onto the switch. Using a design of this type would give us much more flexibility as to where we adhere the Reed switch within the cap casing. In this way, we will be keeping the safety switch as close to the magnet that is intended to activate it while maximizing the distance from the wireless charging portion of the device.

At this point, three options from the same manufacturer have been found that each have different sensitivities listed on the sales website. The three options to be considered are listed below in Table 19, and the only difference that seems to be listed is the sensitivity, because the listed datasheet for each online is the same with a listed range of "pull in range available" being between 10 – 25 AT (Ampere-turns). These sensitivities are given in terms of magnetomotive force (MMF or F) which can be related to the listed magnetic flux density (B) of a magnet by $B = \mu * F/l$, where μ is the total magnetic permeability of the path between the two poles of the magnet and l is the length of that path (as learned in EEL 4205, Electric Machinery). As discussed in the wireless charging section, we are trying to use a cap material that has a high reluctance to magnetic fields so as not to interfere with the interactions between the desired electromagnetic components. Therefore, it is probably safest to assume that the magnetic permeability of the path between poles that would include the Reed switch while the cap is closed is relatively close to that of free space, $\mu_o = 4*\pi*10^{-7}$ N/A^2 [45]. However, in making a calculation for the path for the magnetic field lines, it needs to be considered that these are concentric arcs due to the fact that the Reed switch will need to be coaxial with the magnet. Taking all of this into consideration, it is encouraged that we should simply buy the most and least sensitive options below so that we have flexibility in terms of the placement of the switch in the cap and the magnet on the bottle. Since each switch is only $2.03 and either would be reusable between different iterations of project design testing, it is not thought to be an undue expense to buy two different switches with different levels of sensitivity.

Summary of Reed Switch Options

	MS-2431-3-1-0300	MS-2431-3-2-0300	MS-2431-3-3-0300
Cost	$2.03	$2.03	$2.03
Output type	SPST-NO	SPST-NO	SPST-NO
Must Operate	10 ~ 15 AT	15 ~ 20 AT	20 ~ 25 AT

Table 19 Summary of Reed Switch Options

9.4.5 Magnet Selection

Investigating the various magnets available yielded a handful of promising options for implementation in this project and the synopsis of some of the best choices are listed below in Table 20. The factors that weighed most heavily in the selection process have been size and price (including the minimum quantity that can be ordered). Chances are that we will not need to order a minimum of five of any of these magnets in the prototyping process, so all of the options below were selected in part due to the minimum order quantity being one. Since the current plan is to affix this magnet near the top of the outside of the water bottle, we want to have the magnet be as small as possible so that it does not impede the user from handling the bottle and risk being rubbed off from regular handling. The smallest option is clearly the SMCO5, but the price is over four times that of the 9016 which is the same shape. However, $1.45 is by no means going to be a significant addition to project expenses and we definitely wish to ensure that this magnet will sit as flush as possible to the side of the cylindrical water bottle that will be used for this project. Therefore, it seems that the SMCO5 1.9X3MM will be the best choice due to its size being the smallest.

Summary of Magnet Options

	SMCO5 1.9X3MM	9016	8028	NDFEB 10X5X1.9MM
Cost	$1.45	$0.29	$0.98	$1.16
Shape	Cylinder	Cylinder	Cube	Rect. Prism
Dimensions	0.074"D x 0.118"H	0.079"D x 0.315"H	0.25" SQ x 0.25"H	0.394" L x 0.20" W x 0.070" H
Magnetization	Axial	Axial	Axial	Axial

Table 20: Summary of Magnet Options

9.5 Embedded Hardware

9.5.1 Controller Selection

The controller for the product will most likely be one of the following: FPGA, DSP, ASIC, or a microcontroller. The following controller technologies were investigated and compared, to determine which option would be a suitable choice for the project.

9.5.1.1 FPGA

FPGAs are a very capable choice, they typically have a faster release to the market, are relatively affordable, and they allow for flexibility and re-programmability when writing code. But they also require a significant amount of software/coding overhead in HDL, and they are not the best option when it comes to power consumption. Both electrical engineers in this project do have some preliminary experience in the Verilog coding language; however, the programming process would be much simpler if it were to be performed in another language that is more abstracted beyond logic gates [46][47]. This project will require a sophisticated level of firmware and software that can handle wireless communication, interface with sensors, and report quality data to the user. Because of the aforementioned reasons, an FPGA would not be a suitable choice for this embedded system.

9.5.1.2 DSP

DSP chips excel in both real-time processing and quick arithmetic computation. They also typically have a simpler coding process, and are easily reconfigurable since the data is digital, instead of analog. Since this project may determine water quality based on images represented as parallel output bits, a DSP would be suitable for this application. However, DSP chips, due to their high-speed processing, have a higher power consumption compared to a microcontroller. They also require additional filtering and IC's such as an ADC and DAC. Lastly, DSP chips are typically costly, and the electrical engineers on this project do not have a lot of experience on DSP programming. For these reasons, a DSP would not be a suitable choice [46][48].

9.5.1.3 ASIC

Since ASIC's are mainly comprised of fixed logic gates and digital circuitry, they are very affordable. They also are highly optimized in terms of power consumption and clock cycles. However, since they are designed for one purpose only, they are not re-configurable. Due to their specific functionality, they also typically have a high development overhead [46][47]. For this project, an ASIC would not be a good choice because the device needs to be able to interface with sensors, process data, and wirelessly communicate via Bluetooth. All these concepts are too individualistic and therefore an ASIC does not currently exist with all the necessary capability for this project. This would mean the ASIC would have to be independently developed for the device.

Although it is true that costs per unit go down after development, creating an ASIC from scratch would not be feasible for this project. Therefore, an ASIC (during early/prototype phases) would not be a suitable choice for this project.

9.5.1.4 SOC

SOC's are similar to a microcontroller, but they have more specific peripherals. A Bluetooth SOC would be a great choice for this project. However, it would also have to have enough GPIO pins to support the video output for the water quality sensor, and enough I2C/UART pins to communicate with all of the sensors. An example of an SOC that could be implemented in this project is Silicon Labs' EFR32BG22, or the Nordic Semiconductor nRF52840. Both of these SOC's incorporate Bluetooth 5 and would therefore be useful for this project. They also both appear to have I2C, UART, and GPIO pins. As long as these SOC's have enough of the necessary connections, they would certainly be implementable. However, the main constraints with using this technology are product availability and software overhead. Most of the Bluetooth SOC's are currently out of stock (or very limited stock) at major distributors such as Mouser and DigiKey. Using an SOC would also require becoming familiar with a new architecture and programming interface, which could prove to be very time consuming for this project. For these reasons, an SOC most likely would not be a suitable choice for this project.

9.5.1.5 Microprocessor (SBC)

One specific brand of Single Board Computer (SBC) is the Raspberry Pi series that is capable of running a full-fledged operating system known as Raspbian [49]. The product known as "Raspberry Pi Zero W" is described by vendors as a smaller sized version of the Raspberry Pi, but it also includes built in Wi-Fi and Bluetooth capability [49]. The price for these units is $10, but there are additional accessories needed to be able to run the device. The Raspberry PI would be a good choice for the spectrometer v1 option that involves video signal processing.

This controller technology is a microprocessor, which is similar to a microcontroller but has a different organization of device peripherals. A microcontroller has all processing, memory, and I/O in one chip. Whereas a microprocessor consists of a CPU and utilizes a bus for communicating with other peripherals. This allows for designs to be more flexible and incorporate different levels of memory access (depending on speed and capacity). This would be a worthwhile addition, if water quality data processing requires rapid access to a lot of memory.

Using a product like this would provide a lot of capability in terms of processing power and built-in wireless connectivity, but there are still a few issues that would need to be considered. One such problem is that having a full-blown computer installed in the project device would undoubtedly draw more power than a microcontroller with less advanced features. The addition of every peripheral that would be plugged into an SBC device such as this would add to the power requirements of the entire system, which would inevitably lead to a shorter battery life from whichever size unit is capable of fitting within the casing. Also, using this technology would have a large amount of software overhead for the team. Raspbian is based on Linux OS and would require a lot of research into how to program the device and interface with peripherals. Another constraint that needs to be respected is the size of each piece being included in the final project. The width and thickness of the Raspberry Pi Zero W are both fairly miniscule, but the total length of the board is 2.6 inches [50]. Units investigated for the power system are already well over two inches in length, but it is worth mentioning that the project is supposed to fit in a casing that will serve as a water bottle cap and every addition of large circuitry should at least carry its weight in functionality that is actually needed.

9.5.1.6 Microcontroller

Microcontrollers are relatively affordable and easy to re-configure. They are a great controller to implement for GPIO functionality, which would be favorable for the sensors necessary in this project. They typically support wired communication protocols such as UART, I2C, and SPI. They also often offer low power modes which would drastically improve the overall battery lifetime. Typically, the architecture of the chip can optimize the clock cycles, which helps to improve the overall efficiency of the device [46]. They also have memory storage, which will be helpful with possible digital processing necessary for predicting water quality. Microcontrollers are also more cost effective than other aforementioned options. But they do have limitations on the tasks that they can perform [47]. However, this project appears to have a reasonable scope for implementing a microcontroller. In addition to this, the electrical engineers have the most experience in programming a microcontroller and implementing it into various embedded systems.

Prior to senior design, the electrical engineers on this project have previous experience on working with TI microcontrollers. Both the MSP430FR6989 as well as the MSP430G2553 were utilized in previous embedded programming classes. Due to this familiarity, it would be beneficial (as far as time constraint) if the project sought to implement a microcontroller from the TI family. Other microcontrollers would require a significant amount of time to understand the user manual and become familiar with the device architecture and software interface. Currently, our team has access to MSP430 development kits. So, this would also decrease the project budget if we decided to go with one of the MSP430 microcontrollers.

9.5.2 Controller Selection Summary

Using the results above, each controller type was compared across the following parameters: Estimated cost, software overhead, market availability, and team familiarity. Each column after estimated cost uses a scale from 1-6. In which a 1 represents the most desirable option, and 6 represents the least desirable option. Table 21 clearly demonstrates that a microcontroller would be the most suitable choice for the project. Therefore, more technological investigation on types of microcontrollers will begin to take place.

Controller Selection Summary Table

Controller Types	Estimated Cost	Software Overhead	Market Availability	Team Familiarity
FPGA	$4 to $20	2	2	2
DSP	$2 to $60	3	4	3
ASIC	> $1000	6	6	5
SOC	$5 to $10	5	5	6
SBC	$5 to $10	4	3	4
Microcontroller	$2 to $10	1	1	1

Table 21: Controller Selection Summary

9.5.3 Spectrometer V1 Project Implementation

In order to use the spectrometer (v1) for water quality analysis, the embedded device would need to be able to handle and composite baseband video signals. Using a CMOS camera such as the Turbo EOS2 V2, would require more than a typical microcontroller. The signal (carrying color video and sound) would be much more complex than typical GPIO pins on a microcontroller can support or handle. An additional IC (at the least) would be necessary for project implementation. For example, the TI TVP5150AM1 would be able to convert the composite video signal into standard ITU-R BT.656 output (or YCbCr) which could then theoretically be passed to microcontroller, vectorized, and processed to determine water quality data. Using such an IC would then introduce even more software overhead and product cost. Another option is to use a DSP/chip that is compatible with this baseband signal; however, again this would introduce a large amount of software overhead and it is difficult to find a chip that still has enough peripherals for every other project component. For these reasons, spectrometer v2, which focuses on a photodiode and a stepper, appears to be much more manageable than this design.

9.5.4 Spectrometer V2 Project Implementation

Implementation of the version 2 spectrometer is less output intensive but requires both output and input signals. The spectrometer module's stepper motor requires inputs to cycle through positions. The spectrometer module's output is an analog voltage across the module's photodiode- which, while much less complicated than a composite video signal, requires a voltmeter or equivalent system to convert a microvolt response from the diode into a value that represents said response with a reasonable (1 mV) degree of precision. Using a microcontroller with a built-in ADC should suffice for this job. It is important that it has enough resolution (within a small voltage range) to be able to detect such minute changes in voltage. More specifically, the output of the photodiode would serve as an analog input to the microcontroller, that is then converted to a digital signal and interpreted/analyzed for purpose of water quality.

9.5.5 Spectrometer V3 Project Implementation

The third implementation replaces the diffraction grating with a prism once again due to a change in the sanitizer design. As the sanitizer was updated to use LEDs instead of a laser, the spectrometer was forced to adopt its own stimulation source. As the sample is a (relatively) large reservoir of water, the traditionally popular and widely available infrared diodes were off the table due to water's high absorption coefficient for that regime. Instead, a visible green laser has been added as a pump source for the sample. A prism was interchanged with the previous diffraction grating due to the swap to visible ranges as transmission is no longer a significant concern in affordable glasses. The device was also updated to have a servomotor instead of a stepper motor, although this has the potential to be reserved pending equipment tests. The servomotor will require typical power supply connections (most likely 5V and GND), and then will utilize a PWM signal to change the angle of the device. This can easily be accomplished by the microcontroller’s timing capability. From here, the PWM signal will be driven out on a pin that can be configured to have PWM capability.

9.5.6 Optical Diode Power Consumption

It is possible that the sanitization UVC LEDs may need to be constantly on, while perhaps it makes more sense to occasionally observe water quality (instead of at a constant rate) However, in order to optimize device power consumption and battery life, the optical diodes (from an electrical standpoint) should be pulsed on and off when operating. If the diodes were constantly on, that would create constant current draw which will then decrease the overall battery lifetime. To account for this possible method of operation, the selected microcontroller should have functionality to generate multiple PWM signals. This is because there already is a PWM signal necessary for manipulating the servomotor, so additional signals will be required for pulsing the optical diodes on and off.

9.5.7 Microcontroller Comparisons

9.5.7.1 Desired Microcontroller Feature Set

Prior to investigating the available products on the market today, it is important to first come up with a rough estimate of the desired features that the microcontroller should have. In addition to the features below, it is useful to select a microcontroller (and programming language) that the team is already familiar with. Due to the electrical engineering project focus, having a smaller software overhead will help streamline the product development process. As a side most of the packages/variants of each microcontroller considered were selected since they were the most readily available (in comparison to other packages/versions). Below in Table 22, the feature set for this project’s ideal microcontroller is outlined in detail.

Microcontroller Feature Overview

Feature	Desired Value/Quality
Voltage supply	2-5 V from on-board regulators
Low-Power Draw	< 10 µA
Cost	< $10
On-board Wired Comms Protocols	UART, I2C (possibly SPI)
Current-Driving Pins	Potentially drive Two UV-C LEDs and green laser directly
Low-level Wake-up (LLWU) Pins	Approximately 8 in total (to account for various inputs to the microcontroller).
GPIO Pins	At least 20 pins (including LLWU)
ADC resolution	10-12 bit for millivolt precision sensing
Analog Input Pins	Water quality sensor (1 pin required)
PWM Capability	< 100 µs, at least 2 pins (Servo, and green laser)
Interrupts	Operational Safety, and Process Hierarchy
Memory	Around 50 kB (Overestimate of water quality data/analysis)

Table 22: Microcontroller Feature Overview

9.5.7.2 ATmega328-PU

The ATmega328P is easily the most widely available microcontroller (out of the options considered) in the market today. This is a very popular microcontroller that is most notably used in the Arduino embedded development kits. It has a supply voltage ranging from 1.8 V to 5.5 V, with six different sleep modes. There is a power-down mode that has a constant current draw of 0.1 microamps when the microcontroller is being operated at 1.8V. The clock can run up to 20 MHz (with 4.5 V supplied). This microcontroller also supports SPI, I2C, and USART; the latter being a similar serial connection to UART with slightly different framing protocols. It has 32KB of program memory, 1 KB of EEPROM, and 2KB of internal SRAM. As far as timing capability, it has two 8-bit timers, and one 16-bit timer. All of these can be run off an internal oscillator. This microcontroller also has 23 I/O lines, six PWM channels, an 8-channel 10-bit ADC, and a built-in comparator. It is also worth mentioning that the data sheet specifies interrupt and wake-up features on pin changes. As far as packaging, it comes in a 28-lead/pin package.

This microcontroller certainly has a lot of attributes for this project. Some of the most important being the market availability today as well as the ability to code/flash through the Arduino IDE (with a bootloader). Potentially being able to write code in a simpler IDE such as Arduino and then flash the microcontroller would help mitigate software overhead for the project. Although, the project members are less familiar with this microcontroller, utilizing the Arduino IDE would still have a smaller software overhead than trying to learn a brand-new microcontroller platform from scratch. This option also has a significant amount of independent PWM signals which could come in handy if the project requires additional PWM signals that are not directly related to the servomotor. For the scope of this project, it initially appears that this option has more than enough timing capability, as long as the microcontroller's internal oscillator is reliable enough. It is an added benefit that this option comes with a small amount of EEPROM; this yields an instant erasure method (done through the hardware itself). Another great feature of this microcontroller is the extensive amount of power modes. This would provide more flexibility depending on which peripherals and device internals need to operate at various times.

It is very likely that the Arduino IDE would oversimplify device operation. This would more specifically mean that programming with the Arduino IDE would not have as much capability as programming in another environment more closely related to Atmel. Using a new environment such as Atmel Studio would then involve a larger software overhead than the TI microcontroller options. Extensive time would have to be spent understanding the ATmega328P user guide and how to interface with all device internals and registers. Since this project will already have a large amount of software overhead with Bluetooth and app integration; it is preferable to not have to learn an entirely new embedded system. Additionally, this microcontroller has a smaller ADC resolution in comparison to some latter alternatives. This microcontroller also has a smaller number of I/O pins which could lead to issues during pin configuration later on in the device process. For these reasons, this microcontroller is not the best choice for this project.

9.5.7.3 MSP430G2553IN20

The TI MSP430G2553 is a microcontroller that the team has previously worked with in prerequisite classes. This microcontroller has a supply voltage from 1.8 V to 3.6 V, and an ultra-low power mode of 0.5 microamps (with five additional power saving modes). It also supports serial communication such as I2C, UART, and SPI. It has a built-in ADC with 10bit conversion, a comparator, as well as 16 I/O pins. As far as timing capability, this microcontroller has an internal LF oscillator as well as an internal 32-kHz crystal. It is capable of frequencies up to 16 MHz, and has two timer modules, each with three capture and compare registers. Each CCR of the timer modules appears to be able to have its own PWM signal. As far as the available packages, it comes in 20-pin, 28-pin, and 32-pin SMT packages. However, the 20-pin IN20 package appears to currently have the largest quantity in stock.

This would be the cheapest option considered; since economic constraint is very important for this project, component cost will heavily weigh-in to microcontroller selection. Additionally, team members already possess the development board for this microcontroller, which would help reduce development costs on the project. It also is a very widely available microcontroller in the market right now. Again, due to microcontroller scarcity, that is very important to consider when designing the prototype device. The electrical engineers for this project are most experienced in the TI family of embedded programming. So, the MSP430G2553 would have some of the smallest software/development overhead. It also has a smaller device footprint and therefore would allow for easier component incorporation on the PCB.

If this microcontroller was selected, there certainly may be a few flaws. It is possible that this microcontroller may end up having a scarce amount of GPIO pins, depending on the pin variation/configuration. It would be wiser to select a microcontroller that has more available pins and configuration flexibility. This option also may not have enough flexibility from a timing standpoint, having only two timer modules may limit some of the various timing requirements for the project. Also, this option has a slightly smaller bit-resolution ADC which would then yield less resolution for the water quality sensing. That is a critical flaw when choosing this option, since this particular sub-system is imperative and requires high precision (in near microvolts). This microcontroller choice also has a significantly smaller memory size which may impact the ability to track water quality data. Even though 16 KB may be enough for the project, it would be better to have a surplus of memory for development overhead instead of constantly maxing out memory capacity. These are the main flaws with selecting this particular microcontroller.

Even though this microcontroller appears to have some flaws, it is still widely available in the marketplace and appears to be capable enough for this project. So, this microcontroller is still highly considered for this project, and it is important to consider the proposed pin configuration. The table below represents a proposed pin configuration for the 20-pin version of the microcontroller that contains: UART Tx and Rx, 8 Interrupt GPIOs, 1 Analog I/P, and 2 PWM outputs. Red font color in the table below represents the selected configuration for each pin (mainly related to peripherals). There are other connections (required for flashing the microcontroller) that have not been selected since they will be similar n across every microcontroller with proposed pin configurations.

Proposed 20-Pin G2553 Pin Configuration

20-pin G2553IN20	
Pin #	Configurations
1	DVCC
2	P1.0/TA0CLK/ACLK/A0/CA0
3	P1.1/TA0.0/UCA0RXD/UCA0SOMI/A1/CA1
4	P1.2/TA0.1/UCA0TXD/UCA0SIMO/A2/CA2
5	P1.3/ADC10CLK/CAOUT/VREF-/VEREF-/A3/CA3 5
6	P1.4/SMCLK/UCB0STE/UCA0CLK/VREF+/VEREF+/A4/CA4/TCK
7	P1.5/TA0.0/UCB0CLK/UCA0STE/A5/CA5/TMS
8	P2.0/TA1.0
9	P2.1/TA1.1
10	P2.2/TA1.1 10 11 P2.3/TA1.0
11	P2.3/TA1.0
12	P2.4/TA1.2
13	P2.5/TA1.2
14	P1.6/TA0.1/UCB0SOMI/UCB0SCL/A6/CA6/TDI/TCLK14
15	P1.7/CAOUT/UCB0SIMO/UCB0SDA/A7/CA7/TDO/TDI
16	RST/NMI/SBWTDIO
17	TEST/SBWTCK
18	XOUT/P2.7
19	XIN/P2.6/TA0.1
20	DVSS

Table 23: Proposed G2553 20-Pin Configuration

Using the proposed pin configuration above, this would only yield 3 remaining GPIO pins (w. interrupt capability). This leaves very miniscule room for adjustment in the future. If a board revision required significant overhaul of any subsystem, this may require assigning a lot more than three additional GPIO pins. Therefore, even though this microcontroller is cheaper and has a larger quantity currently in stock, it is still not the right choice for this project. This is because this configuration is already incredibly limited and does not have any room for additional feature sets in the future.

9.5.7.4 MSP430FR6989IPN

Not only is this the most familiar microcontroller considered, but it also is one of the most capable. It supports a wide supply voltage range from 1.8 V to 3.6 V, as well as optimized ultra-low-power modes. The stand-by power mode with the low frequency oscillator is 0.4 microamps. It also has a real-time-clock low power mode with a typical current draw of 0.35 microamps. The device comes with 128 KB of memory; it is worthwhile to note that this memory is non-volatile. As far as timing, this microcontroller comes with five 16-bit timers with up to seven capture/compare registers each. It appears that each capture/compare register is capable of having its own PWM signal. There is also a 12-bit ADC with 16 input channels, and a 16-channel analog comparator. Ports 1,2,3, and 4 are all capable of waking up the microcontroller from low power mode. A port is typically comprised of 8 GPIO pins; therefore, 32 pins in total are capable of waking up the microcontroller (synonymous with Low Level Wake Up terminology). This option also comes with enhanced serial connection which boasts UART with automatic Baud-Rate detection, SPI, and I2C with multiple-Slave addressing. There is also a built-in UART and I2C bootloader. The clock system in this microcontroller also has a lot of flexibility. There is a fixed frequency DCO (with 10 pre-set frequencies), a low power and low frequency internal clock source, and low frequency and high frequency crystals. The MSP430FR6989IPN has either 63 I/O pins. This microcontroller comes in an 80-pin or 100-pin package. However, the 100-pin package is very scare right now and also the 80-pin package seems sufficient for this project. This microcontroller has more than enough features and pin configurations, while creating a smaller footprint than the 100-pin version would on the printed circuit board.

The largest advantage that this microcontroller offers is a lower software overhead, since both project electrical engineers took an entire embedded programming class that utilized this particular microcontroller. This would then mean that the majority of software development would involve incorporating Bluetooth technology as well as an app. This option also offers the largest amount of timing modules; this would certainly be helpful since multiple unique PWM signals may need to be driven from the microcontroller at different times throughout typical device operation. It is always helpful for the microcontroller to have a wide supply voltage range. Depending on the battery and regulators utilized, being able to run the microcontroller at a lower supply voltage (with a lower clock frequency), may help improve battery lifetime. This microcontroller has more than enough memory for the scope of this project. Selecting this option will also provide the most ADC resolution which is certainly favorable when it comes to determining water quality and/or the presence of biological contaminates in the water. Having 32 different pins with capability to wake the microcontroller is also nice; this would allow for the microcontroller to sleep until the button is pressed to start sanitization.

The largest downside to using this microcontroller is that it is almost too capable for the scope of this project. This project will not require too many I/O features. However, this is ultimately a good problem to have since it will allow for the largest amount of flexibility when deciding upon pin configuration. In other words, there should be no shortage of capability when it comes to PWM, external interrupts, serial communication etc. This microcontroller is essentially guaranteed to work for this project without any sort of functional bottleneck (in regard to pin configuration). Additionally, later on in the project the scope will be more well-defined, and more features will have to be added. So, in selecting this microcontroller, there will be more assurance that the project will still be implementable and manageable in the near future. Lastly, this microcontroller is the most expensive option. But it has a lot of value and timesaving to offer so this appears to be more than worth it. Also, this high price appears to be due to the scarcity of this particular model in the market. It may be wise to look into other variants similar to the 6989 microcontroller that are more available today.

Note: For Team Familiarity and Software Overhead, higher numbers are better.

Microcontroller Comparisons 1

Controller	ATmega328-PU	MSP430G2553IN20	MSP430FR6989IPN
Pins	28	20	80
Active Mode Current	200 µA/MHz @ 1.8V	230 µA/MHz @ 2.2V	100 µA/MHz @ 3V
Standby Mode Current	0.1 µA @ 1.8V	0.5 µA @ 2.2V (LPM3)	0.4 µA @ 3V (LPM3)
Price	$2.63	$3.38	$9.43
Availability	46,400 (DigiKey)	7400 (TI)	< 300 (DigiKey)
Software Overhead	3	2	1
Team Familiarity	3	2	1
Supply Voltage	1.8 V to 5.5 V	1.8 V to 3.6 V	1.8 V to 3.6V
Wired Comm. Protocols	I2C, USART, SPI	I2C, UART, SPI	I2C, UART, SPI
Current Driving	20 mA (Vcc=3V, Vo=2V)	25mA (Vcc=3V, Vo=2V)	25 mA (Vcc=3V, V0=1.5V)
GPIO Pins/Interrupt	23/23	24/16	63/32
Analog Input Pins	8	8	16
PWM Output Signals	6	6	> 6
ADC Resolution	10-bit	10-bit	12-bit
Timing Modules	3	2	5
Memory	32 KB	16 KB	128 KB
Max Clock Speed	20 MHz	16 MHz	16 MHz
Development Board	Arduino UNO	Launchpad	Launchpad

Table 24: Microcontroller Comparisons 1

After assessing the aforementioned microcontrollers, it appears that the MSP430FR6989 is currently the best candidate for this project. However, there are other microcontrollers made by TI that are similar to the MSP430FR6989. In the next section, two more alternatives will be considered for microcontroller implementation in this project.

9.5.7.5 MSP430F5529

This microcontroller was recommended by Dr. Richie. It presumably may still provide similar capability to the MSP430FR6989 in a smaller and more low-cost package. It uses a low supply voltage range from 1.8V to 3.6V. This microcontroller also has low power mode functionality with a slightly higher sleep current than the FR6989. It comes with a real time clock, crystal, and low-power oscillator (VLO). It has a typical standby mode current (in LPM3) of 1.4 microamps when the device is being operated at 3.0V. The clock speed can go up to 25MHz. The microcontroller has four different timers, each having anywhere from 3 to 7 capture/compare registers (CCRs). As far as serial communication, it supports UART, SPI, I2C, as well as even USB. The ADC has 12-bits of resolution, 14 external channels, and there is also a built-in comparator. It has 63 I/O pins, and it comes with 128 KB of flash and approximately 8 KB of SRAM. This microcontroller also has serial onboard programming, and it does not require external programming voltage. This device only appears to come in one package, which is a LQFP 80-pin package.

This microcontroller does appear to be a good choice for project implementation. Most of its feature sets are very similar to the MSP430FR6989 microcontroller that was already previously investigated. It does appear that this microcontroller does have a higher clock speed with a differential of +5MHz over the 6989 microcontroller. It also appears to have more than enough timing capability for this project; four timer modules with a plethora of channels are very useful for timing modes of operation. It can drive enough PWM signals for project implementation. The 5529 also has more current driving capability on its output pins. If any optical components or LEDs were going to be driven directly from the microcontroller, this would certainly be a useful addition for the project. The 5529 also has the same amount of flash memory and ADC resolution as the 6989. Lastly, the 5529 is significantly cheaper than the 6989 microcontroller.

If this microcontroller were to be utilized, an entirely new development board would have to be purchased for each programmer. This would add a significant amount to project expenses. Not only in monetary value, but also in the amount of time it would take to become acclimated with a new launchpad. The 5529 also appears to have less GPIO pins with interrupts, so there may be future limitations on device interrupt functionality. It is also worth considering that this microcontroller has a higher sleep current than the 6989, which will ultimately have an effect in battery life of the finished product. The 5529 is still an 80-pin microcontroller which seems a bit excessive for the scope of this project. It would be more beneficial to find a microcontroller that has fewer than 80 pins. However, the most crucial flaw for project implementation is product availability. Currently, TI is out of stock of the 5529, and DigiKey has 1000 on order. Therefore, this microcontroller would not be a suitable choice for this project.

9.5.7.6 MSP430FR5989IRGC

This microcontroller is in the same family as the 6989 but has slightly less capability. It still has the same wide supply voltage range from 1.8 V to 3.6 V. It also has the same max clock frequency of 16-MHZ. The active mode current draw is 100 microamps per megahertz. The standby current in LPM3 with the VLO is 0.4 microamps. The microcontroller also has 128 KB of flash memory. As far as timing, there is a built-in RTC with calendar and alarm functions; there are also five timers (16-bit) with up to 7 compare/capture registers each. There is a fixed frequency DCO (with 10 different options), a VLO, and low frequency and high frequency crystals. It also supports 16-bit and 32-bit CRC checks. There is a 16-channel analog comparator, a 12-bit ADC, and 16 external analog input channels. If an LCD was ever to be implemented in the project, it is assuring to know that it does have its own built-in LCD Driver that supports 320 segments, as well as contrast control. The microcontroller has 48 I/O pins, of which at least 32 pins have interrupt capability. Again, these interruptible GPIO’s can be found on ports 1-4. There also still programmable pullup and pulldown resistors on all GPIO ports. As far as serial communication, SPI, I2C, and UART are still supported. The same development tools for the rest of the TI MSP430 family, could still be utilized if this microcontroller were to be selected. Both packages of the 5989 are 64-pins, however this one is a VQFN package with the smallest body size out of all 5989 and 6989 microcontrollers.

In comparison to the MSP430FR6989, the MSP430FR5989 does offer a few additional project advantages. One of them being a smaller pack and pin-size which will attribute toward a smaller footprint on the PCB. Since the printed circuit board has to fit inside of a water bottle cap, it is important to conserve as much space as possible. This microcontroller is also significantly cheaper than the 6989 -- by a factor of three. The software overhead for this microcontroller is also comparable to the overhead that the 6989 would have; this is because they are from the MSP430 family, and they could be programmed to use the same ports. The other main advantage to selecting is this microcontroller, is pricing and availability. Currently, it is very difficult to find the MSP430FR6989. It would cost upwards of $10 per unit from a very limited supply across online electrical component distributors. However, the MSP430FR5989 is widely available at TI and for a much more reasonable price.

One of the only downsides to selecting this microcontroller is that it does not have as many I/O pins as the 6989. However, it would appear that the 5989 has more than enough GPIO capability for the purposes of this project. Also, the 5989 does not have as much current driving capability as the 5529. But the 5529 is very scarce in the market, and most likely optical components will be enabled from a MOSFET switching circuit that is connected to the microcontroller.

Microcontroller Comparisons 2

Controller	MSP430F5529	MSP430FR5989IRGC
Pins	80	64
Active Mode Current	290 µA/MHz @ 3.0V	100 µA/MHz @ 3V
Standby Mode Current	2.1 µA at 3.0V (LPM3)	0.4 µA @ 3V (LPM3)
Price	$5.50	$3.09
Availability	1000 on order (DigiKey), 0 (TI)	35,000 (TI)
Software Overhead	2	1
Team Familiarity	2	1
Supply Voltage	1.8 V to 3.6 V	1.8V to 3.6V
Wired Comm. Protocols	I2C, UART, SPI	I2C, UART, SPI
Current Driving	40 mA (Vcc=3V, Vo=2V)	25 mA (Vcc=3V, V0=1.5V)
GPIO Pins/Interrupt	63/16	48/32
Analog Input Pins	14	16
PWM Output Signals	> 6	> 6
ADC Resolution	12-bit	12-bit
Timing Modules	4	5
Memory	128 KB	128 KB
Max Clock Speed	25 MHZ	16 MHZ
Development Board	Launchpad	Launchpad

Table 25: Microcontroller Comparisons 2

9.5.7.7 Microcontroller Selection Summary

It would appear that the MSP430FR5989 is the overall best choice for the project. It has the capability of the MSP430FR6989, with a slightly smaller package size and fewer pins. However, for this project there appears to be more than enough functionality for this microcontroller to be utilized. It also has more than enough timing capability, interrupt GPIO functionality, and low-power modes. The Atmega328P was widely available, but it would have required learning a new development platform, and family of microcontrollers. Ultimately, focus shifted towards the TI family of microcontrollers. This is because the electrical engineers on this team are already familiar with rudimentary TI embedded microcontroller applications. The MSP430G2553, was not capable enough and did not allow for design flexibility in the future. After creating a proposed pinout, it was concluded that this option would be too limited for the scope of this project. It also has a much smaller number of timers, and a larger software overhead than the 5989 and 6989 microcontrollers would have in this project. The MPS430FR6989 is scarce in this market, and it takes up a larger footprint on the printed circuit board. It also appears to be too capable for the scope of this project (in a wasteful rather than practical way). Not to mention it is incredibly expensive in comparison to other project options. The MSP430F5529 would also have worked, but it is unavailable in the current market, and would require purchasing new development boards. So, it makes the most sense to go with the MSP430FR5989. This board is similar to the 6989, which team members are already the most familiar with. It provides flexibility and capability for overall design and project development. It is more affordable and is currently available in large quantities today. Lastly, project members already have the MSP430FR6989 Launchpad development board. This, in tandem with regulated code versions and repository management, will help streamline hardware and software development. In further sections, the embedded prototyping, pinout, and hardware design for this particular microcontroller will be discussed in greater detail.

9.5.8 Microcontroller Development

9.5.8.1 Documentation

It can become tedious to find certain details during the development process, so it is important to always have an idea of which documentation may need to be further analyzed in order to accomplish a certain task. TI provides several different documents that will all be necessary to review at some point. The first is the datasheet for the microcontroller. It contains specific electrical characteristics, as well as device pinouts, pin configurations, and packaging/manufacturing information. The next piece of documentation is the family users guide. This lays out how to program the microcontroller, how to write the code, and how to interface with microcontroller peripherals. The last piece of documentation is the Launchpad user guide. This guide contains specific details about the development kit itself, and how all of the various components are connected and arranged on the development board.

9.5.8.2 Development Board

For this project's breadboard prototype, the MSP-EXP430FR6989 Development Kit from TI will be utilized. This board comes with the 100-pin (IPZ package) MSP430FR6989, which is effectively a larger version of the final microcontroller that will be implemented in this project. Fortunately, TI says that this development kit is compatible for development with MSP430FR5989 microcontroller, which is the microcontroller that has been selected for this project. Both of the aforementioned microcontrollers use the same MSP430 platform and header file, therefore the code will easily be interchangeable between protype and final board layout. Also, it has worked out by circumstance that each project programmer already has this development kit. This ultimately helps reduce overall project development costs.

This development kit allows for simple flashing of the microcontroller via USB connection to the PC. It also has 40 external headers that can be utilized in a breadboard prototype. When selecting the desired protype pin configuration, it will be important to select ports that have external jumpers, so that the breadboard design can be created. In addition to this, the board also has buttons and LEDs that could be used to simulate project components. This would decrease the number of components that have to be placed on the breadboard, and it also help simplify the protype design. The board is also capable of providing 3.3V and 5V DC rails for any electrical/optical prototype components. This also eliminates the need for other external power sources.

9.5.9 Bluetooth Technology Comparisons

9.5.9.1 Bluetooth 2.1

Bluetooth 2.1 is an earlier version, but still appears to be very popular in the industry today. It is a low-cost option, with enhanced data rates up to 3 Mbps. Version 2.1 also improved device pairing procedure (and security); this decision drastically helped to improve user operation. This particular version has a range of 33 meters [13]. Due to the larger power consumption of this version of Bluetooth, it most likely would not be a suitable choice for the project.

9.5.9.2 BLE

Also known as Bluetooth 4, this version of Bluetooth is the first version to drastically improve device power consumption. BLE has a data rate of 1Mbps. It also approximately has the same range as Bluetooth Classic (depending on throughput). This version of Bluetooth would implement lower power consumption and therefore improve the overall battery life of the device. Although this version has a lower throughput, it should be enough bandwidth to push through all of the necessary data for this project. Additionally, a pairing range of 10 feet is more than achievable with this version of Bluetooth. Since this version is intended for devices that only need to send periodic data, this would also indicate that this version would be a great choice for the project [52].

9.5.9.3 Bluetooth 5

This is the newest version of Bluetooth that is available today. It still has a focus on low power features, but also increases the data rate and device range. This version offers speeds up to 2Mbps, but also has a high-range option of 125 kbps. If this version were to be used, most likely the high-speed option would be selected. Since, the range for this product is intended for proximity. This version was developed for an industry that is heavily based in IoT, which has some relevancy with the goals of this project [52]. Bluetooth 5, due to its new arrival, is the most expensive option, and appears to be more than capable for the project. Therefore, it appears unnecessary to go with this version, since BLE seems to already be capable enough for this project.

9.5.9.4 Bluetooth Module 2021 Availability

Previously it was stated that a Bluetooth 4.0 Module would be more than sufficient for this project. However, it does appear that Bluetooth modules (in general) are somewhat difficult to acquire in the market today. Since, Bluetooth 5.0 also has the feature set as Bluetooth 4.0, either version would work just fine on this project. In a market suffering from chip shortage, product availability remains one of the most important deciding factors for this PCB prototype design. Therefore, a mixture of 4.0 and 5.0 Bluetooth modules will be further investigated in regard to design implementation.

9.5.10 Bluetooth Module Investigation

Upon beginning the technology investigation for Bluetooth Low Energy (BLE) modules, it was originally hoped that purchasing a prefabricated unit would allow for relatively simple integration and communication with the microcontroller that has been chosen for this project. It has quickly become apparent that one of the main challenges in implementing Bluetooth capabilities in this project will be programming the firmware that runs on the selected module. As is has been stated many times previously, none of the members of this project team have any experience in employing Bluetooth communications in any previous assignments or personal ventures. Therefore, it is of utmost importance that any module chosen for this project should be as simple and straightforward to use as possible so as to ensure that our desired features can be fleshed out within the given time constraints of this endeavor.

9.5.10.1 CYBLE-013025-00

When this device was first discovered, it seemed to be a standard BLE module that could interface with our microcontroller by either using I2C, SPI, or UART. This model of BLE module utilizes the Bluetooth 4.1 protocol, so we would not be getting the newest and most secure version for communicating with smartphone devices. The module is a Surface Mount Device (SMD) that uses the ARM Cortex M3 as the core and is manufactured by Cypress Semiconductor. Also, the CYBLE unit is listed as being shielded which would protect against RF interference. The device is listed as moisture sensitive, but it has yet to be explored how many SMD BLE modules are available that do not have this limitation.

Upon reading into the datasheet, it could be seen that there were two ways that a user could begin to get working firmware onto the device: by either using the Wireless Connectivity for Embedded Devices Smart Software Development Kit to design the firmware needed for project integration or to take advantage of the EZ-Serial™ Firmware Platform (8). The EZ-Serial™ Firmware Platform is described on Cypress' website to allow the use of their compatible products without the need for prior knowledge of Bluetooth stack or programming [54]. Their website claims that there is no need for an IDE to have "Out-of-the-box support for CYSPP mode with no special configuration"[54]; it is not mentioned anywhere on the webpage what "CYSPP mode" is but the user guide for this firmware defines this as "Cypress Serial Port Profile" mode (11). Therefore, at this point the claim of working "out of the box" seems dubious, but it is encouraging to find that this firmware is already programmed onto the module in question (8). It is also worth mentioning that the documentation on this firmware mentions that it is possible to communicate with the module via UART or GPIO control [54]. Considering the fact that this module has dedicated GPIO pins for SPI and I2C communication (8), it seems to be a reasonable assumption that it is possible to interact with the pre-installed firmware by using these two communication protocols.

9.5.10.2 RN4871 BLE Module

This unit made by the Microchip company is another very tempting option for this project. On the product website, it boasts an ASCII command interface for simple interaction with microcontrollers along with a mobile app for interacting with the device. This module is certified for Bluetooth version 5, which is listed as being able to provide more throughput and more secure connections when compared to Bluetooth 4.1. It has not yet been determined if it will be necessary to have a faster connection to the Android app in order to achieve the desired performance but since the analysis for water quality will be completed by the microcontroller, it seems that the transmission of the final results will not be the limiting factor in the successful project execution. Reading though the datasheet of this device, a recommendation has been found for the proper placement of the Bluetooth transmitter on the PCB that will be created (9). Regardless of which module is chosen to satisfy the wireless communication needs for this design, it has become clear that proper placement of the transmitter will play an important role in the creation of the board in order to optimize the performance of the device. It is difficult to determine the extent to which premade app called Microchip Bluetooth Data will meet the needs of our project without testing but according to the Google Play store description, this app has the ability to "Scan and connect LE device. Transfer text typed in the app to peripheral device. Transfer text file data, send and receive across the device and phone" [55].

9.5.10.3 EYSPBNZUA

This BLE module is over twice as expensive as the Cypress option and in much shorter supply, however it has some very enticing improvements that are worth considering. Firstly, the average current required during data transmission is almost half that of the CYBLE unit while also having a 26.6% lower current draw during low-power-mode. The combination of these two upgrades would certainly mean getting a noticeably longer battery life from the final product.

9.5.10.4 Bluetooth Module Comparison/Selection

When examining the three options that have been investigated, the biggest factors in the final choice of module have been gathered in Table 26 below for convenience. It was highly desired to utilize the RN487x series of modules due to the pre-made smartphone app that could have potentially eliminated the need for app design and coding entirely. However, it was found that the availability of these chips is just not going to meet the time constraints of this project.

Therefore, the final two options need to be evaluated based off of the needs for this project. It would be much more energy efficient to go with the EYSPBNZUA module, but there have been major concerns about whether this energy demand decrease would be worth the price being over twice that of the competitor. Considering the fact that the module will only being transmitting for very limited time frames after the water sensor has been activated, it seems that the drastic difference in transmitting supply current would not lead to much of a difference in terms of battery life for the device as a whole. It was also discussed that the exclusive use of Bluetooth Low Energy mode for this project would mean that the Bluetooth 5 functionality would not even be utilized. The Cypress module is also much more available in terms of stock, and this is a reassuring factor for the ability to actually obtain the module. It would be very damaging for the time constraints of this project to dedicate a significant amount of effort into investigating how to implement a Bluetooth module for it to end up being bought out from underneath us and to start from scratch to figure out how the Cypress module works. Finally, it is worth mentioning that the documentation for the EYSPBNZUA module does not mention any firmware or API that is designed for simple and out-of-the-box operation.

In conclusion, it seems that the CYBLE-013025-00 BLE module is the best option for the needs of this project. There is still a good amount of technology investigation that needs to be carried out for the successful operation of this module. In the following sections, it will be discussed how to use the firmware that comes installed in this Cypress module as well as the potential need for an evaluation board for the development phase of this project.

BLE Module Comparisons Table

Controller	*CYBLE-013025-00*	RN4871	EYSPBNZUA
Supply Current Transmitting	26 mA	13 mA	14.2 mA
Lowest Power Mode Current	1.5 µA	60 µA	1.1 µA
Price	$5.94	$7.38	$12.70
Availability	881	16	76
Shielded	Yes	Yes	Yes
Moisture Sensitivity Level	3	2	3
Bluetooth Protocol Version	4.1	5.0	5.2
MCU Interface	I2C, UART, SPI	UART, AIO, PIO	I2C, UART, SPI
Earliest Shipping Arrival	2^{nd} day, overnight	December 24, 2021	2^{nd} day, overnight

Table 26: Bluetooth Module Comparisons

9.5.11 *CYBLE-013025-EVAL EZ-BLE™ Module Arduino Evaluation Board*

The EZ-Serial WICED BLE Firmware Platform User Guide specifically recommends using the appropriate evaluation board for learning and developing with the CYBLE-013025-00 module (11). Being able to connect this evaluation board to Tera Term or any other serial terminal software that is available for a PC would greatly expedite the process of team members familiarizing themselves with the Application Program Interface (API) that is available for interacting with the selected Bluetooth module. Since none of the members of this project team have any prior experience in programming for devices to have Bluetooth interconnectivity, it seems highly prudent for at least one of us to begin learning and preparing code for the microcontroller to interact with the Cypress device. This evaluation board is listed at $50 on Cypress' website, but it is also mentioned that there is a possibility to obtain samples or reduced rates for students. An email has been sent out to a Cypress sales representative for further information, because adding $50 of development cost for just one evaluation board would be a very steep inclusion for a list of production costs that is already rising rapidly from the initial estimates.

9.5.12 *EZ-Serial WICED BLE Firmware Platform*

This preloaded firmware was one of the main selling points for the Cypress BLE module that was selected, yet it will still take a fair amount of effort to ensure that the default settings are going to provide the necessary functionality for this project while also learning how to properly interact with the device on both the microcontroller end as well as on the intended smartphone app. It is possible to install a custom firmware image on the CYBLE-013025-00 if the standard platform does not meet our needs by using the WICED Smart SDK Integrated Development Environment (IDE) (11), but this will only be a last resort measure since it will take a considerable amount of programming overhead.

Parsing through the firmware user guide, it can be seen on page 8 that the firmware on this module has some settings initialized that are important to remember for all coding purposes (11), and these settings are shown below (see Table 27).

EZ-Serial WICED BLE Firmware default UART Settings

	Baud rate	Data bits per frame	Parity bit enabled?	Number of stop bits	UART flow control
Default Setting	115,200	8	No	1	Disabled

Table 27: Default UART settings for Cypress BLE module

Unfortunately, the selected module does not support a text-based output for the API and the units that do support this mode are currently not available due to the limited stock. This means that all of the information that gets passed to and from the BLE module via the built-in API will be shown in the serial terminal as a string of hexadecimal values that describe the API event that has occurred. Thankfully, it is mentioned on page 9 of the user guide that there is a Python script that is available for free on Cypress' website which will convert between text commands and the binary (hexadecimal) messages (11). The particulars about how this Python code works is explored in a later section of the user guide and this report.

There are two main modes that the device can operate in that are selectable via the CYSPP GPIO pin by passing either digital low for Cypress Serial Port Profile (CYSPP) mode or digital high to activate command mode (page 10, 11) . Communicating with the device in command mode is done by sending binary (hexadecimal) data packets with a format that is described on page 11 of the user guide (11). More detail can be explained for command mode if it is necessary, but it seems that CYSPP mode will be the method through which we will be able to actually send data over a Bluetooth Low Energy connection (page 14, 11). CYSPP mode can be activated in three different ways, but it seems that it will be easiest to do so by passing a digital low to the CYSPP pin via the microcontroller.
When operating in CYSPP mode, the Cypress module will only be able to operate in a peripheral role (page 14, 11). It still needs to be investigated if this will be an acceptable

setting due to the fact that the microcontroller will be the unit to dictate when new data is ready to be provided to the smartphone end of the connection. Again, if this ends up not offering the functionality that is needed, it is possible to create a custom program by using the WICED Smart SDK IDE. When a device that is not made by Cypress is to be used to serve as the central role in the BLE communication with the module, the user guide indicates that it needs to be configured to follow a specified procedure. There are examples given for configuring a device to connect to the module in CYSPP mode in section 3.2 of the user guide (11).

When the Cypress module starts up in CYSPP mode, it begins “advertising” with the preconfigured settings. According to the Bluetooth website, advertising for Bluetooth Low Energy is either done to establish a two-way link between devices or to broadcast data without establishing a connection [56]. The EZ-Serial user guide then says that once a wireless connection has been made between the module and the other device (in this case it will be a smartphone), the smartphone must “subscribe” to either acknowledged or unacknowledged data (page 15, 11). It is not yet clear how the connected device makes this choice, but for now we will reserve all discussion on the specifics of operation on the smartphone end to a dedicated section for programming the smartphone app. The device is also given an optional choice to enable receiver flow control so that each end can communicate when it is ready to write new data (11). For this project, we will want to select both acknowledged data as well as RX flow control in order to maximize the reliability of data transfer. Choosing these settings does reduce the potential throughput of data, but the speed at which the app responds is not as important as the integrity of the data when said data indicates the potability of water. After the connection has been established, the EZ-Serial firmware asserts the “CONNECTION” pin in order to tell the microcontroller that it is ready to transmit and receive data (page 15, 11). There are multiple ways listed to close the data pipe between the Cypress module and smartphone, but the option of interest for this project will be to maintain the connection until the smartphone disconnects.

Moving on to the operational examples in the user guide, there is a detailed discussion about how to select baud rates, enable flow control, and manage the sleep states of the Cypress module by interacting with the API vie UART in command mode. Most of this information is of interest but will be reserved for discussion during the project design portion of this report concerning any changes that might be made to the standard settings of the EZ-Serial firmware. It is described on page 27 of the user guide that since the factory default for the BLE module firmware is configured to be in “auto-start” mode, the Cypress device will begin automatically advertising to connect to the smartphone as soon as it is powered on (11). The guide also indicates again that the serial data stream is bi-directional, which means that even though the Cypress module can only function as a “GAP Peripheral” with the EZ-Serial firmware, this will still allow the module to take in data over UART from the microcontroller and to send it via the BLE protocol to the smartphone app.

Much of the rest of the user guide for this firmware describes commands that can be sent to the Cypress module API while the device is in command mode. It does seem possible to update data stored in a "local GATT Server attribute" on the BLE module by using the "gatts_write_handle" command and then tell the smartphone client that this data has been updated by using the "gatts_indicate_handle" command (pg.92-93, 11). However, the module does not appear to automatically connect to a client wirelessly while in command mode, so it seems that the way to utilize this automatically established data pipe would be to only operate in CYSPP mode. The entire firmware user guide has now been investigated and there does not seem to be a clear answer as to how this data pipe between module and smartphone can actually be used by the programmer to send and receive information. The only hint at how the CYSPP transmission actually works was found in the description of the "p_cyspp_set_packetization" command which allows the microcontroller to dictate how data coming into the BLE module over UART is formed into packets for CYSPP transmission (page 108, 11). Since we will be sending values of a fixed data type to the smartphone as indicators of water quality, it seems to make sense to use "mode 2: fixed" setting for CYSPP transmission packetization. If the module waits until it has received a fixed number of bytes over UART from the microcontroller, then we will know for sure that each data packet being sent to the smartphone app will have that many bytes of useable data to be extracted, interpreted, and displayed. It seems inevitable that there will need to be intensive investigation into how a data stream is interacted with in the chosen smartphone app environment, but reading through this user guide has at least clarified that the selected Bluetooth module will be able to connect to a smartphone and transmit data packets that are sent to it via UART.

9.5.13 Bluetooth Module Hardware Implementation

The PCB design and pinout connections to the microcontroller both need to account for the GPIO pins of the Cypress BLE module that we will be connecting to. The intention of this project is to keep the communication between Bluetooth module and microcontroller as simplified as possible in order to minimize the programming overhead that needs to go into the Bluetooth communications aspect. However, it is highly prudent to prepare for more the more advanced interoperability between these two devices in case a need arises for the more advanced functionality that the Cypress module is capable of. The bare minimum number of connections that would need to be made between microcontroller and the BLE module would only be two lines for transmission and receiving. In the design of this project, it has been decided that incorporating the "Connection" and "Data Ready" outputs from the BLE module would be sensible for the purpose of writing more robust communication code for the microcontroller. Since we have already selected a microcontroller that has plenty of GPIO pins to spare, it seems prudent and wise to include connections to some of the BLE module GPIO pins that we do not initially intend to use. In this way, we will be leaving ourselves the opportunity to implement some of the more advanced features of the module, if the need does arise. Therefore, the inclusion of connections for the flow control (CTS and RTS), mode selection (CYSPP), low power mode, and low power mode status pins is a precautionary measure that is being taken to reduce the impact if the project requires redesign further down the line.

9.5.14 Wired Communication Protocols

Upon investigating the various options for Bluetooth modules available on the market, it has become clear that there are multiple choices for how we will pass data from the microcontroller. In order to maximize safety and performance of the water quality sensor, it is important to ensure that we can minimize the risk of data corruption during the transmission process from the microcontroller to the BLE module, and to the smartphone application.

9.5.14.1 UART

A few of the Bluetooth modules that have been explored as possibilities for this project have suggested that using the simplified firmware that comes pre-installed on the devices will limit the communication to said device to the Universal Asynchronous Receiver/Transmitter (UART) protocol. UART may be one of the most simplified wired communication protocols, because it is a form of serial transmission that only needs two wires between the host and client [57]. Unfortunately, simplicity can sometimes mean that sacrifices are made in terms of the quality of the data transmission.

The greatest concern that needs to be addressed if the UART protocol is going to be used is ensuring a high degree of integrity of the communication between the microcontroller and the BLE module. The asynchronous aspect of UART transmission means that the bitstream being sent between host and client are not being synchronized with the frequency of the clock signal; instead, there is a sampling method that the receiving end of each communication uses that is actually driven by the clock signal [57]. There are fixed baud rates that have been decided upon that are used by any UART implementation [57], which simplifies the process of choosing speeds for the client and server. However, neglecting to ensure that the two devices using UART have been initialized to transmit and receive at the same baud rate will inevitably lead to sampling errors in which the receiving end is not properly collecting the incoming bitstream [57]. In case data corruption does occur during transfer, the UART protocol includes a parity bit that will be used for the receiver to check if the number of ones or zeros is odd or even [57]. All of this is to say that although there is a very basic form of error checking in the UART protocol, it is a far cry from more advanced methods of communication that can not only detect when an even number of errors has occurred in one data frame but can also correct the errors that did happen. Nevertheless, it is still worth mentioning that this protocol can maintain much better data transmission integrity if the option to use oversampling is available and turned on [57]. It would be highly desirable to use an oversampling rate of at least 8 or 16 in order to safeguard against the data being transmitted incorrectly.

This project will not need to send a large amount of data through the Bluetooth module and these signal broadcasts will typically be in bursts with long periods between a bottle refill and reevaluation of the water quality. Therefore, it is not a large concern that the bitstream coming from the microcontroller into the BLE module will utilize the full potential of the Bluetooth data transmission rate. The utmost importance of all that has been discussed in this section is to emphasize the need for preserving the integrity of the data being sent to the smartphone app. Even if the microcontroller properly analyzes a water sample and indicates that it is unacceptable via the status LEDs on the unit; if the data were corrupted over the UART transmission even once, this could potentially confuse a user into thinking that the LED was wrong because the readout on their phone tells them that the water is fine. This might seem like a lot of fuss over too simple of a matter, but even the chance of inappropriately telling a user that non-potable water is safe to drink could result in disastrous consequences.

9.6 Microcontroller Software Implementation

This section will outline various microcontroller/development board features that will need to be utilized throughout the project. Prior to prototyping, it is important to first understand how each type of connection/microcontroller peripheral works. This will then serve as a helpful reference for team members writing code later on during more advanced software development. Not only this but outlining how to implement various microcontroller features will make it easier to identify sub-goals for software development. Also, Compartmentalizing the codebase into smaller, more identifiable, and more manageable pieces will help smooth out embedded software development.

9.6.1 GPIO Pins

The most common pin configuration is a GPIO pin. The General Purpose Input/Output's or GPIO's in the MSP430 are categorized as ports and pins. In PX.Y, 'X' represents the port number, whereas 'Y' represents the pin number. Here, the term 'pin' is not synonymous with the general understanding of a pin number on an integrated circuit. In other words, there would appear to be duplicate pin numbers here, but the ports are what distinguish the pins and make them all unique. Each port has several registers that need to be written to in order to configure any pins. The register is a bit-wise representation of each pin on a selected port. For example, the 8 bits from right to left in port 1 represent pins 0-7 respectively. The most common way to configure the GPIO's involves utilizing bitwise operation and/or masking to set and clear certain bits in a register. When performing bitwise operations, it is always helpful to utilize the "BITx" symbolic constants in the MSP430 header file. Then, a define statement can be created at the top of a program, and a more appropriate name can be used for the bit mask.

Configuring a GPIO pin simply involves setting or clearing the respective bit in the PxDIR register. Here, the "x" represents a port number on the microcontroller. Setting a bit in this register will direct that pin as an output, whereas clearing a bit makes the pin an input. All pins are configured as high impedance inputs by default. The next register of importance is the PxOUT register. This is the register used to output a logic level high or low on general purpose output pins. And the last register is PxIN, which is used to read a logic level high or low from input pins. Ports 1-4 on the microcontroller, have interrupt capability. This means that they can wake the microcontroller up from low power mode.

9.6.2 GPIO Interrupt (LLWU) Pins

When configuring a GPIO pin to be able to interrupt the microcontroller, again another register has to be interacted with. First, the pin must be configured as an input by clearing the corresponding bit in the PxDIR register. Then, it is always helpful to utilize the built-in pullup and pulldown resistors inside of the microcontroller. This can be done by setting the corresponding bits in the PxREN (Resistor Enable) register. Alternatively, this register could remain 0 by default, and external pullup and pulldown resistors can be used. The type of resistor is configured in the PxOUT register that corresponds with the port that contains the desired interrupt pin. Setting a bit will make the resistor a pull up. Whereas, clearing a bit will configure the resistor to be a pull down. The next register, PxIES (Interrupt Edge Select), is used to control the type of edge that triggers the interrupt. A one is a falling edge trigger, and a zero is a rising edge trigger. Next, it is always a good idea to clear the flag (that will normally be raised by the interrupt) so that the interrupt event does occur instantly; this is accomplished by clearing the corresponding bit in the PxIFG resistor. It is always important to remember that interrupts (even if individually enabled) will not execute until the Global Interrupt Enable (GIE) bit is set. This can be accomplished with the function "_enable_interrupts"

9.6.3 Interrupt Service Routines (ISRs)

Fortunately, interrupts are kept relatively simple since they are handled by the hardware. After an individual interrupt event has been configured and enabled, as well as the global interrupt bit has been set, the MSP uses a vector table to handle all of its interrupts. Each interrupt service routine has to have its own programmed function that follows the main function in the code. Because of this, every time an interrupt occurs, the hardware knows which operation to perform next. Some interrupt service routines are shared, in this case it is important to always be able to identify which event triggered the interrupt through the use of if-conditional statements. The hardware will automatically clear the flag after an individual interrupt service routine; however, for shared ISRs, it is the programmer's responsibility to clear the flag (designate that the interrupt event has been handled).

9.6.4 Non-GPIO Pin Configuration

Other pin configurations require using a selection register or multiple selection registers to configure non-GPIO functionality. Some of these features may include: PWM output, ADC input, UART Tx, and UART Rx. To do so, it is always best to consult the datasheet to determine which masks need to be used. For example, in order to configure P1.6 into a timer output channel for a PWM signal, the datasheet outlines the following conditions: set bit 6 in P1DIR, set bit 6 in P1SEL1, and set bit 6 in P1SEL0. In order to perform the aforementioned operations, the declarations for each register can be found in the MSP header files. Each pin that is going to have non-GPIO functionality must be configured in this way.

9.6.5 MSP430 Driverlib

This is an API (Application Programming Interface) that is provided by TI. Application Programming Interfaces typically allow programmers to manipulate lower-level software by using higher level logic and a library of pre-configured functions. It appears to simplify many microcontroller tasks, as well as abstract register operations. Using this API would help minimize bitwise operations and masking. However, this API does appear to have some complicated software implementation. It would require creating data structures which would further convolute the code. It most likely is wiser to write the software without this API. This is mainly because team members are already very familiar with low-level register operations and coding. So, this would have a smaller overhead than trying to learn a brand-new API from the ground up.

9.6.6 Timers

The MSP430FR6989/5989 microcontroller comes with a copious amount of timing modules. Each timer is capable of running at its own frequency with a multitude of different channels. There also different modes to operate the timer in, such as: continuous mode, and up mode. Channel configuration registers must be written to with a number of different masks in order to configure the timers. The first step to using a timer is selecting the clock. The MSP has certain default clocks, but these can be reconfigured in the code. After the clock has been selected, an input divider is then used to set the period. The input divider can divide the clock frequency by a factor of 1,2,4, or 8. This is used to generate more optional frequency references for the timer. From here the timer will either count up to 65,536 cycles in continuous mode, or another number of cycles set by the user in up mode. This number is limited to a resolution of 16-bits. Channel zero of the timer always specifies the number of cycles (zero-referenced) counted up to during up mode. Each timer comes with different capture/compare registers/channels (that also have external microcontroller output pins). These can then perform logical operations in different amounts of cycles/time.

When trying to create a PWM signal, channel zero sets the period of the signal in up mode. And other capture and compare registers, can be used to change the duty cycle and/or shape of the signal. Consult the documentation for more information about all of the different types of PWM signals that the MSP430 microcontroller is capable of generating.

9.6.7 Low Power Mode

The MSP430 has many built-in functions for entering low power modes. "_low_power_mode_x" is the best way to easily enter any low power mode. Here "x" represents a number from 0 to 4. This function automatically enables the selected power mode and activates the GIE bit so that the microcontroller can wake up from sleep. Higher value of low power mode results in a lower sleep current. Additionally, various low power modes have different clocks disabled/enabled. For example, low power mode

3 only uses ACLK. Whereas low power mode 0 utilizes the DCO, SMCLK, and ACLK. For more information, view the family user guide to determine which power mode is correct for device modes of operation. Note: Only use low power mode 4 if an external GPIO pin is able to wake the microcontroller up from its sleep.

9.7 Structural Design

It will be necessary to create a structure that can house the entirety of the components while conforming to a convenient form factor for users to easily handle. Along with being easy to handle, this cap will need to be structurally rigid enough to withstand the grip of users while they remove and secure it into the bottle multiple times a day. This casing will need to be able to be screwed into the water bottle in order to securely close the cap in place, which will not only keep water inside the bottle but also activate the Reed switch to allow operation of the laser features. The consensus among group members is that it would be better to adjoin a custom casing of our design onto the bottom part of the metal cap that comes with the bottle. This technique will be advantageous for this prototype design being created for this class due to a number of different factors. Firstly, it will greatly aid the operation of the laser features to have the bottom-most part of the cap, which sits inside of the bottle, be a reflective metal in order to minimize absorption of the lasers.

It will also drastically ease the design of the cap casing if it is not necessary to properly recreate the part that screws into the bottle. Even if a custom cap is created that does screw into the bottle, it will still be desired for the screw part to be metal, but this cannot be the composition of the entire cap casing. The biggest challenge would be actually finding a way to get a custom-made metallic cap based off of a CAD design. If a company were willing to create small batches of these custom metallic designs, it would most surely be expensive due to the production process. This thought process has led us to conclude that it would be much more practical and beneficial to utilize the 3D printers that are available in the UCF design labs.

9.7.1 Original Prusa i3 MK3 (3D Printer)

It will be helpful to be able to design and create our own casing on the UCF campus because it will allow us to create an updated design cheaply and quickly if there is a need for revision. It still needs to be investigated if there are resources on campus for training to use these printers. Otherwise, it will be necessary to take advantage of the tutorial videos and literature that is available for free at prusa3d.com.

9.7.2 PrusaSlicer (CAD)

The Computer-Aided Design (CAD) software that is used for designing and printing with the Prusa brand 3D printers is called PrusaSlicer and can be downloaded for free at prusa3d.com. This software is compatible with Windows, MacOS, and Linux operating systems, so this will not cause any issue for team members to be able to work on the 3D modeling on their own. Although we have not used this software before, it is most likely that students are able to simply bring a digital copy of the CAD drawing into the design laboratory in order to print the 3D model.

9.7.3 Prusament (Filament)

There are many different choices of filament that can be used with the Prusa 3D printer, and it will be important to choose a material that is optimal for this bottle cap design. As previously mentioned, this structure is going to be twisted by users on a very regular basis and therefore it will be necessary to select a material that has high shear strength. This casing will also need to provide support to the electrical and optical components that will rest inside of it and so an increased level of hardness is also a factor that will need to be considered. In order to not interfere with the wireless charging system, the chosen material will ideally have a low conductivity, so as not to induce large Eddy currents in the casing itself. Finally, it is also highly desirable that the bottle cap is not heavy because this will increase the chance that a user might accidentally drop it and damage the components within. Since density is typically correlated with a higher strength and hardness, there will need to be some optimal compromise in the material selection process. It would be helpful to begin this search by inquiring about the campus availability of filament. If there is a selection of complimentary filament for the 3D printers on campus, it might be best to take advantage of an opportunity to cut down on the project budget.

9.8 Application Software

9.8.1 iOS App Development

Upon investigation, it appears that the common development environment for creating iOS apps is called Xcode and can be downloaded for free on any machine running macOS 11 or later. The coding language for iOS apps that was developed by and endorsed by Apple is known as Swift, and one can begin developing an app in this language using the SwiftUI interface within Xcode. Unfortunately, there is limited support for macOS 11 on Apple computers released before 2013 [58], and this is a direct limitation on our ability to use Xcode for iOS app development.

9.8.2 Android App Development

The official Integrated Development Environment for creating Android Apps is known as Android Studio [59]. This IDE provides a lot of flexibility for developers because it is available for Windows, macOS, and Linux. The official programming language for developing Android apps used to be Java, but it was replaced by a newer language known as Kotlin in 2019 [60]. Kotlin is considered to be easier for beginners to code in compared to Java, and this is a promising feature for this project when considering that none of the group members have experience in programming an app for smartphones.

9.8.3 Object-Oriented Programming Basics

Since most of this group has not coded in Java or Kotlin before, many of the object-oriented functionalities of these languages are being learned through the lenses of students who have hitherto written code mostly in C. The more that online guides and references about programming for Android have been explored, it has become increasingly evident that there is a need to investigate some of the most basic concepts and terminology that are used in object-oriented programming such as Kotlin.

The titular feature of object-oriented programming is the software object itself. In terms of coding, an object is a “bundle of related state and behavior” [61]. These states of an object are analogous to a value that describes it, such as name or color, while a behavior is an ability that the object can perform [61]. The various states of an object are stored in fields, which are similar to variables, and the behaviors of an object are “exposed” through its methods, which are like functions [61]. Therefore, an object is a piece of software that has fixed methods through which it can change its states and interact with any other code.

Objects in programming languages like Java are created from a prototype called a class [61]. A class in object-oriented programming is similar to a schema in psychological terms in that creating an instance of a particular class is how we consistently attribute certain properties to objects of that class type. Knowing how a class is defined allows us to see an objects functionality that it can perform and the potential states that it can have.

The concept of inheritance leads to the idea of a higher level of classes known as a superclass. Grouping similar classes together underneath a superclass allows each of these subclasses to inherit common features the superclass that they are all categorized in, and this is done in Java by using the keywords “extends” to reference a particular superclass in the new class declaration [61]. This is a coding concept that is very useful for creating new classes that are based off of a pre-existing superclass so that the programmer can focus on the added functionality that they are including to define their new class.

It has already been discussed that an object “exposes” itself to software via methods that are a part of their definition, but these methods can be identified as an interface [61]. Interfacing with an electrical object in the physical world is typically done through some sort of control panel and defining an interface in object-oriented programming is a way of laying out a grouping of methods that can be called upon in a class declaration via the keyword “implements” [61]. This idea of an interface is a way of standardizing and formalizing the ways in which anything can interact with a defined class.

The final concept that is presented in the Java tutorials being referenced as [61] is a “package”. In Java, the term package refers to a grouping of interfaces and classes that are all related, and this is a way of organizing them [61]. The multitudinous number of classes within Java and Kotlin gives rise to the need for further organization of these pieces of code. The term Application Programming Interface (API) is what programmers call the library that contains packages of classes and interfaces that form the building blocks for any code written in that particular programming language. Now that these rudimentary terms of object-oriented programming have been covered, it is much easier to follow the discussions online about how we may proceed in using a given API to create the Android app for this project.

9.8.4 Android BLE App Development

In order to successfully create an Android app that can communicate with a Bluetooth Low Energy module, it is necessary to explore some basic terminology and programming design considerations. Without getting bogged down with information that will not be necessary for the execution of this project, it has become very clear that there are certain terms that need to be discussed in order to make even the humblest beginning in BLE coding. The term that seems to show up most often in BLE discussions is GATT, which stands for Generic Attribute profile, and this describes the formatting of services and characteristics, as well as the processes for interfacing with these attributes [62]. A GATT Service is a grouping of related characteristics that refer to a particular device feature and a GATT Characteristic is basically a data field that can be edited and read from [63].

The Cypress BLE module will be acting as a peripheral or server that will store the values of water quality within a GATT characteristic maintained on the module. After the microcontroller updates the values of the water quality within the BLE module, the module can send an indication to the smartphone (acting as the central or client) that will tell the client it may ask for an updated value of the characteristic (data field). It was previously explored how the Cypress module could update an acknowledged GATT data characteristic (via UART from the microcontroller) that the smartphone has subscribed to. Since we will be using the acknowledged data characteristic, this should push an indication to the client (smartphone) to say that the data is ready to be sent. An indication is similar to a notification, except that it means that the data packets received by the smartphone will need to be acknowledged in order to guarantee their delivery [63]. This is the general process that will be undertaken in order to transmit water quality data to the smartphone app, but there is still a need to discuss how the Cypress module and the Android app are going to connect in the first place.

The Android BLE guide by Chee Yi Ong [63] specifically recommends creating an app that is designed for Android 5.0 (API 21) or newer, due to the better BLE Application Programming Interfaces (APIs) that are available. It is also explicitly stated within Android Studio that Android 5.0 Lollipop was created with support for Bluetooth Low Energy. Android Studio also indicates that using newer APIs will reduce the percentage of phones that can run the app, so it seems that API 21 will be the best option to use for developing our app.

Looking further into the BLE guide by Chee Yi Ong [63], there is a list of classes that are suggested to be very useful in the process of coding in Kotlin for the integration of BLE connectivity. Reading about these various classes and their uses offers insight into how one can actually code an application to interface with a device via Bluetooth as well as what steps are necessary to successfully find and connect to the device in the first place. The “BluetoothAdapter” class is what represents the Bluetooth hardware on the smartphone itself [63]. Interfacing with an instance of this class can provide the programmer with information such as which Bluetooth devices are bonded to the Android phone along with supplying the capability of start a scan for BLE advertisements. The class that can start a BLE scan is called “BluetoothLeScanner”, and this is provided by the “BluetoothAdapter” class [63].

Once a scan for BLE advertisements has been initiated, it is possible to filter the results based on the Universally Unique Identifier (UUID), device name, MAC address, service data, or manufacturer specific data [64]. It will be necessary to query the Cypress module from the microcontroller via command mode in order to obtain any of these identifying pieces of information in order to let the Android app know which device it should be looking for. If this application was going to be used to connect to one of any smart water bottles being mass produced based off of this project, we would probably filter results based off of the manufacturer specific data. However, for the purposes of designing a prototype device for the satisfaction of this project’s requirements, we will most likely use either the UUID or device name to find and connect to the Cypress module.

Once a BLE scan has been narrowed down based on the known device identifier, the device advertisement will be represented by the aptly named “ScanResult” class [63]. Using the “getDevice()” method we will be able to expose the “BluetoothDevice” handle, which is a class that represents that actual Bluetooth device in the Kotlin/Java code, and this will also enable the application being created to actually connect to this device [63]. The “BluetoothGatt” class is what will be used for interfacing with the Cypress module’s GATT profile so that we may access the services and characteristics of the BLE device. As discussed in the EZ-Serial firmware guide, it will be necessary for the application to subscribe to either the acknowledged or unacknowledged data characteristic that will be exposed after connecting with the Cypress module in CYSPP mode [11]. It is the intention of this project to prioritize data integrity over application execution speed, so we will be using the “BluetoothGatt” class in order to subscribe to the “BluetoothGattCharacteristic” of the Cypress module, which will most likely be called “Acknowledged Data”. Finally, the interface that will be primarily utilized for getting callbacks about characteristic indications and reads will be “BluetoothGattCallback” [63].

The term “callback” was not made clear within the previous resource that is being used, so it is necessary to discuss what a callback is. Programming languages like Kotlin use callbacks as one way of dealing with asynchronous communications or subroutines [65]. Basically, when a program is waiting for some function or communication to be completed, it is standard practice to avoid “blocking” the program from continuing to do anything else while it is waiting [65]. Using a callback happens when a function is passed to another function as a parameter with the intention of returning upon completion of the desired task and the reception of a callback [65]. The idea seems to be similar to the concept of an interrupt, where we may continue processing other things in the program while we wait for a callback to inform the program that the previously requested task has been completed and so the program may return to processing the result that it had asked for. Since BLE hosts only send information to a client whenever it is ready to send an updated data packet, it makes sense that a Kotlin program for an Android app would need to utilize callbacks as a way to prevent blocking of the program while it waits for any communication from the BLE.

This section will end by discussing the some of the major warnings that the guide by Chee Yi Ong [63] has laid out in preparation for creating a Kotlin program to handle BLE communications. The first suggestion is that operations should be coded to queue up and only be executed one at a time [63]. Each time a request of any type is made by the program, we will want to wait to receive a callback for verification before allowing the next operation to take place. Also, it is suggested that one should keep in mind that any operation being executed has the chance to fail and that a good program should be able to deal with such failures as they occur [63]. Error handling is a process that is essential to creating code with any level of robustness and having the program to methodically queue its operations will ensure that we are properly processing and receiving all of the data as it is transferred to the app from the BLE module.

9.9 Version Control Systems

Version control systems (VCS) are a functional upgrade from shared network drives, enabling more effective cooperation among teams on large projects. This is primarily due to their user's ability to easily explore more unconventional ideas by using branches while leaving a 'save point' in the form of a previous version. These systems also have what is essentially immunity to accidental deletions, as any missing files can be reacquired with minimal effort and panic- a significant improvement over the days of using volatile memory and an uninterruptible power supply at the office as storage. There are several popular platforms for version control including GitHub, AWS Code Commit, and Azure DevOps. All three of these use Git to handle repositories.

9.9.1.1 GitHub

GitHub is a popular VCS owned by Microsoft and commonly used by university students due to its compatibility with most integrated development environments (IDE). Users of GitHub can upload files directly to the website, use the VCS's app, Git, or use an IDE with Git built in. Once a repository is created it can be 'pushed' to (files from the user's computer are uploaded), 'pulled' from (files from the website are retrieved), and reverted to previous instances, as the website saves enough information to 'undo' changes between pushes. This ability to revert changes is tantamount for programming, as procedural mistakes can be more easily fixed without the need for redundancy on the user's side.

9.9.1.2 Azure DevOps

Azure DevOps, also by Microsoft, expands upon the VCS base of GitHub with additional administrative utilities and training geared towards software development. Its popularity resides in the corporate world more than the student realm although both are used by each sector due to the contrasting origins of the two systems. While heralded as the successor to GitHub, uptake is slow even at Microsoft itself, which uses GitHub for most of its open-source projects [66].

9.9.1.3 AWS Code Commit

AWS Code Commit is Amazon's take on a VCS, utilizing Git to manage user-side repositories as well. Its primary advantage is reduced cost for private repositories compared to GitHub, which is not applicable to an open-source project.

9.9.1.4 Version Control Summary

Of the version controls explored, none have a strong advantage. AWS Code Commit lacks the administrative features found in Microsoft's contributions, however, making it less desirable. Of the three, GitHub is the most attractive due to greater familiarity with it, and the ease with which it can be linked by most IDEs.

9.10 User Operation

While product durability, effectivity, and sustainability are largely determined by the design and manufacturing process, certain operating practices and incidents can affect the device's functionality in detrimental or even hazardous manners. Some such incidents can be mitigated during the design phase, but others require user cooperation and attentiveness.

9.10.1 Ice Cube Usage

When researching about UVC and its ability to sanitize water, a question about the presence of ice cubes in drinking water was raised. Would the device user be able to put ice in the water? In order for this to be acceptable, the UV radiation would have to be able to sanitize not only the water, but also the ice cubes. A study conducted by Ladanyi, and Morrison concluded that the UV was able to penetrate at least 19 cm of ice while killing the bacteria. However, penetration depended heavily on the optical qualities of ice [67]. Since the consideration of ice would greatly complicate the process of testing and verifying sanitization, the device user should be strongly advised not to include any ice in the water bottle. Therefore, the project scope will focus on sanitizing water without the presence of any ice.

9.10.2 Protection from Impact

One major concern for use of this product will be the lack of shock resistance of the unit. Water bottles and the caps that seal them shut are items that can normally withstand being dropped on a regular basis. This is especially true for uses that involve more active and vigorous activities; it is a very common occurrence that users will drop a water bottle either intentionally or on accident. However, the components of this design will not allow for such impact at all and such shock to the unit could very easily destroy the water quality sensor. Although there will be a fair number of solid-state components that are used for making this project model, the elements of the water quality sensor are going to be highly sensitive to any sudden force. This susceptibility to shock is a quality that it inherent in the design of the spectrometer implementation for this project. The proper alignment of the LED, prism, servomotor, and other optical components for the current model of this water analysis sensor is going to need to be fine-tuned in the calibration process in order to get consistent results that have any meaning. Being able to build a structure that will hold these pieces in their proper place is already going to be enough of a challenge, as there is a limit to the types of materials that we will have access to for creating a casing with the 3D printer that is available. Compounding this limitation of material selection with the fact that none of the members of this team have much experience with using CAD software, it is easy to see that the most reasonable scope of this project will be a design that is not able to withstand being dropped. If future versions were to be developed after this initial prototype, creating a more physically robust design would be one of the first priorities due to the aforementioned danger of such shock.

9.10.3 Water Temperature Range

The bottle content's temperature will, of course, fluctuate- it is unreasonable to expect users to repeatedly fill their bottle with water that is a specific temperature, let alone a unique temperature set by the project development team. The temperature of the bottle's contents directly impacts all system functions to a certain degree, although it affects the optical components and battery most of all. The battery, as with all electrical storage components, will not tolerate boiling temperatures.

To that end, hot water (that is to say, water above body temperature) should not be used in the device. Hot water would also negatively impact the efficiency and durability of the optical components, causing the laser diode (which uses the cap's metallic structure as a heat sink) to have greatly reduced power, potentially causing a failure of the spectrometer and the registering of inaccurate levels of contaminants.

Cold water is not a significant issue for the battery, as industry sources claim that lithium-ion batteries experience a ~10% performance reduction at water's atmospheric freezing point [68]. The laser diode will experience a decrease in characteristic wavelength which may prove problematic in the spectrometer, however it should be possible to implement a software solution which calibrates the rotational mechanism to 'tare' to the content's temperature during operation. An additional concern lies in the laser diode's increased quantum efficiency at lower temperatures, which can potentially cause damage to the device by sustaining greater optical output than the semiconductor can withstand. This increase in power can be handled by simply reducing the operating intensity at room temperature, which will cause superior (but not destructive) performance at lower temperatures.

10 Project Management

The project management section covers the project's cooperative management strategy for designing and programming the device.

10.1 Software Development

Software development in a team environment necessitates administrative control- specifically a shared codebase with backups, task management with issue reporting, and proper programming practices.

10.1.1 Shared Codebase

As discussed in section 9.9, a version control system is an effective way to share project files across multiple devices and users. The shared code base will utilize GitHub to sync its various contributor's source files, using Git to establish copies of the project repository on each member's computer. Git allows new files to be pushed to the server-hosted repository where they are available for use by other users. Prior to programming sessions, contributors should pull the server-hosted repository (which updates their repository with any changes).

10.1.2 Issue Reporting/Assignments

Part of the administrative process is reporting difficulties and issues with during programming, and division of labor. To this end, the version control system, GitHub, will be used to assign tasks and report issues with parts of the software. This can be accomplished using the repository's "issues" tab, which is akin to submitting a support ticket. Contributors will leave notes and status updates as issues to keep software-related issues in proximity with the repository. This system also avoids scheduling and issues and miscommunications arising from using other forms of reporting.

10.1.3 Proper Programming Practices

The project's source code will strive towards inheritance over composition and will break large tasks into 'helper' functions which perform a single task. These functions and their parameters should be established prior to writing actual code and can be safely assigned to any team member without need for excessive knowledge of the parent function. Unit testing will also be employed during the app development process.

10.2 Virtual Breadboard Prototype

Embedded team members will all have the same TI development kit and create their own individual breadboard design. The codebase will account for all device features. However, some features may be implemented for each team member’s breadboard prototype, while other more limited/expensive development features may only be on one team members breadboard. An example of this, would be the Bluetooth module development kit. Dean is going to purchase this and implement it into his breadboard prototype. So, even though Ryan and Matthew will not have it, the same codebase will need to account for all device peripherals. A micro-pin table will be utilized so that every prototype will be virtually identical and therefore implementable with the same firmware.

11 Power Distribution Design

The elements to charge the battery are going to be bought as premade modular components that will have their own small PCBs to be connected in stages. This will begin with the Universal Qi Wireless Receiver Module by Adafruit, which will be wirelessly magnetically coupled with a store-bought Qi compliant power transmitter. The wireless power receiver will be connected to an Adafruit Micro-Lipo Charger for LiPoly Batteries, which will help to manage the proper charging of the battery. This charger is then connected to the battery 2,500 mAh Lithium-Ion Polymer Battery, which will also be purchased from Adafruit. Once the battery has been charged, the power that it provides will need to be regulated so that the components are receiving the proper voltage and current. The targeted voltages and maximum current demands of each major component have been summarized in Table 28. Current limiting diodes are used for the laser diode due to its steep I-V relation, which makes small variations in voltage result in large changes in output power and input current. The diode's weakness, namely their sharp drop-off in time-space, is mitigated by pulsing the laser on only when needed.

Power Distribution Diagram

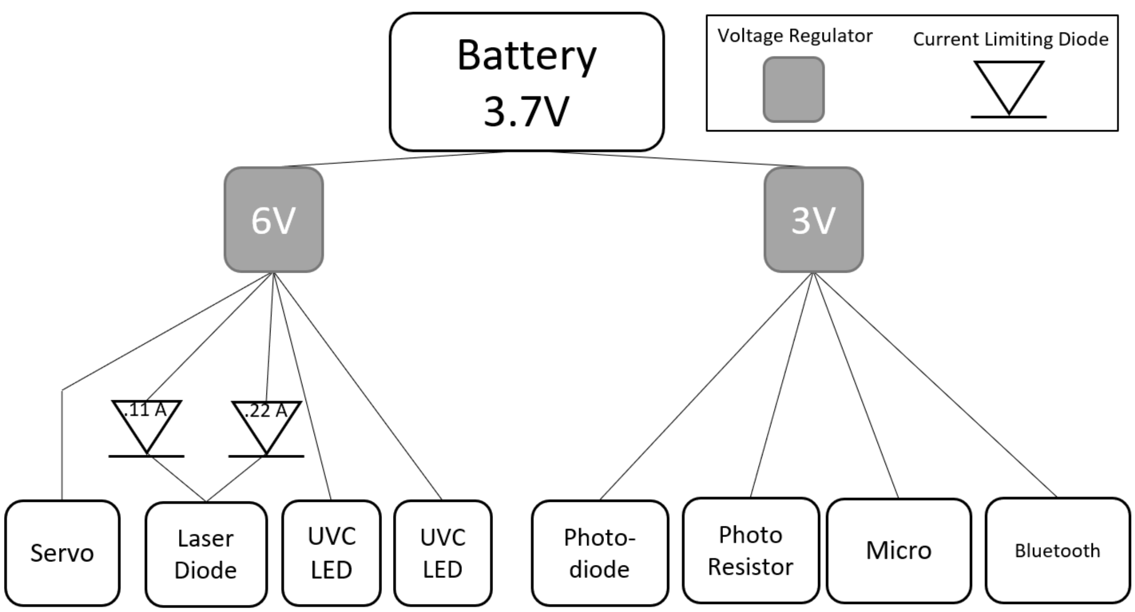


Figure 16: Power Distribution Diagram

Summary of Voltage and Current Requirements

Component (Part Number)	Desired Voltage	Maximum Voltage Ripple	Max Current
Microcontroller (MSP430FR5989)	3.0 V	As small as possible for ADC operation. Include noise bypass capacitors for V_{REF} (10 µF and 470 nF in parallel)	2.7 mA (at FREQUENCY: fMCLK = fSMCLK = 16MHz)
BLE Module (CYBLE-013025-00)	3.0 V	100 mV	28 mA (during BLE transmission)
Servo Motor (MS18)	6.0 V	Not Specified	90 mA
UVC LED 1 ()	6.0 V	Not Specified	100 mA
UVC LED 2 ()	6.0 V	Not Specified	100 mA
Spectrometer Green Laser ()	3.7 V	Not Specified	250 mA
Spectrometer Photodiode ()	<30 V	Not Specified	2 pA (leakage current rated at 10 mV)

Table 28: Summary of Voltage and Current Requirements

11.1 Voltage Regulators

Between all of the optical components, the servo motor, the microcontroller, and the Bluetooth module, there is a multitude of different voltage and current requirements for this project. It has been decided that including more than two voltage regulators in the project design would not only be wasteful in terms of project spending, but it also would require an inappropriately large amount of space on the custom PCB which will be less than three inches in diameter. The voltage of the Li-Po battery is 4.2V at full charge, nominally 3.7V, and stops discharge at 3.0V when it is depleted (12). Therefore, these were the voltage input specifications used for the TI WEBENCH Power Designer online.

The first regulator to be created will be the 3.0V DC-DC converter that will power the microcontroller, the BLE module, and provide biasing for the spectrometer photodiode. Inputting the voltage range of our battery selection with the desired output voltage of 3.0V, there were many balanced designs presented at first. Choosing an output current for this regulator was difficult, because the current demands for a regulator that is already stepping down the voltage are expected to be very small. It might seem like one would simply make the output current to be about 40 mA, since the components being connected to it are not expected to draw any more than this during peak operational periods. However, it was found that regardless of reducing the scale of expected output current, the designs presented by WEBENCH still showed the same efficiency drop-off curve as the current draw approaches zero. Since there was no efficiency improvement to be had by designing for a lower amperage, it seemed sensible to design the regulator with components rated for current comparable to the output of the other regulator; the calculation of which is discussed later in the respective section. When initially narrowing down the various design options, the main consideration for this regulator was the output voltage ripple. The datasheet for the MSP4305989 clearly recommends keeping the voltage source noise to a minimum in order to achieve the most accuracy in the measurements made by the Analog to Digital Converter (ADC) (6)7).

Investigation into multiple different components throughout this project have made it clear that sometimes obtaining specific parts for a design can lead to complications caused by the supply chain. It also is desirable to keep the custom PCB as simple as possible for multiple reasons, but in particular we wish to minimize the amount of space that will be required to implement the power conversion design. A considerable amount of space in the cap casing will already be taken up by the battery and the hardware that will be purchased for properly charging it, so in order to be mindful of the size of the cap casing that will be designed, it is important to ensure that we also keep the BOM area as small as possible. However, in comparison of the two most balanced designs that are shown below in Table 29 listed by the switching IC used for each respective design, it was decided that reducing the number of components to be included while maximizing the efficiency would make the most sense for our project design. Although there will not be nearly as much power consumed by the digital devices and one photodiode when compared to the components that will be connected to the 6.0V regulator, it is still better for heat and battery life considerations to prioritize the efficiency of the converters.

Comparison of 3.0V Buck-Boost Converter Designs

	TPS63024	TPS63020
BOM Area	102 mm^2	62 mm^2
BOM Cost	$1.42	$1.43
BOM Count	7	10
Efficiency	96.1%	93%
Frequency	2.5 MHz	2.4 MHz

Table 29: Comparison of 3.0V Buck-Boost Converter Designs

Next, we need to select a design for the 6.0V voltage regulator for the rest of the components. Again, many options were provided by the TI WEBENCH Power Designer, but the two designs that were most balanced in the desired qualities are listed below in Table 30. Selecting an appropriate output current for this design definitely had more concrete constraints that could be calculated. First of all, it is the intention of this project to code the microcontroller in such a way that will allow us to operate each of the major current consumers one at a time. This will allow us to minimize the peak current draws from the battery which will not only preserve its state of health, but also reduce the heat generated per second (assumed to be proportional to the square of the current due to I^2R losses). Thus, it seems that we will hopefully being drawing the max output current from this regulator while operating the green laser for the spectrometer. However, there is one more design factor that needed to be considered when choosing an output current and that is the recommended maximum output current of the battery. The Adafruit website for purchasing the selected battery suggests that it is best if the current draw is limited to 1A from the battery. It was therefore time to make a simple conservation of power equation that could estimate the current output of a boost regulator when the input is 1 Amp. The intention was to select a design with an efficiency well over 90%, yet these listed efficiencies are the maximum possible efficiency that can be obtained as the current draw approaches the designed upper limit. Thus, it seemed reasonable to calculate with an estimated efficiency of 85%, since the lower limit listed on many of the efficiency curves shown by WEBENCH were around 86-87%. Taking into consideration the lower limit of the battery voltage as it approaches the end of its charge, which is 3.0V, we are left with a power equation as shown below:

$P_{in} \text{ x } \eta = P_{out}$
$V_{in} \text{ x } I_{in} \text{ x } \eta = V_{out} \text{ x } I_{out}$
$I_{out} = (V_{in} \text{ x } I_{in} \text{ x } \eta)/V_{out}$
$I_{out} = (3.0V * 1A * .85)/(6.0V)$
$I_{out} = 0.425 \text{ A}$

With this output current calculation finished, the value was used for the TI WEBENCH power designer to generate the initial set of design options to choose from. Many balanced designed were initially sorted out of consideration due to the emphasis on selecting a regulator with an efficiency as high as possible. Some of the highest efficiency designs had at least 20 different components just for the single voltage regulator and these were not ideal selections in the context of this particular project.

Comparison of 6.0V Boost Converter Designs

	TPS61372	TPS61089
BOM Area	112 mm^2	91 mm^2
BOM Cost	$2.09	$2.37
BOM Count	11	14
Efficiency	91.4%	95.4%
Frequency	1.49 MHz	493.58 kHz

Table 30: Comparison of 6.0V Boost Converter Designs

11.2 LDO Linear Regulator Addendum

After much deliberation on the effects of signal noise from the switching regulator, it has recently been decided that an LDO linear voltage regulator would be more appropriately suited for supplying the 3.0V rail with power. Even though great lengths were previously gone to in the pursuit of minimizing the noise from the 3V switching regulator, the final report created by the TI WEBENCH online tool showed that the output voltage of the device would still have a peak-to-peak voltage ripple of 30 mV when the output current dips below 170 mA. Considering that we do not intend to ever draw this much current from the 3V rail and how sharply the curve of the $V_{p\text{-}p}$ increases as the amperage drawn decreases, this means that we simply cannot expect to achieve a stable reference voltage for any of the analog components in the process of converting optical intensity into a stable input voltage for the microcontroller to read.

Therefore, it has been decided that we should rely on a Low Dropout (LDO) linear voltage regulator. Looking online for in-stock components that are RoHS compliant and sorting by increasing price, there was one model of regulator that quickly stood out as meeting all of the needs for this regulator design. The MIC5365 LDO voltage regulator by Microchip Technology / Micrel is rated to have an output voltage of 3.0V, a dropout voltage of 155-310 mV at max rated current that is 150 mA, and it accepts an input voltage between 2.5-5.5V (17). The datasheet also recommends using a 1 µF capacitor at the input and output pins, respectively, that will help stabilize the voltage that will supply the 3.0V rail. The maximum line regulation of this device is 0.3% and has a typical regulation of 0.02%, which means that we can expect for our 3.0V rail to be held within 0.6 mV of whichever nominal value (±2% from 3.0V) our regulator is suppling to the digital and analog loads. When compared to the ripple that was inherent in the design of the switching regulator, this device will be holding our 3V rail 50 times more stable with the line regulation. This will provide the highest quality results of the ADC when taking readings of the water quality, which will be extremely important considering the vital nature of determining whether or not water is potable.

12 Embedded Hardware Design

This section will outline all of the processes, considerations, and iterations that go into designing the printed circuit board. Now that technical investigation and a general project overview are complete, it is time to start organizing and interfacing all of the various electrical components. This process will begin with studying data sheets, designing various circuits, selecting pin configurations for the microcontroller, creating breadboard protypes, and utilizing EAGLE's schematic software. From here, the board layout will be created based on the schematic in EAGLE. Then the board will be ordered and assembled overseas. The entirety of this design process will require careful balance and consideration between software and hardware capability. When designing the printed circuit board, the software will be created alongside the EAGLE schematic. This will allow for quick and easy adjustments between the software and the hardware/pin configurations. The hardware design will be broken up into several subsections. This helps better organize and plan out the hardware design process.

12.1 Utilizing EAGLE Software

Other PCB software could be utilized, but team members for this project have the most experience with EAGLE. This preliminary experience comes from the pre-requisite Junior Design course as well as a Co-op/internship. This software allows for quick and easy schematic creation, as well as board layout creation. In addition to this, EAGLE will keep track of all components/footprints and generate a bill of materials that can then be given to the board manufacturers. EAGLE also will generate the GERBER files that board manufacturers will use to create the printed circuit board. Thus, this software is an invaluable tool that allows the team to efficiently create a printed circuit board for the project.

EAGLE will first be utilized for fast and streamlined schematic creation.
During this time, it is imperative to select appropriate symbols, footprints, and devices. The symbol is the schematic representation, the footprint is the board layout representation, and the device merges the two together in EAGLE. Fortunately, a lot of integrated circuits and microcontrollers already have user-created EAGLE libraries that can be found online. The device simply has to be inspected to make sure dimensions and layer rules match, and then imported into a project library. Some modifications will need to be performed in EAGLE. For example, the microcontroller footprint silkscreen will be modified, as well as the pin names in the symbol. The schematic is the best referential point for how the board functions. Later on in the testing phase, it will prove to be a very useful and valuable tool. Therefore, the schematic should be easy to follow, and have logical organization and compartmentalization of all of the various device sub-systems. One way to improve organization is to take advantage of busses in EAGLE. So not every connection will have to be made, and yet the busses will make it clear which microcontroller pins connect to each sub-system. Likewise, the more resemblance there is between the schematic and the board layout, the easier it will be to analyze and test the product.

The next part of EAGLE design will be using the previously created schematic to place components and route traces between components on the circuit board. When designing the board, there are several important factors to keep in mind. Some components will have to be placed in certain locations. An example of this would be the buttons and indicator LEDs that have to be visible to the user from the exterior of the bottle cap. It also would be wise to make sure that voltage regulators are placed on top of a supply voltage plane (bottom and top of board). This would increase the heat sinking for the regulator and protect it from excess voltage that could possibly damage it. It would also be smart to create an even ground pour between the top and bottom of the board to minimize cross talk between PCB layers. In general, the board layout should be easy to follow, logical, and closely match the schematic. It is a good idea to create sub-sections throughout the board that compartmentalize device sub-systems. This makes it easier to identify device systems later on. Also, it is preferrable to fit all components on the front of the board. This eliminates the need for a back-side PCB stencil and decreases overall costs. However, some optical components may require being attached from the back side of the board, depending on the physical layout and orientation of all PCB-external components.

12.2 Microcontroller

Since this component is the central unit that will interact with just about every part of the circuit board. It appears to be a logical starting point for the hardware design process. First the pinout/configurations of the microcontroller will need to be heavily analyzed and scrutinized. Less common pin functionalities will be assigned to device subsystems, then followed by more common typical GPIO functions.

12.2.1 Device Sub-system Connections

In order to design the pin configuration for the MSP430FR5989 microcontroller it is first important to consider the different sub-systems and their input/output to and from the microcontroller. This will both provide better organization throughout the design process and ensure that all components have been considered and nothing is overlooked. It would be very troublesome to start assigning pins for the microcontroller and overlook components that have to be forcibly implemented into the design later. This section will outline these numerous systems and also determine which type of connections they will need to have with the microcontroller. Latter parts of the design process (such as microcontroller pin configuration) cannot be performed until every feature has been analyzed and broken down into specific types of connections that will interface with the microcontroller. This section is similar to outlining the ideal microcontroller which was performed under technical investigation. However, it now has more specificity since the microcontroller has been selected and more detailed information is known about the microcontroller's features and pinout. For the optical components, Table 31 covers the sensor connections required for the spectrometer/Water Quality and Table 32 establishes the outputs and inputs needed for the Sanitizer. The electrical systems and processes

(Bluetooth module, microcontroller flashing/Battery voltage, and indicators) are summarized by Table 33, Table 34, and Table 35 respectively.

Water Quality

Servomotor	PWM timer output signal
Photodiode	1 Analog I/P to the Microcontroller
Green Laser	PWM Signal to pulse laser enable signal. MOSFET to switch on circuit.
Water Quality Button	GPIO LLWU

Table 31: Water Quality Sensor Connections

Sanitization

UVC LEDs (Quantity of 2)	GPIO enable pin and MOSFET to switch on circuit.
Reed Switch	LLWU/Interrupt Pin on microcontroller (GPIO P1-4)
Sanitization Button	GPIO LLWU
UVC LED Feedback	Analog Input

Table 32: Sanitization Connections

Bluetooth Module

Pairing Mode Button	Uses either Sanitization or Water Quality button (hold for 3 seconds)
UART	Tx and Rx
Flow Control	CTS and RTS (2 GPIOs in total)
Mode Selection	CSYPP (GPIO)
Connection	GPIO
Data Ready	GPIO
Low Power Mode	LP_MODE, GPIO
Low Power Mode Status	GPIO

Table 33: Bluetooth Connections

Microcontroller Flashing and Battery Voltage

Programming Header	For microcontroller flashing
Battery Read Voltage	Analog I/P to microcontroller
Battery Read Enable	GPIO

Table 34: Microcontroller Flashing Connections

LED Indicators

R/G/Y Combined LED	3 GPIOs
Blue Sanitizing LED	GPIO

Table 35:LED Indicators Connections

12.2.2 Pin Configurations

Pin configuration is crucial when designing an embedded product. Microcontrollers today have a number of different ways that they can be configured based on different applications. It is important that pins with multiple features are designated one specific job. This then requires implementation from a software perspective to set the job/task for each pin. The MSP430FR5989 has a lot of pins, and therefore it is relatively quick and easy to create a pin configuration for this project due to an excess number of pins with various capabilities. It is always more important to map scarcer pin functions first, and then more common ones later.

12.2.2.1 Pin Configuration 1

This pin configuration focuses on creating a fully functioning prototype (while utilizing the Launchpad's features), and later applying the same configuration to the final 5989 microcontroller. In order to create a breadboard prototype for the project, it is first necessary to analyze the MSP430FR6989 (100-pin model) pinout. This is because this is the microcontroller that comes on the MSP-EXP430FR6989 Launchpad which is being utilized for the prototype breadboard design. It is very likely that the microcontroller on the final printed circuit board will follow the same layout. The main reason for this is that the code from the breadboard prototype will be easily transferable to the final product. Although the pin numbers are different on the 5989, the pins and ports still have the same functions, and the same header file can be used, so it should be pretty easy to convert between the two microcontroller pin configurations.

First, the UART Tx and Rx pins were assigned. Next the analog inputs and PWM signal pins were assigned. The last assignment involved GPIOs (both with and without the ability to interrupt the microcontroller). It is important to ensure that GPIOs are assigned to pins on the launchpad that have external jumpers to plug into the breadboard. Pin assignment was performed in this order so that scarcer project functions could be configured first, then followed by more prevalent/common project functions (such as GPIOs). Also, other microcontroller connections are made on the launchpad; however, only pin configurations related to I/O project functions were considered in this section. Table 36 below outlines the MSP-EXP430FR6989 Launchpad pin configuration for the breadboard protype project:

Development Board Micro-Pin Configuration

MSP430FR6989 Launchpad Development Micro-pins					
Pin #	Pin Functions	Project Selection	Project Function	Launchpad Feature	Launchpad Exter
65	P1.1/TA0.2/TA1CLK/COUT/A1/C1/VREF+/VeREF+	P1.1	Sanitization Button LLWU	BL Button	
64	P1.2/TA1.1/TA0CLK/COUT/A2/C2	P1.2	Water Quality Button LLWU	BR Button	
63	P1.3/TA1.2/ESITEST4/A3/C3	P1.3	BT LP_MODE		Y
2	P1.4/UCB0CLK/UCA0STE/TA1.0/S1	P1.4	Red LED		Y
3	P1.5/UCB0STE/UCA0CLK/TA0.0/S0	P1.5	Green LED		Y
4	P1.6/UCB0SIMO/UCB0SDA/TA0.1	TA0.1	PWM-Servo		Y
5	P1.7/UCB0SOMI/UCB0SCL/TA0.2	P1.7	UVC LEDs		
51	P2.0/UCA0SIMO/UCA0TXD/TB0.6/TB0CLK	UCA0TXD	UART TX		Y
50	P2.1/UCA0SOMI/UCA0RXD/TB0.5/DMAE0	UCA0RXD	UART RX		Y
49	P2.2/UCA0CLK/TB0.4/RTCCLK	P2.2	BT Data_Ready		Y
14	P2.4/TB0.3/COM4/S43	TB0.3	PWM-Green Laser		Y
16	P2.6/TB0.5/ESIC1OUT/COM6/S41	P2.6	BT CTS		Y
17	P2.7/TB0.6/ESIC2OUT/COM7/S40	P2.7	Reed Switch LLWU	(short to GND)	Y
23	P3.0/UCB1CLK/S34	P3.0	BT LP_STATUS		Y
39	P3.3/TA1.1/TB0CLK/S26	P3.3	BT RTS		Y
42	P3.6/UCA1CLK/TB0.2/S23	P3.6	BT CSYPP		Y
43	P3.7/UCA1STE/TB0.3/S22	P3.7	BT Connection		Y
62	P8.7/A4/C4	P8.7	Battery Read En		
67	P9.0/ESICH0/ESITEST0/A8/C8	A8	Battery Read Voltage		Y
68	P9.1/ESICH1/ESITEST1/A9/C9	A9	UVC Feedback Photoresistor		Y
69	P9.2/ESICH2/ESITEST2/A10/C10	A10	Photodiode Input		Y
72	P9.5/ESICI1/A13/C13	P9.5	Yellow LED		Y
73	P9.6/ESICI2/A14/C14	P9.6	Blue LED		Y

Table 36: Pin Configuration 1

This configuration would certainly make the most sense if the final product was on the Launchpad. However, the schematic for the MSP430FR5989 microcontroller would become much more convoluted. Traces would not be routed to the microcontroller in logical busses, and lots of vias would have to be used to interconnect every component correctly. It was also noticed later on that this configuration utilizes pins and ports that don’t even appear on the final microcontroller. So, this would lead to more software overhead since the pins would need to be reconfigured. Although this is a viable option, it would be wiser to create a pin configuration more focused on the final product.

12.2.2.2 Pin Configuration 2

Another option for pin configuration would be to first inspect the MSP430FR5989 pin-out and base the configuration off of that. This method would inevitably result in a more complex breadboard prototype due to connections no longer being made based on the launchpad itself. However, this ultimately results in a cleaner board routing for the final project, so it is worth it. The configuration still needs to work with the development board, so there are some important considerations that need to take place. The first major consideration that needs to take place is the fact that all connections (with exception of the launchpad buttons) need to have an external header on the MSP-EXP430FR6989 Launchpad. So, first the BoosterPack header on the launchpad was inspected and it was determined which pins/ports had an external header. The main reason for this, is that the development board needs to be able to connect to the prototype breadboard. Table 37 below outlines the header connections available on the development board:

Development Board BoosterPack External Headers

Port 1	Pins 3,4,5,6,7
Port 2	Pins 0,1,2,3,4,5,6,7
Port 3	Pins 0,1,2,3,6,7
Port 4	Pins 0,1,2,3
Port 5	None
Port 6	None
Port 7	None
Port 8	None
Port 9	Pins 0,1,2,3,4,5,6

Table 37: Development Board External Headers

After figuring out this information, it was important to only select pins that have an established BoosterPack header/jumper connection. From here, the pin configuration process began to take place. The very first pins configured were the Launchpad buttons. Since these buttons have a fixed port/pin number on the development board, this configuration had to match on the printed circuit board. Then the UART Rx and Tx pins were selected. From here the PWM signals were selected. They both have drastically different timing/duty cycle requirements, so they were configured to be on separate timers. They also needed to be on timing channels greater than zero since this is the only way to trigger a PWM event. Channel 0 on the timing modules is used for the setting the period in up-mode. Next, the analog inputs were selected. They were kept relatively close to each other during the pinout; however, the launchpad buttons got in the way from routing all of them in one bus of traces to the microcontroller. Then, all of the GPIOs with interrupts were chosen—that meant these had to be on Ports 1-4 (the buttons are already on port 1). And lastly, the remaining GPIOs were configured in a logical fashion. E.g., all of the Bluetooth GPIO connections were kept very close to each other in the pinout. Table 38 below outlines Pin Configuration 2:

Final Microcontroller Pin Configuration

MSP430FR5989IRGC Pin Configuration					
Pin #	Pin Functions	Project Selection	Project Function	Launchpad Feature	Launchpad External Header
2	P1.4/UCB0CLK/UCA0STE/TA1.0	P1.4	UVC LEDs		Y
3	P1.5/UCB0STE/UCA0CLK/TA0.0	P1.5	Battery Read En		Y
4	P1.6/UCB0SIMO/UCB0SDA/TA0.1	TA0.1	PWM-Servo		Y
5	P1.7/UCB0SOMI/UCB0SCL/TA0.2	P1.7	Servo-En		Y
6	P2.4/TB0.3	TB0.3	PWM-Green Laser (low power)		Y
7	P2.5/TB0.4	TB0.4	PWM-Green Laser (high power)		Y
9	P2.7/TB0.6/ESIC2OUT	P2.7	Reed Switch LLWU	Jumper to GND	Y
13	P5.3/UCB1STE	P5.3	Blue LED		Y
14	P3.0/UCB1CLK	P3.0	Red LED		Y
15	P3.1/UCB1SIMO/UCB1SDA	P3.1	Yelllow LED		Y
16	P3.2/UCB1SOMI/UCB1SCL	P3.2	Green LED		Y
17	DVSS1	DVSS1	Ground (SBW)		Y
18	DVCC1	DVCC1	(SBW)		Y
19	TEST/SBWTCK	SBWTCK	Spy-Bi-Wire Clock		Y
20	RST/NMI/SBWTDIO	SBWTDIO	Spy-Bi-Wire Data In/Out		Y
32	P2.1/UCA0SOMI/UCA0RXD/TB0.5/DMAE0	UCA0RXD	BT Module UART Tx		Y
33	P2.0/UCA0SIMO/UCA0TXD/TB0.6/TB0CLK	UCA0TXD	BT Module UART Rx		Y
36	P1.3/TA1.2/ESITEST4/A3/C3	A3	Battery Read Voltage		Y
37	P1.2/TA1.1/TA0CLK/COUT/A2/C2	P1.2	Water Quality Button LLWU	BR Button	
38	P1.1/TA0.2/TA1CLK/COUT/A1/C1/VREF+/VeRE	P1.1	Sanitization Button LLWU	BL Button	
40	P9.0/ESICH0/ESITEST0/A8/C8	A8	UVC Feedback Photoresistor		Y
41	P9.1/ESICH1/ESITEST1/A9/C9	A9	Photodiode Input		Y
42	P9.2/ESICH2/ESITEST2/A10/C10	P9.2	BT LP_MODE		Y
43	P9.3/ESICH3/ESITEST3/A11/C11	P9.3	BT Data_Ready		Y
44	P9.4/ESICI0/A12/C12	P9.4	BT CTS		Y
45	P9.5/ESICI1/A13/C13	P9.5	BT LP_Status		Y
46	P9.6/ESICI2/A14/C14	P9.6	BT RTS		Y
47	P9.7/ESICI3/A15/C15	P9.7	Photoresistor En	Green LED	Y
60	P4.0/UCB1SIMO/UCB1SDA/MCLK	P4.0	BT CSYPP		Y
61	P4.1/UCB1SOMI/UCB1SCL/ACLK	P4.1	BT Connection		Y

Table 38: Pin Configuration 2

12.2.3 Spy-Bi-Wire Programming

Programming the MSP430FR5989 that we put on the custom PCB will require that some header pins are installed to allow for easy access to the microcontroller. The TI Launchpad that contains the 6989 chip is able to be used for programming an external target by utilizing the eZ-FET portion of the board. The jumpers that connect the eZ-FET isolation block to the rest of the Launchpad can be removed in order to expose the header pins that will allow for the 5989 to communicate with a PC for programming. This programming is done with a simplified form of JTAG communication configuration that TI has named Spy-Bi-Wire (6). In this interface the header pins for Vcc, Vss, SBWTCK (clock signal), and SBWTDIO (data in/out) of both the Launchpad (eZ-FET side) and the 5989 target (custom PCB) will be connected temporarily via jumper wires to allow for a program to be delivered from Code Composer Studio onto the microcontroller that will be used for this project. There is still a discrepancy between the 3.3V Vcc pin that resides on the Launchpad and the 3.0V DVCC1 pin that is supplied by the battery and linear regulator locally. Therefore, we will not be connecting these power supply pins together but it is our intention to add header pins on the custom PCB for DVSS1, DVCC1, SBWTCK, and SBWTDIO to ensure that we are able to access all of the pins when it does come time to program the device. If the DVCC1 and DVSS1 pins are not needed for programming because the device is being powered locally, then this will at least give us easy access to measure the supply voltage that is being supplied to the 5989. This measurement will be an important part of testing to ensure that the voltage supply for the 3.0V rail is being regulated within the limits that are put forth by the regulator datasheet.

12.3 Design Schematics

Now that the pin configuration is complete, it is now time to work on the schematic in EAGLE. The schematic design process involves dividing the project into smaller more manageable sections and working on one section at a time. During this time, component electrical characteristics (such as resistances, voltages, and currents) should be selected, and devices should be imported into EAGLE.

All resistances should follow EIA-96 guidelines. The design should have enough flexibility as if another board revision is not intended; so, any future implementation should already be accounted for in this revision. For example, if a bypass capacitor or external pull-up/pull-down resistor may be necessary, then this part will be included on the board (but left open). It does appear that external pull resistors will not be necessary. This is because the microcontroller ensures all port pins are high impedance with Schmitt triggers prior to pin configuration (and before enabling I/O functionality). So, it seems that internal pull-up and pull-down resistors will work for the entirety of this project--without causing any false interrupt events. It is still a good idea to always clear the flag before enabling an interrupt in the code to further prevent unintended interrupt events from taking place. This design also requires further investigation into how external components will interface with the board. As a placeholder, temporary jumpers are used to represent these external connections. In the coming weeks, more reliable connectors will be selected for interfacing with each external component.

It is also important to update the packages of each device and make sure they are correct before creating the printed circuit board. For routing nets on the schematic, busses will be utilized so that the schematic connections will be simplified and easier to follow. This section will outline the full Rev. 1 schematic, as well as explain the design process that went into creating each device sub-system.

12.3.1 Voltage Regulators

This section will outline the schematics for the 3V and 6V switching regulators, as well as the final +3V LDO regulator. For more info about the design of these regulators, see Section 11. In Figure 17 and Figure 18, the 3V switching regulator and the 3V LDO regulator are shown, respectively. Whereas, in Figure 19 the +6V boost regulator is shown.

12.3.1.1 +3V Switching Regulator

This design was previously thought to be a suitable +3V schematic for the design. So, it was initially included in EAGLE, and is therefore now provided in this document as a reference. However (as stated previously), it was decided later on that a LDO would be a better choice due to analog voltage reading requirements for the final product.

+3V Switching Regulator Schematic

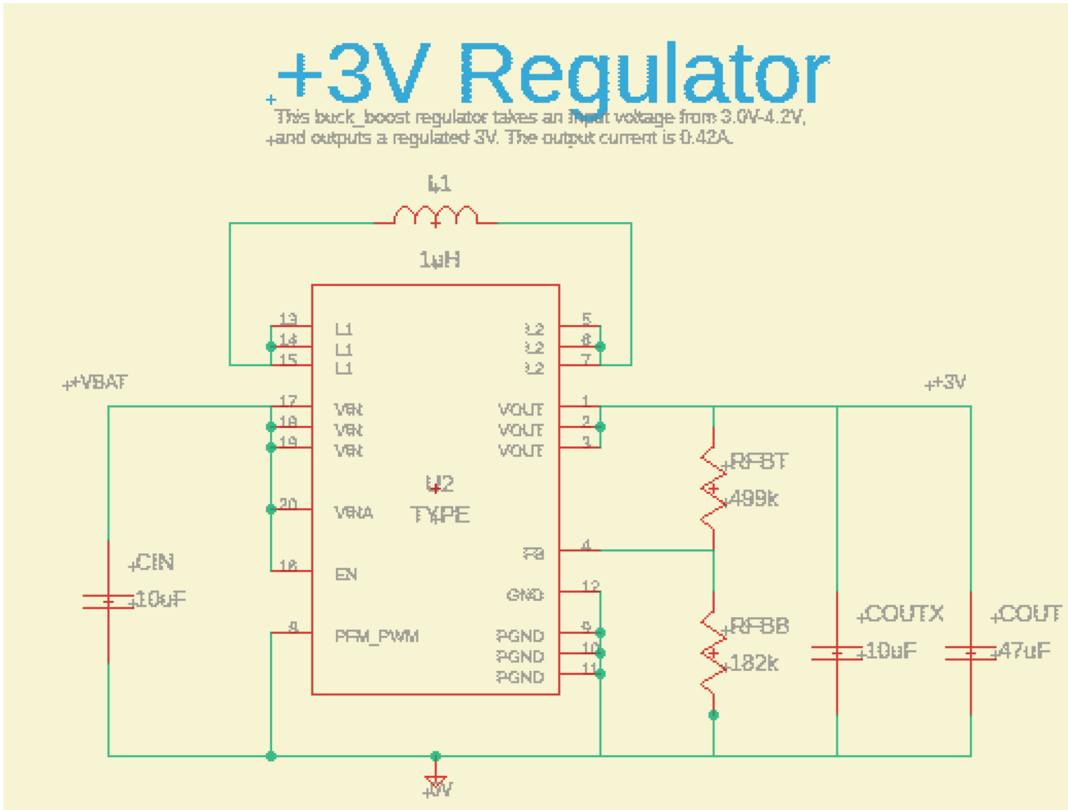


Figure 17: 3V Switching Regulator Circuit

+3V LDO Regulator Schematic

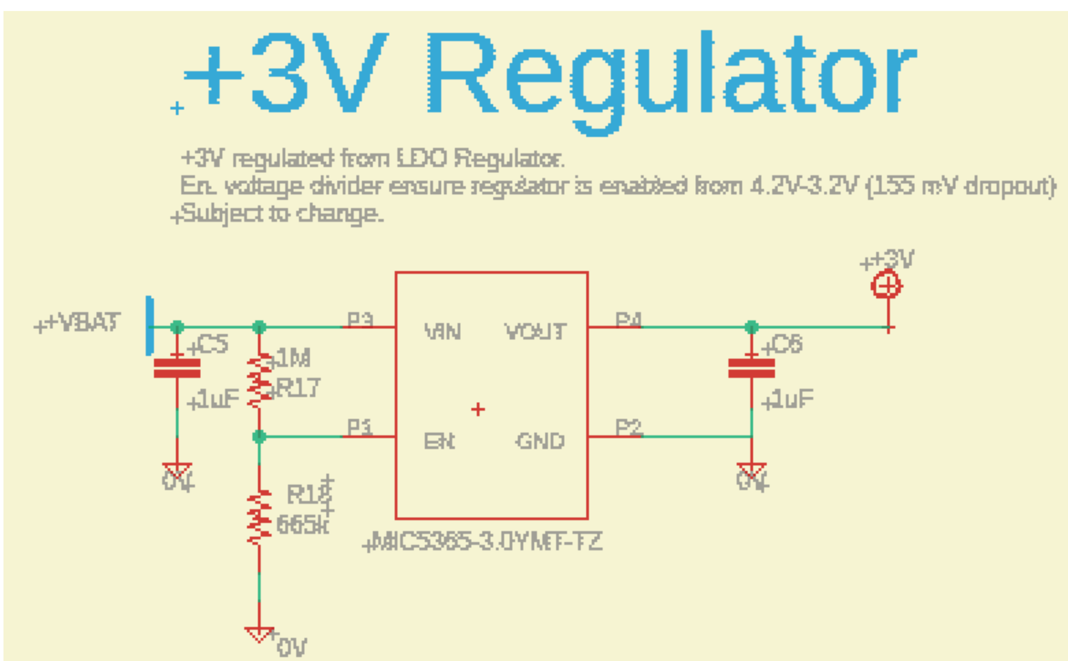


Figure 18: +3V LDO Regulator Schematic

12.3.1.2 +6V Switching Regulator

+6V Switching Regulator Schematic

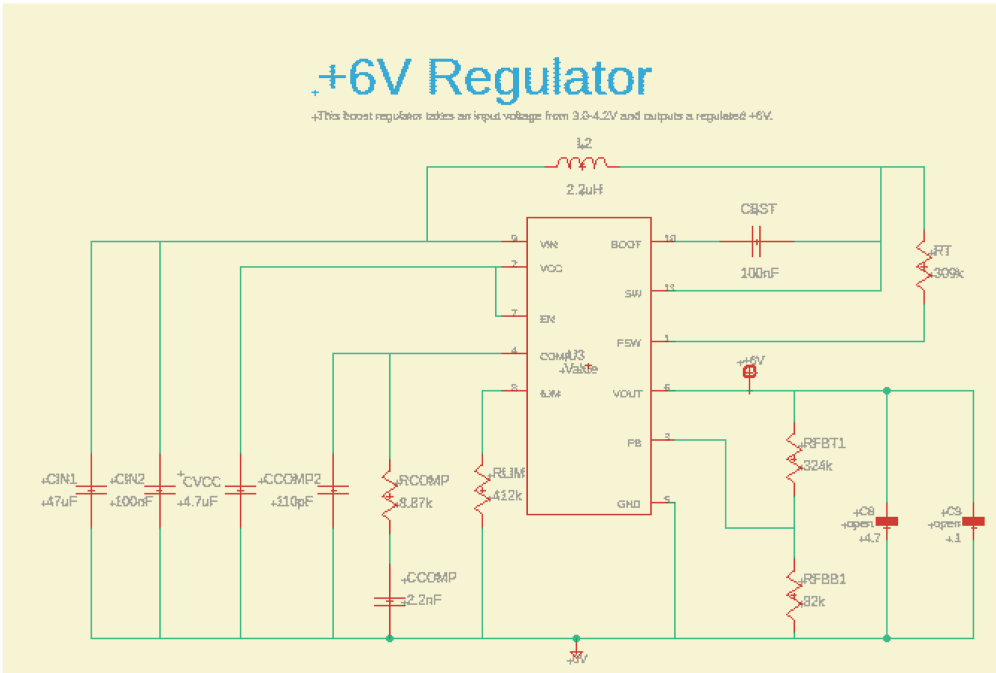


Figure 19: +6V Switching Regulator Schematic

12.3.2 Microcontroller

The microcontroller schematic follows the pin configuration provided in the previous design section. All necessary GPIO connections are routed to a bus which can then connect to any other section in the EAGLE schematic. This makes the schematic a lot easier to both route and follow. The microcontroller symbol does organize the pins by their port/pin designation according to TI. However, it was previously stated that the microcontroller should be configured so that connections make logical sense based on their designated pin number. Shown below in Figure 20 is the microcontroller schematic.

Microcontroller Schematic

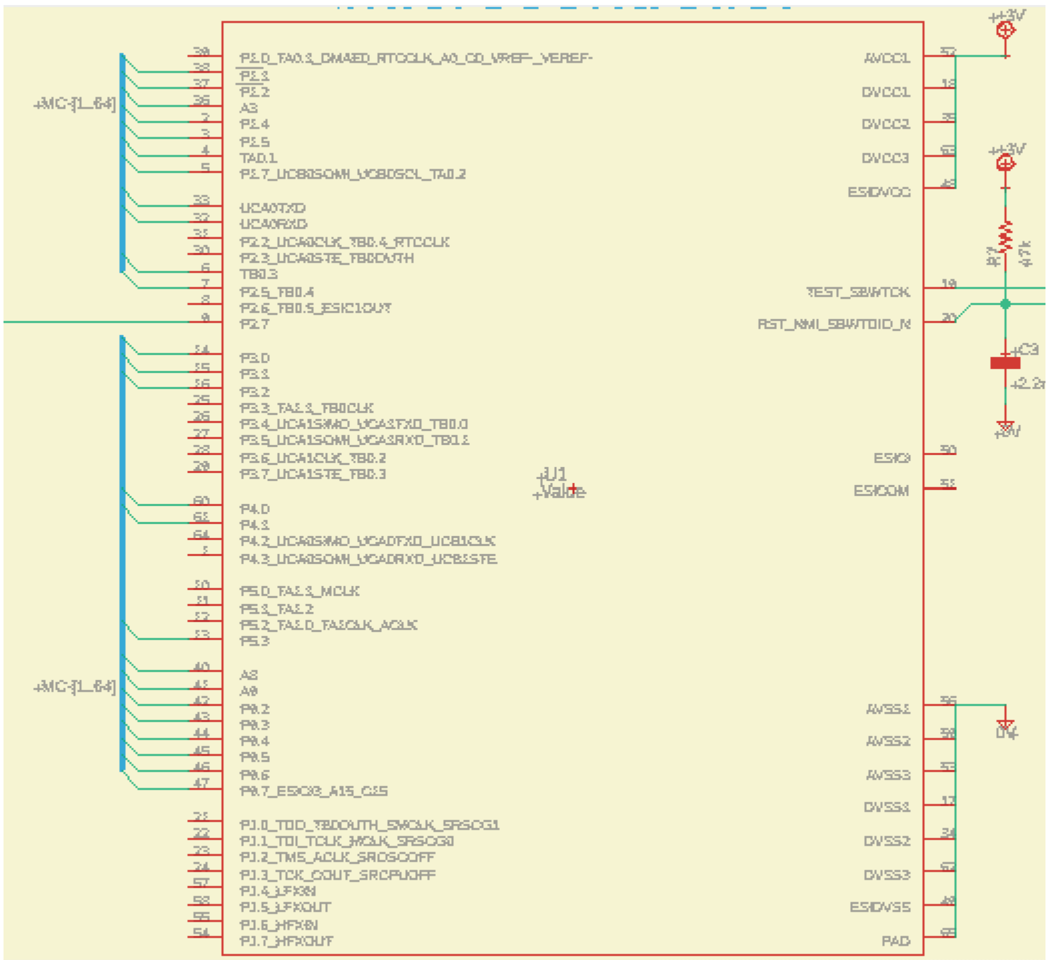


Figure 20: Microcontroller Schematic

It is also recommended by TI to include power system decoupling capacitors (1uF and 100 nF in parallel) for each VCC/VSS pair. These connections will be made in the final board layout. But they are loosely represented in the schematic. There also may need to be more noise/bypass capacitors added to the microcontroller schematic (VREF, VeREF). However, at this time it is not clear. So, further consultation is being sought to resolve this matter. In Figure 21 below, the power decoupling schematic is shown.

Power Decoupling Schematic

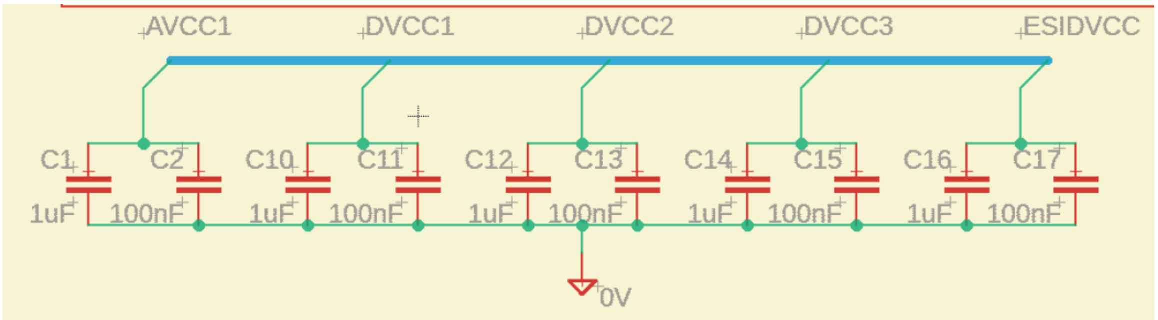


Figure 21: Power Coupling Schematic

12.3.3 *Battery Circuit*

The battery charging circuit and battery are connected externally via Adafruit sub-boards. However, a known JST PH 2-pin connector will be used to plug the battery into the board. From here, it is desired to have a power switch for the user to toggle device power on/off. Initially this was done directly through a switch; however, it was difficult to find a switch that could support enough current. So, now the switch is attached indirectly and is used to toggle a P-Channel MOSFET to turn the device on/off.

From here, it also desired to have a method to read the battery analog voltage so that crude battery life estimations can be reported to the user. This is done by enabling a N-Channel MOSFET which then asserts a voltage drop across R5 so that the P-Channel MOSFET will pass the battery voltage to a voltage divider. This voltage divider then divides down the battery voltage so that it is never larger than VCC for the microcontroller. This is the only way to read this voltage, and it will require later multiplying the result by the inverse scale in the software to obtain an estimated battery voltage reading. Shown below in Figure 22 is the battery circuit schematic.

Battery Circuit Schematic

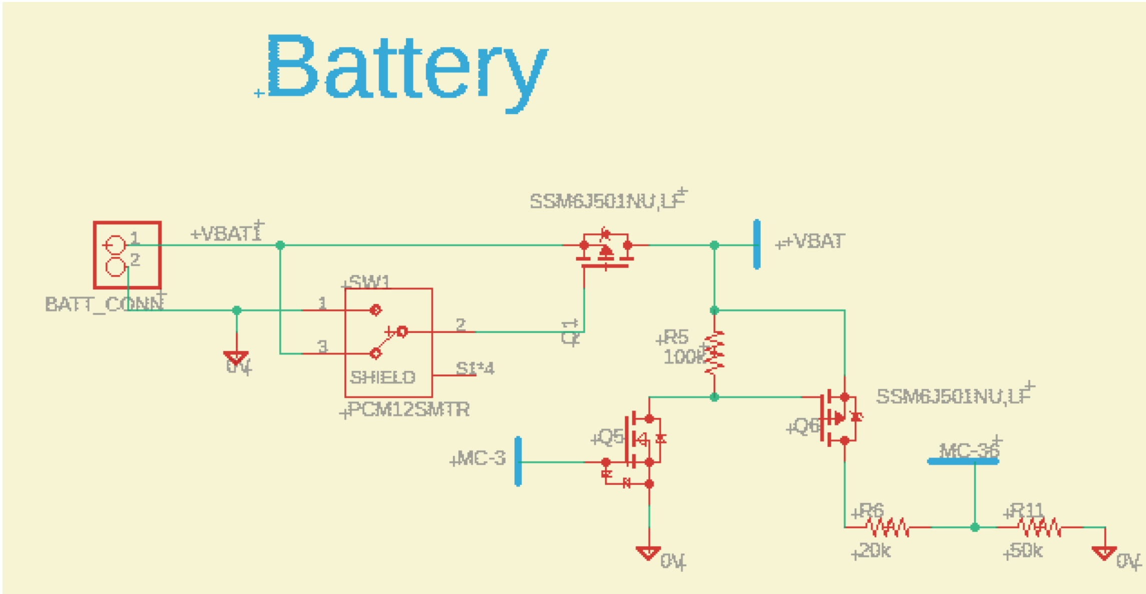


Figure 22: Battery Circuit Schematic

12.3.4 Indicator LEDs

The indicator LEDs circuit mainly involves LEDs connected in an active low configuration to the microcontroller. The red, green, and yellow LEDs are all a part of one device. Whereas the blue LED is a separate component. All anodes of this circuit are connected to the +3V rail. The cathodes are all connected to their own microcontroller pin with a current limiting resistor. In Figure 23 below, the schematic for the indicator LED circuit is shown. In Table 39 below, the calculations for LED operation and voltage/current approximation are shown. Note: Forward voltage and currents are approximations based on component datasheets and are subject to change depending on LED brightness in the final printed circuit board.

Indicator LED Electrical Characteristics

LED	Approx. Forward Voltage	Approx. Current	Approx. Resistance
Red	1.73 V	1 mA	1.27 kΩ
Yellow	1.83 V	1 mA	1.18 kΩ
Green	2.6 V	1 mA	402 Ω
Blue	2.7 V	1 mA	301 Ω

Table 39: Indicator LED Electrical Characteristics

Indicator LEDs Schematic

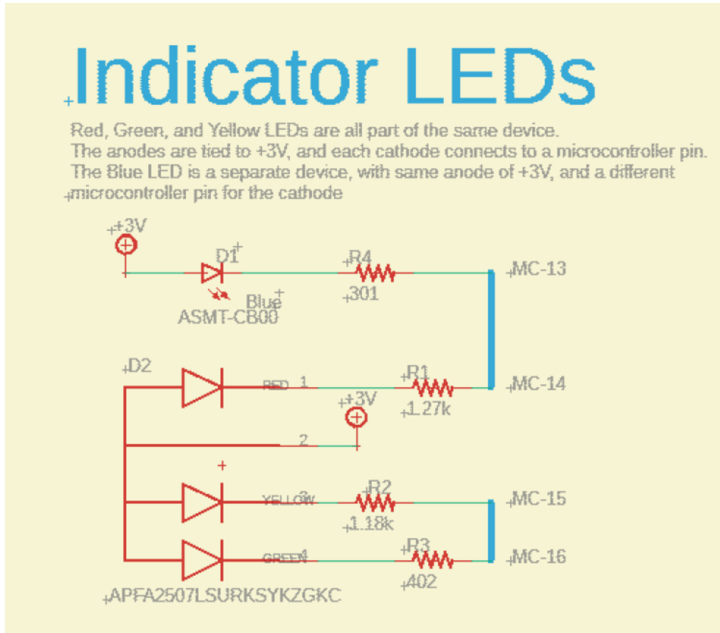


Figure 23: Indicator LEDs Schematic

12.3.5 Push Buttons

The push buttons will be used for activating various device modes such as: Water Quality, Sanitization, and Bluetooth pairing. A button press will trigger each button's individual event; however, either button can be held down for a few seconds to active device pairing. Each button connects to 0V, so it is an active low configuration. Internal pull-ups will be utilized for each button signal. Each button also requires its shield to be connected to 0V. In Figure 24, the schematic for the push buttons is shown.

Push Buttons Schematic

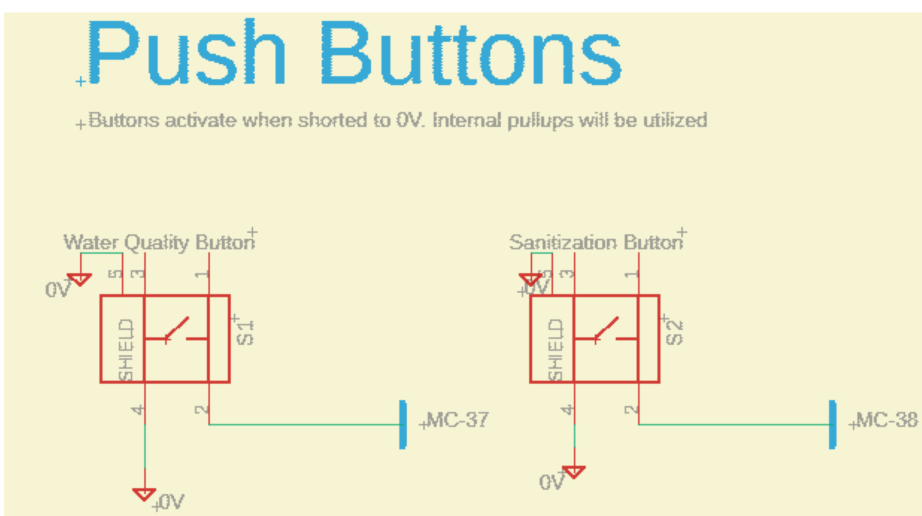


Figure 24: Push Buttons Schematic

12.3.6 Laser Diode and Analog I/Ps

The laser diode will be connected externally and requires two different modes of operation. In low power mode, it draws 110 mA through the two current limiting diodes seen on the left. The 1k resistor in parallel with these diodes is used to adjust the current of each current limiting diode. When high power mode needs to be enabled, Q2 needs to be turned on so that an additional 100 mA is drawn to operate the laser diode at a total current of 210 mA. Q3 is used to provide the PWM Enable signal that pulses the laser diode on/off. Lastly, 0 ohm resistors are included at the gate of each MOSFET in case the response time of the MOSFET switching needs to be modified at a later time. Note: Another version of this circuit may be implemented that uses a bipolar transistor to increase the regulated current of one current limiting diode. This would eliminate two CLDs in the final design.

Each optical analog input has its own necessary biasing voltage and microcontroller pin for reading said voltage. The photodiode has a negligible current draw; however, the photoresistor will be much higher and is therefore provided an enable signal to the microcontroller. The connection for each is currently thought to be a through-hole connection to the PCB. But this is subject to change. Shown below in Figure 25 is this section's schematic.

Laser Diode and Analog I/Ps Schematic

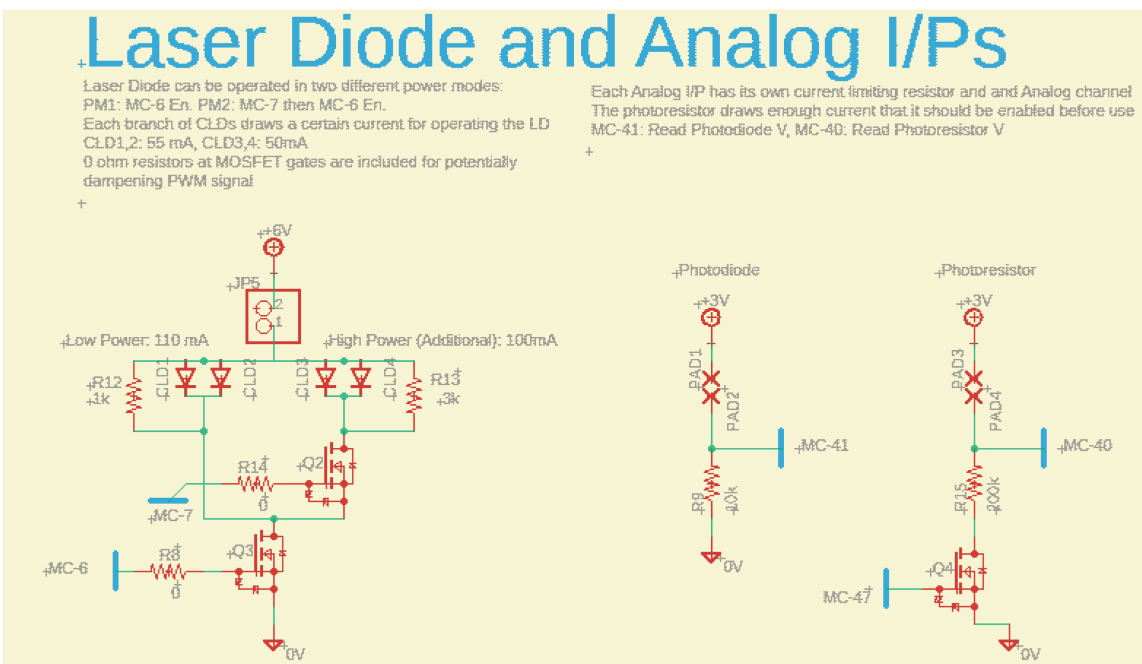


Figure 25: Laser Diode and Analog I/Ps Schematic

12.3.7 UVCs and Servomotor

The UVCs will be connected in parallel externally, and they have their own enable signal that connects to the microcontroller. The servomotor is also external and requires connections to 0V, 6V, and a PWM signal. Since the motor is an inductive load, an open resistor across the servomotor is included in case the instantaneous current from turning off the motor requires a discharge path. The servomotor also has its own enable pin on the microcontroller. Shown below in Figure 26 is the UVCs and Servomotor schematic.

UVCs and Servomotor Schematic

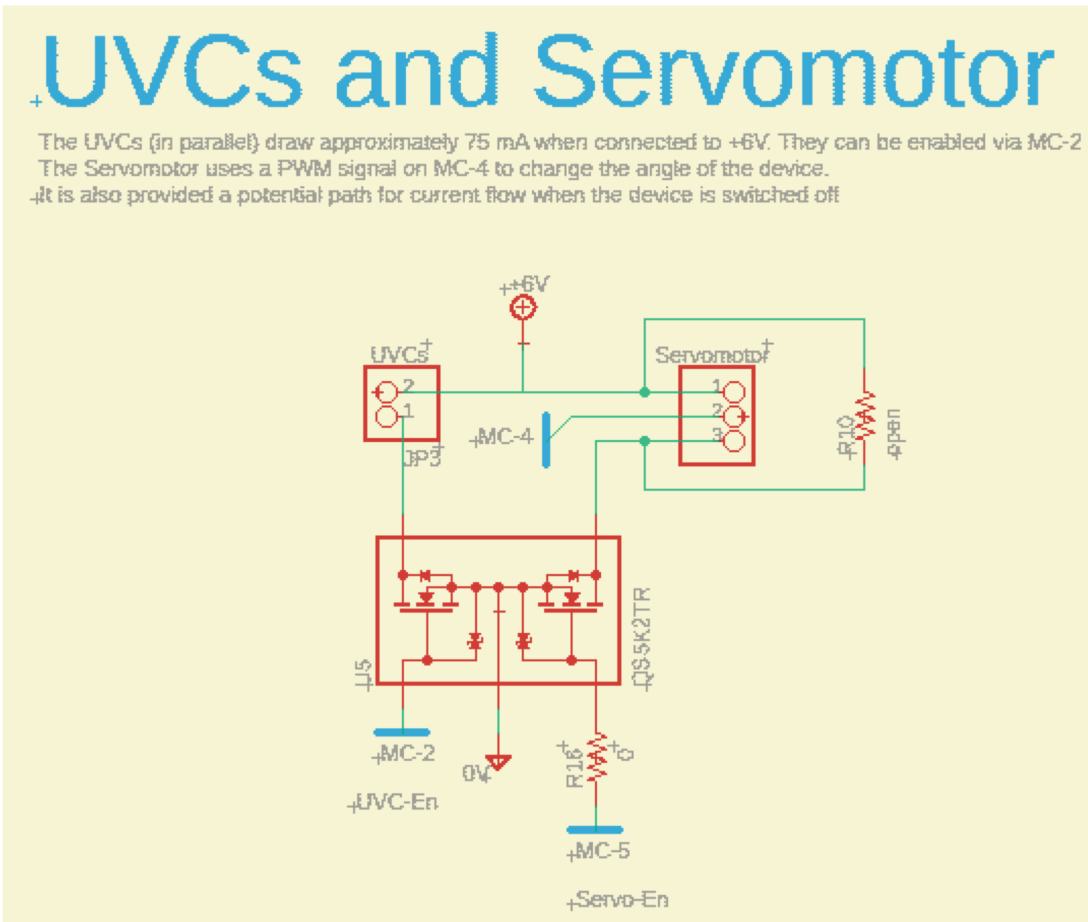


Figure 26: UVCs and Servomotor Schematic

12.3.8 Bluetooth Module

The Bluetooth module requires power connections, and GPIO connections defined in the pin configuration table, Table 8-1 of the Bluetooth Evaluation Module guide, and Section 13.2 of this document. There also is an XRES pin which is recommended by Cypress to be attached to a 0.47uF capacitor to hold the pin low for approximately 50 ms during device power-up. Shown below in Figure 27 is the Bluetooth Module Schematic.

Bluetooth Module Schematic

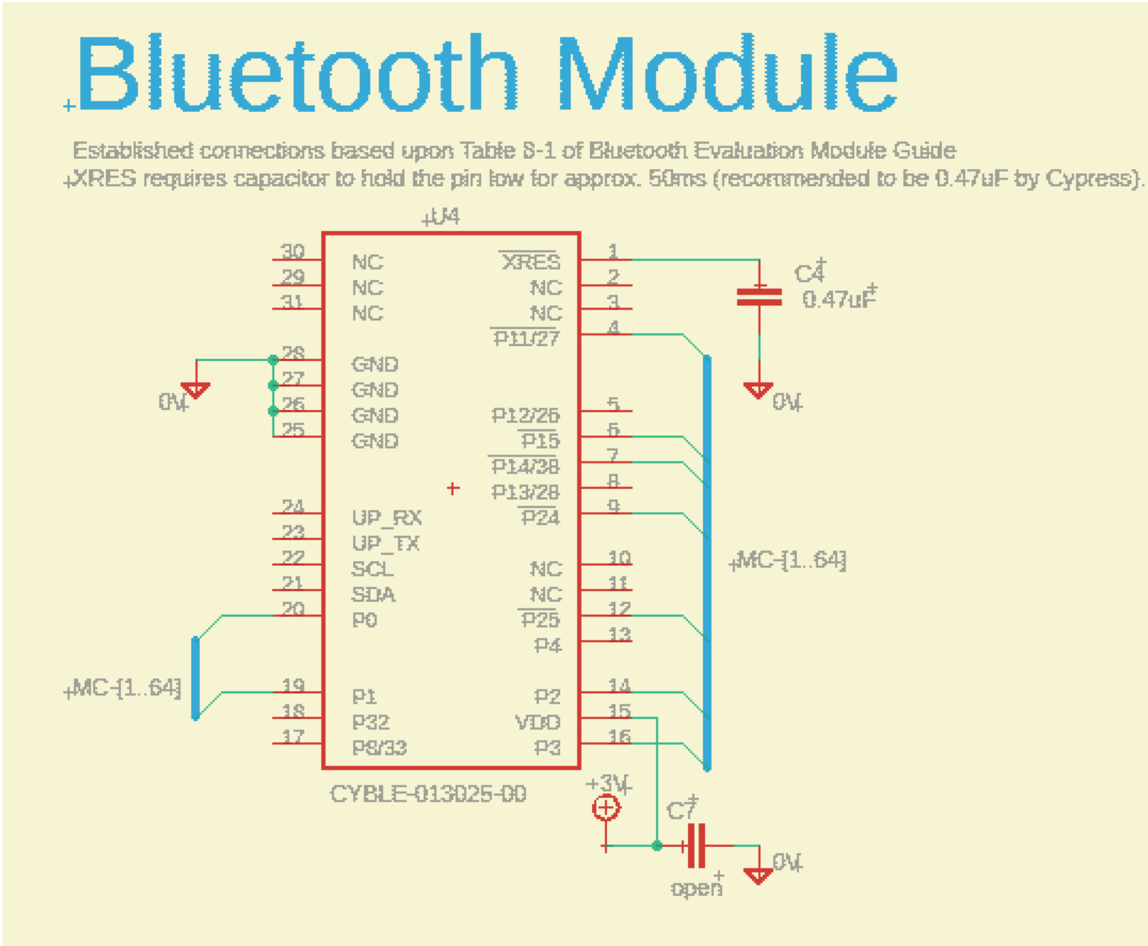


Figure 27: Bluetooth Module Schematic

12.3.9 Reed Switch

The reed switch will be connected externally and is normally open. It only triggers an event when shorted to 0V (active low). This will be pulled high internally by the microcontroller. Seen in Figure 28 below is the reed switch schematic.

Reed Switch Schematic

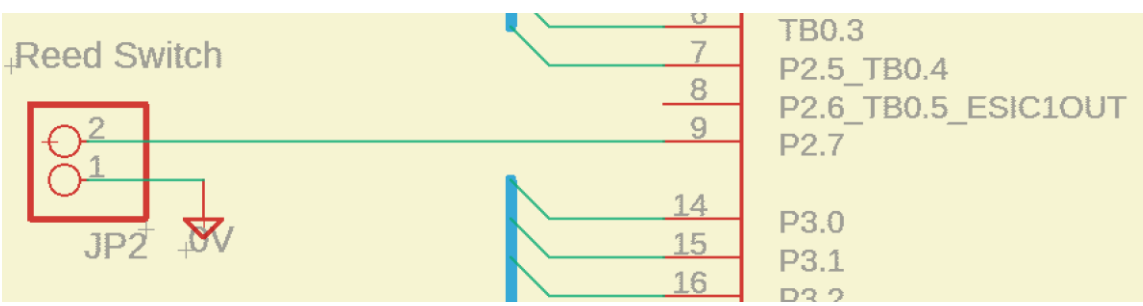


Figure 28: Reed Switch Schematic

12.3.10 Microcontroller Flashing/Programming Header

At this time, it seems that a 5-pin programming header should suffice. Some of these connections may not be necessary in the future. This serves as a placeholder for how to flash the microcontroller. This schematic is shown in Figure 29 below.

Programming Header Schematic

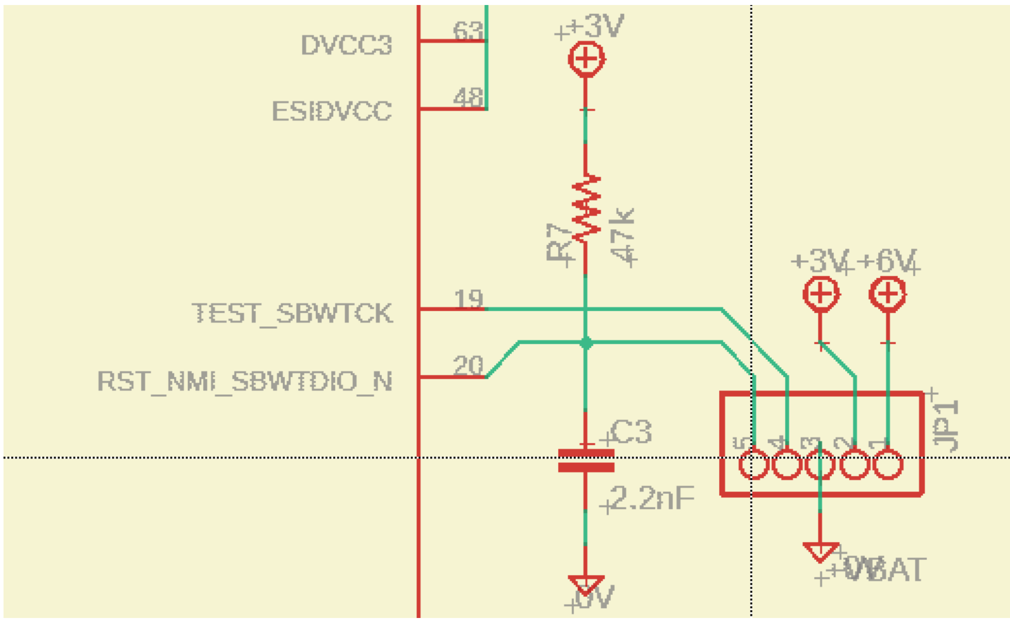


Figure 29: Programming Header Schematic

13 Prototyping and Breadboards

13.1 UVC Sanitization

A basic breadboard prototype was constructed (see Figure 30: Sanitizer Optical Setup below) and was successfully used to sterilize water from the UCF reflection pond for a demonstration. For actual implementation in the final device, the PCB used for each LED will be trimmed into a rectangular board, as the extra contacts are not needed and cause space conflicts.

Sanitizer Optical Setup

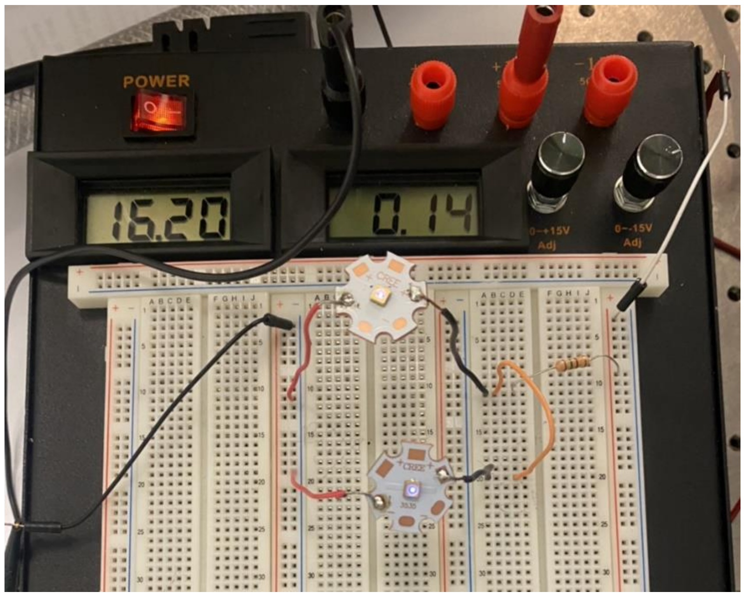


Figure 30: Sanitizer Optical Setup

13.2 Bluetooth Module

In order to test the microcontroller code that will be communicating with the BLE module, it will be very helpful to utilize the CYBLE-013025 Evaluation Board in conjunction with the MSP430 LaunchPad in order to easily test the connectivity between them. The goal is to have a development environment in which we may simulated the connections that we intend to make between the microcontroller and the BLE module, which are listed below in Table 40. These port and pin numbers are printed on the evaluation boards that have been obtained and each pin is conveniently accessible via the development board header pins. Therefore, it was necessary to connect the appropriate pins with the use of jumper cables and the resulting prototype configuration for BLE testing can be seen below in Figure 31.

CYBLE-013025 Pin Connections

Name	**Assignment**	**CYBLE Pad Number**		**MSP430 Pin Number**	**MSP430 Function**
UART_RX	P2	14	<--------------->	33	UCA0TXD
UART_TX	P0	20	<--------------->	32	UCA0RXD
UART_RTS	P1	19	<--------------->	46	P9.6
UART_CTS	P3	16	<--------------->	44	P9.4
CONNECTION	P14	7	<--------------->	61	P4.1
CYSPP	P27	4	<--------------->	60	P4.0
DATA_READY	P15	6	<--------------->	43	P9.3
LP_MODE	P24	9	<--------------->	42	P9.2
LP_STATUS	P25	12	<--------------->	45	P9.5
VDD	VDD	15	<--------------->	VDD	
GROUND	GND	25	<--------------->	GND	
GROUND	GND	26	<--------------->	GND	
GROUND	GND	27	<--------------->	GND	
GROUND	GND	28	<--------------->	GND	

Table 40: CYBLE-013025 Pin Connections

EZ-BLE Eval Board Connections

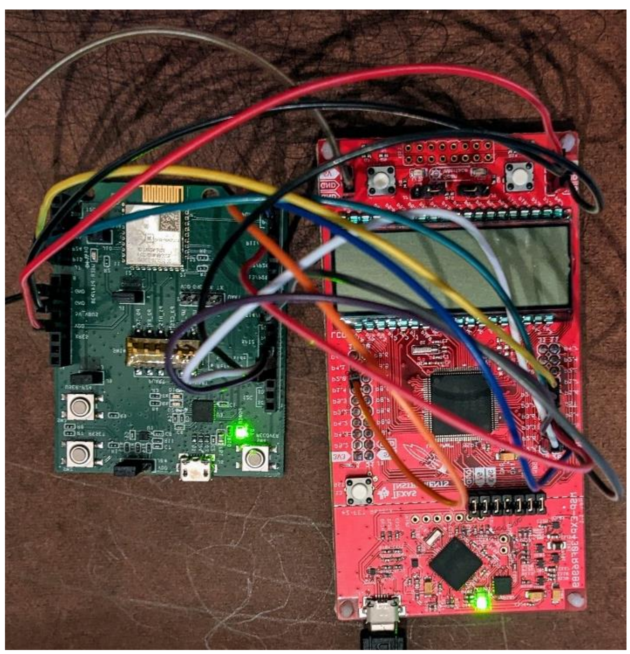


Figure 31: EZ-BLE Eval Board Connections

The final considerations for the development environment that needed to be dealt with before the coding process would begin were the DIP switches and jumpers on the pre-made boards. Jumper J8 on the Cypress board was used for selecting the V_{DD} voltage level, which was to be supplied to the board via a 3.3V header pin on the MSP430 LaunchPad. There was also jumper J10 on the Cypress board that was initially connecting P14/P38 to an LED on the evaluation board, but this needed to be adjusted because P14 will be used as the UART receiver input for the CYBLE-013025 BLE module. All 6 DIP switches on the Cypress board need to be set to the OFF position in order to put the board into "General Application Mode", which will bypass the use of the micro-USB connection so that we may access each port via the female headers and simulate the direct connection that is intended between the two integrated circuits. All of the jumper and DIP switch settings can be viewed in the evaluation board documentation (18).

The hope is that the CYBLE module has already come flashed with the aforementioned EZ-Serial WICED BLE Firmware Platform, so that we may begin to directly test the functionality of the BLE module in CYSPP mode as we intend to use in the final product. However, in order to ensure that we are aware of what settings the BLE module currently has programmed on it, the simplest option available is to connect the Cypress evaluation board to a computer terminal via US, put the module into command mode, and to send some prewritten commands over UART in order to interact with the EZ-Serial API. This will return some binary (hexadecimal) messages back to the PC UART terminal of choice and these will indicate the settings currently selected in the BLE module firmware. If the responses received do not directly line up with the examples listed in the EZ-Serial WICED BLE Firmware Platform User Guide (11), then we will need to either start figuring out how to use the Python script for converting the binary messages into text and/or download the EZ-Serial firmware from the Cypress website and flash the module that is currently embedded in the evaluation board.

13.3 Water Quality

The water quality system's electronic components are visible in a prototype setup below (see Figure 32), with the laser (whose bulk is primarily a removable housing with lens which serves as a heat sink and focusing system, and will be removed for the final device) visible on the upper right, servo and mirror on the upper middle, and photodiode on the lower left (currently in an open-circuit). The role of a laser driver is performed by a buck converter in this prototype, dropping a 5-volt output from the board to an appropriate voltage for the desired power level. The servomotor is connected directly a 5-volt output from the board in this prototype as the servo has a dedicated, internal PCB and can handle a range of input voltages. The only control pin used in this prototype is p1.6, which is the PWM signal for the servomotor. This PWM signal is altered in the prototype by pressing p1.1 or p1.2, which increase and decrease by an increment defined in the project code (which must be greater than the dead bandwidth). Not shown is the ADC connection across the photodiode's series resistor which would read the induced voltage, as this role is performed by a digital multimeter for the prototype.

Water Quality Sensor Board Connections

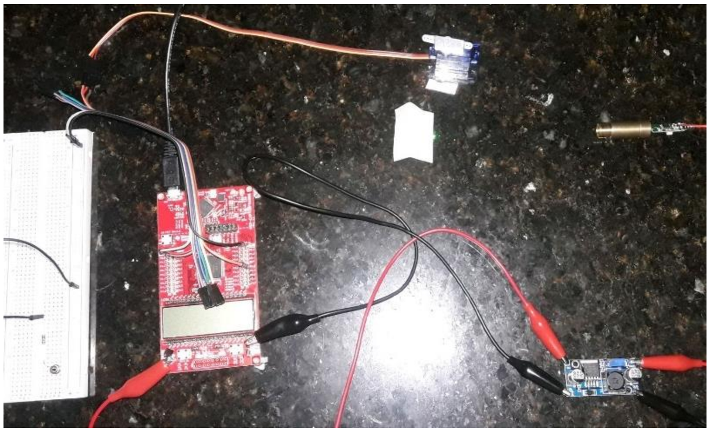


Figure 32: Water Quality Sensor Board Prototype

14 Device Operation

The following section contains some of the information which would be expected in an owner's manual of the project's envisioned final product.

14.1 Battery Management

Due to the project's emphasis on water resistance (an absolute necessity given the frequent proximity to water during the device's intended usage), the device does not have any ports for charging or data transfer and, instead, accomplishes this wirelessly. To that end, charging the device is as simple as placing the cap upside-down (with or without the bottle on top, although the latter is preferable) on a Qi-compliant charging pad and leaving it alone. The bottle design also allows for upright placement under a specialized charging tower or inside a specialized charging recess, although neither of these are required.

To preserve battery life between uses the power switch, which will be located opposite the status LEDs on the device's wall, can be turned off. This disconnects the embedded hardware and other systems from the battery, resulting in a further reduction in energy expenditure over time. To re-enable the device for sanitizing or quality assurance, the user need only switch the power back on.

14.2 Primary Functions

The device has two separate functions for processing water quality -sanitization and quality analyzation- which will be activated by two buttons on the wall of the device. These buttons will be positioned next to the status LEDs, which will provide information regarding the active and completed processes to the user. Only one of the two functions will be active at any given time to prevent damage to the battery and power requirements which cannot be fulfilled. A third primary function enables the pairing of Bluetooth connections.

14.2.1 Sanitizer

The sanitization function, when activated, will run for a period no longer than three minutes in duration. While it is operating, the water quality analyzer will not be available to initiate. This process will purify the bottle's contents of biological contaminants. It is not advised to add ice to the bottle as this can negatively affect sanitization, and contents such as hot drinks or beverages of any kind are ill-advised, the former due to reduced efficiency of the optical components at warmer temperatures and both as beverages may contain particles (such as milk, which is a colloid of fat particles in water) which may successfully shield pathogens, although the device's reflective chamber makes such an issue unlikely. When the process is completed the status light will turn off and both functions will again be available for selection.

14.2.2 Trace Element Detector

If, instead, the water quality sanitizer is activated, the device will begin to run the scan. The scan, which will complete in a minute or less, will test the water for various contaminants which pose a health risk to the user. If contaminants are detected during the scan, the red status LED will illuminate, indicating that the contents are not safe to drink. At this point the contents should be discharged and an alternate source of drinking water acquired (or the previous source can be run through a filter and tested again). The water quality analyzer is not a comprehensive test and will not detect all potentially toxic contaminants, so users are advised to never drink water with a toxic flavor, as the human tongue is a fairly good chemoreceptor. If the scan completes without triggering a warning, the status LED will turn off and both functions will be available for use.

14.2.3 Bluetooth Pairing

Finally, if the intended function to be activated is Bluetooth pairing, either button should be held down for three seconds. This will put the device into pairing mode and a corresponding android device with Bluetooth capability can pair with it to receive data regarding the water quality analyzer's results as well as status information.

15 Testing

Components, systems, and programs require testing to verify their specifications and effectivity and to gather data regarding operational parameters which impact other systems.

15.1 Components

Components are singular devices which are purchased from manufacturers and retailers without assembly by the project team. The tests to be run and their results are collated below in Table 41 and Table 42 for the spectrometer and sanitizer respectively.

Spectrometer Component Tests

Component	Tests
	Results
Servomotor	Test 1: The PWM signal for the servomotor demo code needs to be verified on an oscilloscope. It is hard to ensure that the servomotor is properly configured without observing the PWM signal's period and duty cycle on test equipment. Test 2: The angular resolution will be determined by verifying the shortest distinguishable change in PWM duration that registers as a new position (advertised to be ~7 microseconds). This will be accomplished by decreasing the increment until new positions are no longer registered by the servomotor. Test 3: A work-around for the dead band width will be tested by moving to an intermediary position in the opposite direct intended prior to moving to a new position within the dead band width from the previous location.
	Result 1: Oscilloscope waveforms clearly show that the PWM demo code starts out with a 20 ms period and 1.5ms duty cycle. Then, then the duty cycle increases in 10 microsecond increments up to 1.506 milliseconds.
Prism	Test 1: The acrylic prism's refractive index will be experimentally determined using a known wavelength and the angular deviation produced.
Laser	Test 1: The optical output power at the high and low settings will be determined using a power meter
Photodiode	Test 1: The biasing voltage will be determined by constructing a basic circuit and determining the minimum biasing voltage.
Mirrors	Test 1: The reflectivity at the characteristic wavelength will be determined by measuring the loss in power after multiple reflections.
Lens	Test 1: The focal length of the lens will be verified by focusing a collimated light source and recording the distance to the focus.

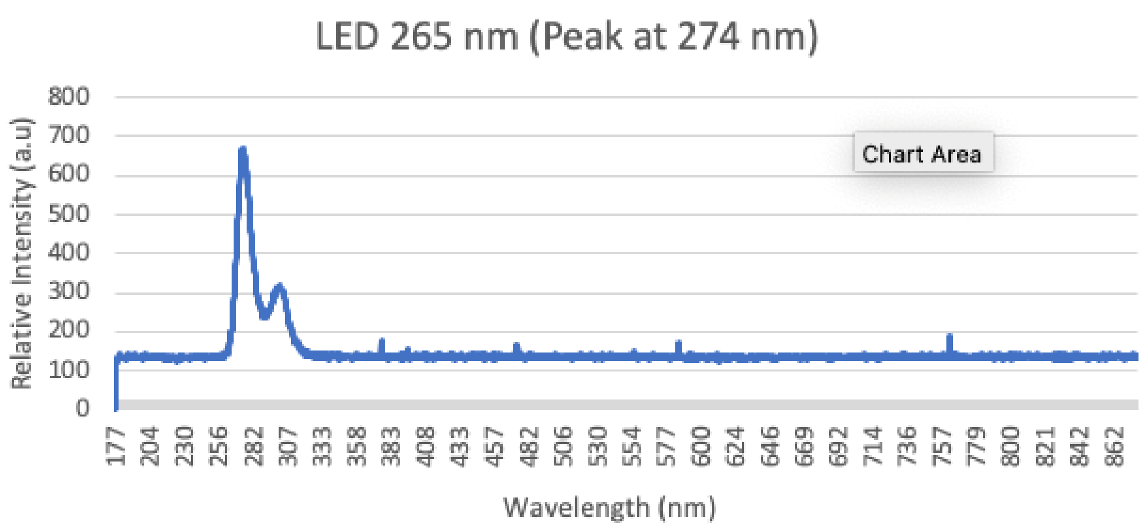
Table 41: Spectrometer Component Tests

Sanitizer Component Tests

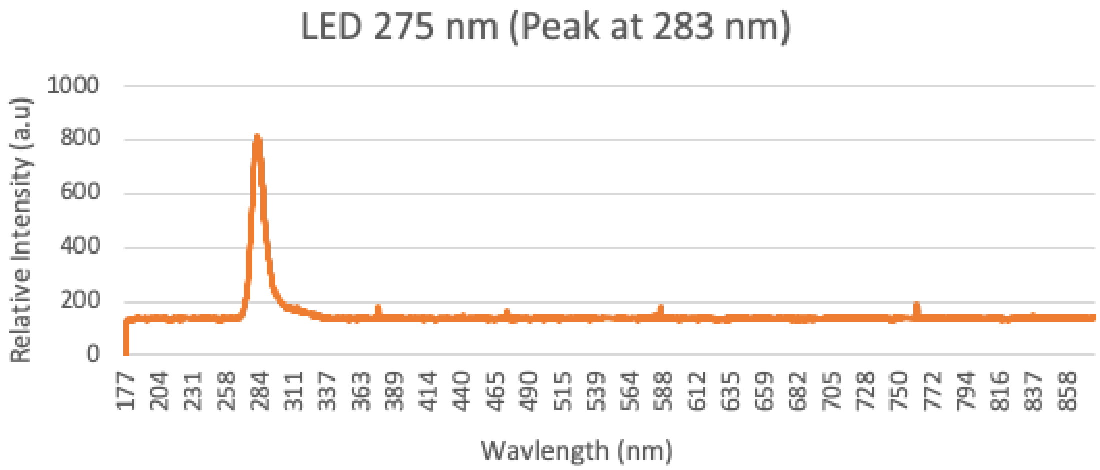
Component	Tests
	Results
LED	Test 1: Check the optical spectrum of the LEDs to see what the actual peak wavelength values are. Test 2: How much is the optical output power of the LEDs to determine the sanitization time.
	Result 1: For the 3535 LEDs the peak wavelengths for the 265 nm was 275 nm and for the 275 nm it was 285 nm as shown in Figure 33. Result 2: The optical power produced was by both the LEDs was between 0.6 mW to 0.8 mW.
Lens	Test 1: Check if the beam is focused and collimated or not. Test 2: How much optical power of the LED is lost after passing through the lens.
Beam Splitter	Test 1: Check if the beams can be combined.
Photoresistor	Test 1: The photoresistor will be tested to determine whether it can, in conjunction with a fluorescent filter or alone, reliably detect UV wavelengths.
Reed Switch	Two Reed switches of different magnetic sensitivities will be purchased, and both will be connected to a multimeter separately to determine an average distance from the magnet that activates the switch.

Table 42: Sanitizer Component Tests

UV-C LED Spectrums Measured by a Spectrometer



(a)



(b)

Figure 33: UV-C LED measured spectrum.
(a) Showing the spectrum of 265 nm LED (b) Showing the spectrum of 275 nm LED

15.2 Systems

Systems are multi-component constructs which perform the major functions of the project (such as sanitizing the bottle's contents, analyzing the bottle's contents for inorganic contaminants, or charging the battery). The tests for these systems are collected below (see Table 43).

Project System Tests

System	Tests
	Results
Spectrometer	Test 1: The spectrometer's minimal spectral resolution will be tested by using a monochromatic source and recording the number of positions which result in an accurate measurement of the incident power. Test 2: The spectrometer's ability to detect contaminants will be tested by mixing contaminants into distilled water and observing the change in results.* *To perform this test, chemicals of known concentration will have to be allocated.
Sanitizer	Test 1: The efficiency of sanitizing the water by killing the pathogens.
	Result 1: Using the Safe Home Bacteria Test from Bed Bath and Beyond, tested two samples to test the presence of harmful bacteria's with (5 minutes of illumination of UV-C LEDs at 6V and 50 mA) and without sanitization and the test proved that the LEDs sanitized the dirty water.

Microcontroller	Test 1: The current draw of the microcontroller during sleep mode and active mode will be measured with an in-series current sensor. This will further help improve battery life approximations and ensure that the microcontroller is properly entering and exiting low power mode.
Push Buttons	Test 1: Ensure that buttons properly send trigger event signal to microcontroller with simple test code to turn on LEDs. Test code will also be used to check button hold functionality
Battery	Test 1: The capacity of the battery and the ability to continuously discharge 500 mA will be observed throughout the course of a day. A regular use case for the device will be established. We will also measure the temperature of the battery as it discharges. Test 2: The voltage of the battery will be measured at different points of charge to help establish a relationship between voltage and state of charge. The direct measurements will be compared to the results from the ADC and “Battery Read” circuit.
Voltage Regulators	Test 1: Connect regulator outputs to oscilloscope to measure nominal voltage levels and peak-to-peak values while unloaded Test 2: Connect to oscilloscope and take voltage measurements during operation of each of the components Test 3: Check voltage level of battery when discharged enough to turn off the enable pin of the linear regulator to ensure desired turn-off voltage and battery end-of-life voltage
Bluetooth	Test 1: The Connection/pairing time for the app and water bottle will be tested, as well as the Bluetooth range of the device. Test 2: The maximum Bluetooth range of the device will be tested. Test 3: The reliability of data transfer will be checked by programming known values to be transmitted and reported by the app

Table 43: Project System Tests

15.3 Programs

Programs are software applications which control project systems and perform computational analysis and processing of data output from the various systems.

Program Tests

Program	Tests
	Results
Trace Element Detection	Test 1: An array representing clean water will be passed through the program to test whether false positives are an issue. Test 2: Arrays representing contaminated water will be passed through the program to test for false negatives. Test 3: Arrays representing hard (but not toxic) water will be passed through to determine whether it is possible to distinguish between contaminated water and hard water (nontoxic solutes).
Charge Status	Test 1: The battery's voltage will be measured and fed to the app where it will be compared to the discharge curve of the battery to determine whether the program effectively reports the charge status.

Table 44: Program Tests

15.4 Servomotor Optical Demo Code

For an upcoming optical system demonstration, it was requested that a simple program would be created for manipulating the servomotor. The requested demo code should default to positioning the servomotor at 90 degrees, and then have implementation for manually adjusting the servomotor. Button 1 on the launchpad should decrement the servomotor angle by approximately 1.7 degrees, and Button 2 decrements the angle by the same amount. The servomotor should only move from 90 degrees to 100 degrees; therefore, the code should prevent the servomotor from incrementing/decrementing past this range.

For creating the code, the first step was to configure P1.6 to be a timer module output. Next the period of the timer was set as 20 ms. This was generated from SMCLK running at 1 MHz. The timer was utilized in up mode and programmed to a count up to a value of 20,000-1. Thus, resulting in a 20-millisecond period for the PWM signal, which matches the servomotor requirements laid out by the datasheet. Then a reset/set pattern channel was configured to create the 1.5 ms duty cycle. Of course, this code also required configuring the buttons and activating low power mode. Then the port 1 ISR determines which button was pressed and increment/decrements the duty cycle accordingly. Conditional statements were used to prevent the servomotor from moving outside of the requested range.

The PWM signal was observed on the oscilloscope, and the waveforms exactly matched expected results. When the code boots up, the duty cycle is 1.5 ms and the period is 20 milliseconds. Each button press increments the duty cycle by 10 microseconds (for the same period), and the servomotor does not travel outside of its angular range.

16 Facilities and Equipment

This sections details some of the facilities and equipment to be used for the project, as specialized devices and tools are needed for manufacturing and testing prototypes as well as performing field research on the optical effects.

16.1 Facilities

The facilities used to manufacture and test for the project are listed in Table 45: Facilities below.

Facilities

Location	Designation	Usage
CREOL A210	PSE Senior Design Lab	Printing and prototyping the device housing as well as testing optical components.
CREOL 157	Dr. Ivan Divliansky's Lab	Prototyping and testing the sanitizer.
ENG1 456	ECE Senior Design Lab	Prototyping the embedded hardware and troubleshooting electrical issues.

Table 45: Facilities

16.2 Equipment

The equipment used to manufacture and test the project components and systems is detailed in Table 46 below. Common tools such as hacksaws, screwdrivers, and breadboards will not omitted.

Equipment

Device	Usage
High-speed Rotary Tool	Cutting the prism to meet the desired specs
Prusa 3D Printer	Creating the 3D printed cap casing to house the components of the project
Spectrometer	To measure the optical spectrum of the UV-C LEDs. Evaluating the
Power meter	To measure the optical power intensity of the UV-C LEDs.
Multimeter	To test the currents and voltages flowing through the UV-C LEDs in different schematics.
Soldering machine	To solder wires to the UV-C LEDs so it can be implemented on the breadboard.
Safe home (Enviro Test Kits)	To check the presence of bacteria before and after sanitization.

Table 46: Equipment

17 Constraints and Standards

17.1 Realistic Design Constraints

Constraints are limitations and requirements dictated by the customer, client, or environment at large. While the Smart Water Bottle Project does not currently have a customer, the customer constraints can be anticipated using the shared consumer experiences of the project leads. Environmental constraints both ecological and market-based are explored as well in this section.

17.1.1 Economic and Time Constraints

When designing this product, economic constraints are some of the most crucial constraints to consider. This applies from both developmental and consumer standpoints. As far as design, this project is self-sponsored and therefore all team members would like to try to minimize development and production costs as much as possible. A large amount of discussion and collaboration should take place before each and every purchase made for this project; this team has a very limited amount of capital to invest on this project and would therefore like to minimize wasteful purchases and/or decisions.

When sourcing any electrical or optical component, it is crucial to heavily factor in the cost for each decision. A more expensive component must provide significant value in order to justify modifying the economic constraint. Another reason to keep the design as cost-effective as possible is because of some of this project's marketing goal to be more affordable for the average consumer. When conducting a cost-analysis of the current smart water bottle market, it was clear that there is room for a more affordable yet effective and capable smart water bottle. Minimizing project/development costs helps better meet this requirement to have a more marketable and attractive product.

Another way to minimize project costs is to utilize equipment and tools already available at UCF. This project will require an extensive amount of test equipment for: signal analysis, water quality, and laser tuning. Fortunately, UCF can provide labs, testing equipment, signal analysis equipment, and more to help mitigate these costs while ensuring proper device operation. In addition to this, utilizing development boards already obtained from previous classes (that utilize the same microcontroller) can also slightly decrease development costs.

The project faces three separate time constraints: runtime, build time, and shipping time. Product runtime must be minimized as a bottle that takes a significant amount of time to sanitize and analyze will find its appeal drained before its contents are. Similarly, a slow app and connection speed will result in product disinterest. In today's society everyone is used to immediacy and being able to do, or learn, or receive something nearly instantaneously. When designing this product, it is important to keep this in mind and try to speed up device processes as much as possible. Speeding up certain processes such as sanitization would result in lower current draw from running the laser; this is certainly desirable since a lower current draw would result in longer battery life for the device.

Another important consideration is the short period of time allotted for the design of this project. Having a little over half a year to design a prototype is not a lot of time at all. Especially during the initial project phase, it is important not to waste any time and ensure that every team member is keeping a solid pace. The team meets two times weekly to discuss project management and ensure that semi-weekly goals are being accomplished. This is also a valuable time for discussion and collaboration on overarching project ideas. This team works on this project nearly every day to avoid falling behind schedule. Additionally, creating both tentative and concrete milestones has helped to motivate the team to accomplish many goals on a weekly basis. It is always easier to compartmentalize one larger milestone into smaller milestones (with mandatory deadlines) to ensure that development better remains on schedule.

The market today is struggling to recover after the global pandemic, that is one of the biggest contributors to time constraint for this project. Semiconductors, and even batteries are hard to come by. It is imperative that parts are ordered rapidly and arrive in expedient fashion so that they can be tested and implemented into the project design. Even board assemblers have significant delays right now, so it is crucial that the prototype boards are designed as quickly as possible to allow a significant amount of time for boards to arrive. Having this mindset of proactivity will ultimately help increase the development process; the sooner the boards have arrived, the longer amount of time engineers will have to program them and develop the software. But, as far as assembly, right now a turnkey assembly from PCBWay is likely going to be utilized. However, the main microcontroller that is going to be used on this project is hardly available right now. So, one way to combat this is to order the microcontrollers now and ship them to PCBWay for assembly at a later time. This will help speed up the assembly process since the board manufacturer will not have to spend a long period of time trying to source the microcontrollers for the board design.

17.1.2 Environmental, Social, and Political Constraints

The product addresses several environmental issues- the usage of disposable plastic bottles (which it seeks to reduce), the threat of waterborne diseases (which it seeks to eliminate), and the presence of toxic contaminants such as arsenic. That is not to say, however, that the device is inherently and invariably beneficial to the environment, society, or the political realm.

A potential environmental issue can be found in the product's energy storage and usage. The product must have limited battery capacity to reduce the toll from lithium's damaging production methods while still remaining functional. Using components capable of low-power modes (such as the microcontroller) is also a must to increase the lifetime of the battery as well as reduce the amount of power needed for operation. Appropriate software modifications should also be employed to reduce the power consumption of the device during operation. All of these considerations are summarized below in Table 47.

Environmental Constraints

Microcontroller	Sleep Mode Energy Consumption	< 30mA
Battery	Maximum Capacity	≤ 2,500 mAh
Bottle	Material	Metallic, with the exception of a plastic or cork seal and upper lid
Overall Product	System durability	>50 ops

Table 47: Environmental Constraints

The product also has social constraints, as certain design choices are liable to alienate large portions of society from the product. Economic constraints, which have been discussed previously, are certainly a major factor in social availability. Oher factors include color and part interchangeability, however. Device color is important as (although there is not universally agreed-upon 'favorite color') there are a few which stand out. To this end the device should not be colored yellow or orange, as these are generally considered the least-common favorite colors. Part interchangeability is key with regard to the bottle itself, as users are likely to drop and dent the bottle without damaging the device itself. To this end, the entire product (energy storage, wireless capability, charging, sanitization, analysis, and device control) must be contained entirely inside the bottle's cap.

17.1.2.1 Restriction of Hazardous Substances Directive (RoHS)

RoHS was established to reduce the amount of Electrical/Electronic equipment waste that continues to grow each year. This standard was originally established in the EU in an effort to better protect the environment and public health. Strict laws have been followed since 2003 that restrict the use of certain hazardous substances. The directive seeks to limit the use of the following ten substances: lead, cadmium, mercury, hexavalent chromium, polybrominated biphenyls (PBB) and polybrominated diphenyl ethers (PBDE), bis(2-ethylhexyl) phthalate (DEHP), butyl benzyl phthalate (BBP), dibutyl phthalate (DBP) and diisobutyl phthalate (DIBP). Currently, U.S. manufacturer must abide by this standard in order for their product to enter that particular market [69]. This standard in general, is a great one to follow for this project. It would not make sense for a project focused on reducing reusable bottles of water and water conservation to be producing electronics that are comprised of toxic materials. So, it is nice to know that PCBWay follows RoHS and therefore this project's board will be RoHS approved [70]. Additionally, due to the size and scope of the EU market, it is likely that the majority of the components sourced for this project will be RoHS compliant.

17.1.3 Ethical, Health, and Safety Constraints

The project faces a potential ethical dilemma regarding assumed safety. Safeguards must be in place that ensure that the sanitizer does, in fact, sanitize. This can be accomplished simply by wiring a visible LED in series with the sanitizer LEDs. The visible LED can/will only shine if the sanitizer LEDs are operational. The project requires some constraints regarding health and safety, primarily for the UVC LEDs and laser, although the electrical components are also potentially hazardous.

The UVC LEDs and laser must be connected in series with a switch which ceases to function when the device is opened. This can be satisfied using a Reed switch (or other mechanical switch) or by exploiting the metallic body constraint and using it as a terminal. The latter solution runs the risk of electrostatic damage and other undesirable phenomena, however. The UVC LEDs and lasers will thus be contained in an opaque metal bottle which, barring intentional destruction, will not transmit any radiation.

The battery and electrical components require a waterproof (or highly water-resistant) housing to avoid electrical shock, poisoning of the bottle contents, and explosive decomposition of the battery itself. This will be satisfied through the appropriate usage of sealants around apertures (e.g., portholes for status LEDs) and a minimalistic cap design potentially created using 3D printing.

17.1.4 Manufacturability and Sustainability Constraints

Several constraints governing the project for the sake of more efficient and cost-effective production as well as sustainability in the market are examined in Table 48 below.

Manufacturing and Sustainability Constraints

Sanitizer		
Cost	<$30	Maximum cost
Time	2-3min	Maximum time
Quality	99.9%	Effectiveness of sanitization
Design Size	4” diameter x 5” height	Maximum spacial requirements
Spectrometer		
Cost	<$150	Maximum spectrometer budget
Time	1 minute	Maximum analysis runtime
Size	<36 in^3	Maximum volume of prototype
Durability	>50 ops	Minimum number of expected operations

Table 48: Manufacturing and Sustainability Constraints

17.2 Standards

The standards section covers standards the project intends to comply with. These standards provide enhanced interface capabilities with the broader market realm as well as performance benchmarks (see Table 49 below).

Standards

Bluetooth		
RF Communication	The Bluetooth wireless standard is maintained by the Bluetooth Special Interest Group (SIG) [71] and the selection of wireless communication transmitter will comply to the standard.	
Qi Wireless Charging		
Induction Charging	The Qi Wireless charging standard is maintained by the Wireless Power Consortium [8], and the power receiver that will be purchased for this project will adhere to this standard.	
Sanitizer		
UV-C light	The UV light when exposed is harmful for the human beings.	Need to make it cost efficient so the price should not exceed this amount.
Safety	It will have a reed sensor on the bottle cap that will switch the laser off whenever the cap is opened.	Need it to be set and working for this amount of time for proper efficiency.
Wavelength	250 nm – 280nm	Need to be 99.9% efficient to kill all the microorganisms present.
Current	<0.3 A	Should be fit into a limited space as we want it compact to fit into the bottle cap.
Voltage	5.7 V	Need a way to protect the electronics from over-heating.
Spectrometer		
Sensitivity	The spectrometer must be capable of providing identifiable readings for at least the maximum allowable concentrations of arsenic, nitrates, nitrites, copper, fluoride, and lead in potable water in accordance with the EPA regulatory standards [3].	

Table 49: Standards

18 Specifications

This section contains specifications on various utilized project components. Enough documentation will be provided to allow for reproducibility of results. This part of the document also serves as a central location for the specific operating details of each and every device sub-system, namely Table 50 for the Compact Circular Spectrometer, Table 51 for the UVC sanitizer, and Table 52 for the power supply.

Analysis/Spectrometer Specifications

Component	Aspect	Specification
Servomotor	Size	23 mm long by 12.5 mm wide by 22 mm tall
	Angular Range	180 degrees
	Voltage	~5 V
	Current (unloaded)	90 mA
	Signal duration	.5-2.5 ms
Prism	Principle Angle	Equilateral
	Material	NBK-7
	Size	~5 mm^3
Laser	Intensity	100 mW
	Voltage	3.7 V (maximum)
	Current	250 mA (maximum)
	Wavelength	532 nm
	Size	12 mm in width and 35 mm height
Photodiode	Responsivity (at laser wavelength)	>= 0.3 A/W
	Biasing Voltage	-30 V (maximum reverse bias)
	Rise time	~400 ns
	Size	5.4 mm diameter and 17.8 mm height
Mirrors	Reflectance	>90% (at laser wavelength)
	Size	½” x ½”
Lens	Type	Fresnel Plano-Convex
	Focal Length	10 mm
	Material	Acrylic
	Size	1/2" diameter and 1.5 mm height

Table 50: Analysis/Spectrometer Specifications

Sanitizer Specifications

Component	Aspect	Specification
LED	Quantity	2
	Type	SMD LED
	Peak Wavelengths	265 nm and 275 nm
	Voltage	6 V
	Current	100 mA
	Intensity	6 -15 mW
	Material	AlGaN-based Flip Chip
	Mass	~83 mg each
	Size	3.6 mm-3.6 mm, 1.71mm thick
	PCB size	20 mm diameter, 1 mm thick
	PCB Material	Copper
Lens	Size	~3.6 mm in diameter
	Material	UV-Fused Silica
	Focal Length	~ 5 mm
Beam Splitter	Ratio reflected/transmitted	1-1
	Material	UV-fused silica
	Size	~1/2" x 1/2" x 1/2"

Table 51: Sanitizer Specifications

Power Supply Specifications

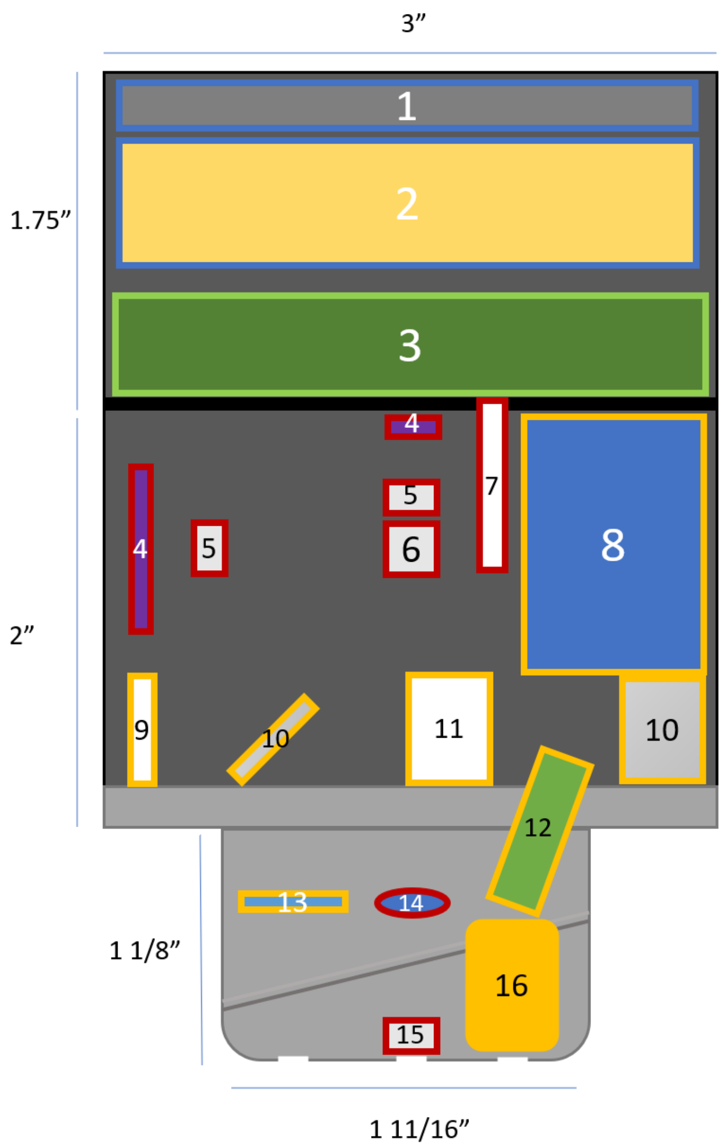
Component	Aspect	Specification
Battery	Capacity	2,500 mAh
	Type	Lithium ion
	Mass	43 g
	Nominal voltage	3.7 V
	Size	1.8" x 2.4" x 0.26"

Table 52: Power Supply Specifications

19 Structural Diagram

The project device will be very constrained by space even with the admittedly large design. As can be seen in the structural diagram below (see Figure 34), it does appear that the components chosen, which are to-scale in the diagram, are physically capable of fitting inside the proposed design. It is worth noting that a production-model device would use significantly more compact, purpose-built components, leading to a much smaller (and cheaper) device- a good example of such a product would be a CD reader/writer, which has a complex optical path, mechanical components controlling the optical path with extreme precision, and feedback all at a low cost and small size.

Structural Diagram



Dean Pickett	
Ryan Koons	
Matthew Woodruff	
Neeil Gandhi	
1	Wireless Charger
2	Battery
3	Microcontroller
4	UVC LEDs
5	UV Lenses
6	UV Beam Splitter
7	Photoresistor
8	Servomotor
9	Photodiode
10	Vis Mirror
11	Prism
12	Laser Driver
13	Vis Lens
14	UV Output Coupler
15	UV Output Lens
16	Green Laser Diode

Figure 34: Structural Diagram

20 Summary and Conclusions

As one might have expected, even though there have been many adjustments made in the technical approaches used for solving each problem that has been considered thus far, the overarching goals of this project have still not changed. Being able to sterilize the contents of a water bottle, detect the presence of inorganic contaminants, and to display this information on a smartphone app that is connected via Bluetooth Low Energy has proven to bring up more nuanced challenges as each aspect of the project has been explored. The extensive research and initial development phases for this device have been extremely fruitful over the course of Senior Design 1, and yet there is still much more to be done in the next semester in order to meet the initial project goals that have been laid out.

The investigation into power transmission that has been carried out has shown that the preliminary plans of creating a simple inductive charging setup between two coils that were wound by ourselves and operating at the 60 Hz provided right out of a wall socket would be needlessly dangerous when this would potentially overcharge a Lithium-Polymer battery in the unit. This particular chemistry of battery was an outstanding option on the market due to the higher charge storage per unit volume and the lower price with respect to the charge capacity. Therefore, extensive considerations were made to accommodate this type of energy storage. The most pronounced choice in this department was the use of premade power transmitters and receivers that are compliant with the Qi wireless power transmission standard. In terms of regulating the voltage supply rails for the device components, it was decided that using a switch-mode Boost topology would give the most efficiency for the devices requiring more power and being unaffected by the voltage ripple that is inherent in these designs. Meanwhile, despite many efforts being made to design a 3V switch-mode converter for the digital components, photodiode, and photoresistor, it was decided in the end that it is of utmost importance to optimize this converter for voltage regulation. This meant that the Buck-Boost design that was initially proposed was given up in favor of a Low Dropout Linear Voltage Regulator that would not have an output ripple from the device switching.

The research investigation involved in water sanitization system had to go through multiple steps. Starting with which method would best fit our project and if it is safer using only one method or multiple methods to deciding on what UV source to use with what wavelengths and the optical design for such, then sanitization testing and the last, feedback mechanism. There was a lot of growth and learning in this area as we researched more into it, which had to be suited for the project design constraints such as cost, size, and energy consumption. Taking all the things into account the decisions we made have been proven to give a successful first sanitization test of cleaning a bacterial water. There is still a lot of progress and tests that needs to be done. Need a proper optical layout for the LEDs and the feedback mechanism according to the project needs, exact time of sanitization required by the LEDs with the given input voltage and current for safe sanitization of water and the efficiency of sanitizing using the UV LEDs.

The water quality analyzer experienced many modifications during the technological investigation phase and will continue to evolve as prototypes are created with functional optical paths. The need for total isolation of the system from external optical sources makes complete prototypes difficult- an issue the final setup itself is well-adapted for thanks to the stainless-steel water bottle and opaque cap design. The optical effect intended to be primarily responsible for the system's efficacy has yet to be produced as well, which must be explored further.

Looking into the topic of safety for the device use, there was initially some contention about how the microcontroller would be able to detect whether or not the cap is secured onto the bottle in order to prevent accidental laser operation while the cap is off. It seemed like a good option to use a pressure activated switch that would be pressed while the cap was secured, but it was deemed too easy for a user to accidentally engage this sort of safety mechanism with their fingers while handing the cap. Consequently, we turned towards switches that could be actuated by the presence of a magnetic field because these sorts of devices are becoming increasingly more common in smart devices. The question about whether to use a Hall effect or Reed switch was resolved rather quickly, because of the component simplicity and lower power requirements. The biggest concession that had to be made with this decision was that the bottle would need to have a magnet attached near the lip, where a user will drink from. How this magnet will be attached and how far up the lip of the bottle it will need to be located is going to depend largely on the sensitivity of the Reed switch, and this will need to be tested once all of the components have been delivered.

There was also a great deal of effort put into research regarding the Bluetooth module that would be used for this project. Due to the power limitations of having a battery that can fit in a water bottle cap, it was highly desirable to select a Bluetooth Low Energy device that would only transmit data whenever it was necessary to communicate with the smartphone app. The Cypress module that was found seems to have promising out-of-the-box capabilities but until a functioning app is programmed for a smartphone, it will be forthcoming as to how easy this communication process is actually going to be.

The topic of Bluetooth leads directly into the conversation of developing a custom smartphone app that will be showing the details of the water quality analysis and potentially give an approximate reading of the battery's state of charge. Developing an app for iOS was ruled out rather quickly due to the requirements for using the officially recommended development environment for Mac. We turned to Android development and we able to identify that programming in a language similar to Java, known as Kotlin, would be most easily done by using the Android Studio IDE that was recommended on the Android website. A fair amount of preliminary investigation was accomplished in preparation of coding this app, which included choosing the Android API 21 that explicitly includes support for BLE devices.

We discovered that we have access to 3D printers in the CREOL senior design labs and this was unanimously deemed to be perfect for our needs in terms of creating a custom water bottle cap casing. It is still our intention to create a housing out of either PETG or

PLA filament that are available in the lab, will encase most of the device components, and be attached to the bottom half of the metal bottle cap. This combination of plastic and metal casing will give us the reflectiveness and tenacity for the bottom that will be exposed to reflected lasers while providing a non-ferrous housing for the rest of the device that will not interfere with either the inductive charging or the magnetic actuation of the Reed switch.

For the embedded hardware design, the goal was to design a schematic that is flexible, simple, and functional. Flexibility in the schematic/board design may eliminate the need for future board revisions which would therefore reduce project costs. The design process began by creating sub-systems to help organize and compartmentalize each part of the printed circuit board. From here, project components were extensively researched by all team members in order to come up with the best design. This process also required a large amount of microcontroller operation research. After technological investigation, the pin configuration had to be selected for the desired microcontroller. This pin configuration needed to be able to work with both the final microcontroller and the development board. The pin configuration was created so that the traces on the final product would be easy to follow. From here an initial schematic was created. This required numerous conversations/discussions about how each system would interface with the microcontroller. Each sub-system required considering power consumption requirements, how it will interface with the board, and how it will attach to the microcontroller. Many devices/symbols had to be created in EAGLE in order to represent project components in the schematic. A fair amount of MOSFET switching logic also had to be figured out in order to determine how to properly enable/disable different project requirements. In addition to this, less integral product components had to be selected and their datasheet had to be inspected. These components also had to be available in the market today. The schematic still needs to have more well-established external connections, and some MOSFETs may need to be swapped for other parts. However, the schematic is nearly done and is a fair representation of future device operation. In the coming weeks, the placeholder connections will be further established, device footprints will be edited, and the board layout will be created, inspected, and ordered from PCBWay.

There is still much more testing, programming, and redesign that is forthcoming in Senior Design 2, but it can be said with confidence that this semester has proven to be extremely fruitful in the process of creating our project. We are prepared to order our first round of PCBs for testing and will continue coding the software for the microcontroller that will be the brain of our device. There is still a considerable amount of work to be done in terms of calibrating the optical components, designing the CAD model for the cap casing, and programming the Android App that will be used to connect to our device. However, it seems that the groundwork that has been laid over the course of this semester has prepared all of us for the upcoming challenges that lie ahead in order to bring our final design to fruition.

21 References

1. Connor, Bradley A. (2019) "Travelers' Diarrhea - Chapter 2 - 2020 Yellow Book," *Centers for Disease Control and Prevention*. [Online]. Available: https://wwwnc.cdc.gov/travel/yellowbook/2020/preparing-international-travelers/travelers-diarrhea. [Accessed: 08-Jun-2021].
2. National Research Council (US) Safe Drinking Water Committee. "Drinking Water and Health" Volume 2. *Washington (DC): National Academies Press (US); 1980*. Available from: https://www.ncbi.nlm.nih.gov/books/NBK234592/ doi: 10.17226/1904. [Accessed 10-Jun-2021].
3. Environmental Protection Agency, "National Primary Drinking Water Regulations," *Environmental Protection Agency*. [Online]. Available: https://www.epa.gov/ground-water-and-drinking-water/national-primary-drinking-water-regulations. [Accessed: 10-Jun-2021].
4. Larason, T. (2020), NIST Transportable Tunable UV Laser Irradiance Facility for Water Pathogen Inactivation, Review of Scientific Instruments, [online], https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=930308. [Accessed June 10, 2021].
5. Swagatam. (2019) "3v, 4.5v, 6v, 9v, 12v, 24v, Automatic Battery Charger Circuit with Indicator," *Homemade Circuit Projects*. [Online]. Available: https://www.homemade-circuits.com/3v-45v-6v-9v-12v-24v-automatic-battery/. [Accessed: 10-Jun-2021].
6. "Inductive Versus Resonant Wireless Charging: A Truce May Be a Designer's Best Choice," DigiKey, 02-Aug-2016. [Online]. Available: https://www.DigiKey.com/en/articles/inductive-versus-resonant-wireless-charging. [Accessed: 22-Jun-2021].
7. P. K. D. A. 28, "What is Eddy Current Loss? - definition and expression," Circuit Globe, 28-Nov-2019. [Online]. Available: https://circuitglobe.com/what-is-eddy-current-loss.html. [Accessed: 22-Jun-2021].
8. E. Notes, "Qi Wireless Charging Standard," Electronics Notes. [Online]. Available: https://www.electronics-notes.com/articles/equipment-items-gadgets/wireless-battery-charging/qi-wireless-charging-standard.php. [Accessed: 23-Jun-2021].
9. L. Ada, "Li-Ion & LiPoly Batteries," Adafruit Learning System. [Online]. Available: https://learn.adafruit.com/li-ion-and-lipoly-batteries/voltages. [Accessed: 24-Jun-2021].
10. "A Designer's Guide to Lithium (Li-ion) Battery Charging," DigiKey, 01-Sep-2016. [Online]. Available: https://www.DigiKey.com/en/articles/a-designer-guide-fast-lithium-ion-battery-charging. [Accessed: 24-Jun-2021].
11. "LDO ＜What is an LDO?＞," ROHM. [Online]. Available: https://www.rohm.com/electronics-basics/dc-dc-converters/what-is-ldo. [Accessed: 01-Jul-2021].
12. I. Batarseh and A. Harb, Power Electronics Circuit Analysis and Design. Cham: Springer International Publishing, 2018.
13. R. Bearson, "What is the difference between Bluetooth versions?" *Ear Rockers*, 19-Mar-2021. [Online]. Available: https://earrockers.com/difference-between-bluetooth-versions/. [Accessed: 25-Jun-2021].
14. "Learn About Water," *Water Quality Association*. [Online]. Available: https://www.wqa.org/Learn-About-Water/Common-Contaminants. [Accessed: 10-Jun-2021].

15. CDC, "Drinking Water Treatment Methods for Backcountry and Travel Use," 20 February 2009. [Online]. Available: https://www.cdc.gov/healthywater/pdf/drinking/Backcountry_Water_Treatment.pdf.
16. "Technology/Support" *SETi | Technology | UV LED*. SETi Sensor Electronic Technology, Inc. [Online]. Available: http://www.s-et.com/en/technology/uvled/ . [Accessed: 17-Jun-2021].
17. Tissue Optics and Photonics: Biological Tissue Structures - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Electromagnetic-spectrum-and-types-of-interaction-with-matter-UV-ultraviolet-EUV_fig1_288887087. [Accessed 24 Jun, 2021].
18. "Everything You Should Know About UVC LED Light (2020 Update)." *ShineLong*. 04-Aug-2020. [Online]. Available: https://www.shinelongled.com/uvc-led-light/. [Accessed: 20-Jun-2021].
19. Bergeron, Bryan. "UV Sanitizer: How to Build One and Measure Its Efficacy." *Nuts and Volts Magazine. [Online]*. Available: https://www.nutsvolts.com/magazine/article/uv-sanitizer-how-to-build-one-and-measure-its-efficacy. [Accessed: 22-Jun-2021].
20. Center for Devices and Radiological Health. "UV Lights and Lamps: Ultraviolet-C Radiation, Disinfection, and Corona." *U.S. Food and Drug Administration*, 2021 [Online]. Available: https://www.fda.gov/medical-devices/coronavirus-covid-19-and-medical-devices/uv-lights-and-lamps-ultraviolet-c-radiation-disinfection-and-coronavirus. [Accessed: 23-Jun-2021].
21. “Technology/Support,” *SETi | Technology | UV LED*. [Online]. Available: http://www.s-et.com/en/technology/uvled/. [Accessed: 01-Jul-2021].

22. O. Lawal, J. Cosman, and J. Pagan, “UV-C LED Devices and Systems: Current and Future State,” *IUVA News*, AquiSense Technologies LLC, 2018. Available: https://uvledsource.org/wp-content/uploads/54-UVC-LED-Devices-and-Systems.pdf. [Accessed: 25-Jun-2021].

23. Craig Moe, “UV-C Light Emitting Diode,” *RADTECH REPORT*, 2014. [Online]. Available: https://www.radtech.org/magazinearchives/Publications/RadTechReport/mar-2014/UV-C%20Light%20Emitting%20Diodes.pdf. [Accessed: 28-Jun-2021].

24. “UVC 275nm LED modules, services and strengths of Universal Science,” *Universal Science*, 11-Jun-2021. [Online]. Available: https://www.universal-science.it/en/moduli-led-uvc-i-servizi-e-i-punti-di-forza-di-universal-science/. [Accessed: 15-Jul-2021].

25. Djhamer, “UV LED Exposure Box,” *Instructables circuits*, 08-Nov-2017. [Online]. Available: https://www.instructables.com/UV-LED-Exposure-Box/. [Accessed: 15-Jul-2021].

26. T. C. Larason, “National Institute of Standards and Technology transportable tunable ultraviolet laser irradiance facility for water pathogen inactivation,” *AIP Publishing*, 01-Jul-2020. [Online]. Available: https://aip.scitation.org/doi/10.1063/5.0016500. [Accessed: 15-Jul-2021].

27. J. R. Bolton, I. Mayor-Smith, and K. G. Linden, “Rethinking the Concepts of Fluence (UV Dose) and Fluence Rate: The Importance of Photon-based Units – A Systemic Review,” *Wiley Online Library*, 24-Sep-2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/php.12512#. Accessed: 15-Jul-2021].

28. S. E. Beck, H. B. Wright, R. A. Rodriguez, K. G. Linden, T. M. Hargy, and T. C. Larason, “Wavelength Dependent UV Inactivation and DNA Damage of Adenovirus as Measured by Cell Culture Infectivity and Long Range Quantitative PCR,” *ACS Publications*, 22-Nov-2013. [Online]. Available: https://pubs.acs.org/doi/pdf/10.1021/es403850b. [Accessed: 15-Jul-2021].
29. J. Ferraro and K. Nakamoto, “Introductory Raman Spectroscopy,” 2nd edition. Elsevier, 2003.
30. T. Collette and T. Williams, “The Role of Raman Spectroscopy in the Analytical Chemistry of Potable Water.” *Royal Society of Chemistry*, 2002 [Online]. Available: https://pubs.rsc.org/en/content/articlehtml/2002/em/b107274a. [Accessed: 25-Jun-2021].
31. J. Bridgeman, A. Baker, D. Brown, and J.B. Boxall, “Portable LED Fluorescence Instrumentation for the Rapid Assessment of Potable Water Quality.” *Science of the Total Environment*, 2015 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S004896971500501X. [Accessed: 25-Jun-2021].
32. P. Liang, Y. Qin, B. Hu, T. Peng, and Z. Jiang, “Nanometer-size Titanium Dioxide Microcolumn On-line Preconcentration of Trace Minerals and their Determination by Inductively Coupled Plasma Atomic Emission Spectrometry in Water.” *Analytica Chemica Acta*, 2001 [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0003267001010108. [Accessed: 25-Jun-2021].
33. R. Karka, P. Kumar, B. Bansod, and C. Krishna, “Analysis of Heavy Metal Ions in Potable Water Using Soft Computing Technique.” *Procedia Computer Science*, 2016 [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0003267001010108. [Accessed: 25-Jun-2021].
34. THORLABS, “Right-Angle Prisms,” *THORLABS*, 2021 [Online]. Available: https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=142. [Accessed: 25-Jun-2021].
35. THORLABS, “UV Ruled Reflective Diffraction Gratings,” *THORLABS*, 2021 [Online]. Available: https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=26. [Accessed: 25-Jun-2021].
36. R. Paschotta, “Image Sensors,” *RP Photonics Encyclopedia*, 2021 [Online]. Available: https://www.rp-photonics.com/image_sensors.html. [Accessed: 25-Jun-2021].
37. R. Christ, R. WernliSr., "Video," *The ROV Manual*, 2014 [Online]. Available: https://www.sciencedirect.com/topics/engineering/composite-video-signal. [Accessed: 04-Jul-2021].
38. RP Photonics, "Photodetectors," *RP Photonics*, 2021 [Online]. Available: https://www.rp-photonics.com/photodetectors.html. [Accessed: 03-Jul-2021].
39. LAFVIN, "LAFVIN 5 Sets 28BYJ-48 ULN2003 5V Stepper Motor + ULN2003 Driver Board for Arduino," LAFVIN, 2021 [Online]. Available: https://www.amazon.com/LAFVIN-28BYJ-48-ULN2003-Stepper-Arduino/dp/B076KDFSGT/ref=sr_1_11_sspa?dchild=1&keywords=micro+stepper+motor&qid=1625523294&sr=8-11-spons&psc=1&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUEyVkNSSFRXMjFSUFQmZW5jcnlwdGVkSWQ9QTAzMzgxOTUzQUlSMTZIWllETjY2JmVuY3J5cHRlZEFkSWQ9QTAyMjc5NjAyTFBDTFhTNUtKTzBHJndpZGdldE5hbWU9c3BfbXRmJmFjdGlvbj1jbGlja1JlZGlyZWN0JmRvTm90TG9nQ2xpY2s9dHJ1ZQ==. [Accessed: 05-Jul-2021].
40. NW, "5pcs 6V dDia 10mm Micro 2 Phase 4 Wire Stepper Motor 18 Degress Mini Stepping Motor," NW, 2021 [Online]. Available: https://www.amazon.com/Micro-Phase-

Stepper-Degress-Stepping/dp/B07CSLQK11/ref=sr_1_9?dchild=1&keywords=micro+stepper+motor&qid=1625523294&sr=8-9. [Accessed: 05-Jul-2021].

41. AUKUYEE, " AUKUYEE Plastic Gear Set, 75Pcs Single Double Reduction Gear Worm Gear for DIY Car Robot QY12," AUKUYEE, 2021 [Online]. Available: https://www.amazon.com/Quimat-Plastic-Single-Double-Reduction/dp/B06XCG24HZ/ref=sr_1_2?dchild=1&keywords=gear+assortment&qid=1625523703&sr=8-2. [Accessed: 05-Jul-2021].
42. Deegoo-FPV, "4Pcs SG90 9g Micro Servos for RC Robot Helicopter Airplane Controls Car Boat," Deegoo-FPV, 2021 [Online]. Available: https://www.amazon.com/Micro-Servos-Helicopter-Airplane-Controls/dp/B07MLR1498/ref=sr_1_1_sspa?dchild=1&keywords=servomotor&qid=1625529856&sr=8-1-spons&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUFJMTNSTU5PMDVVUkImZW5jcnlwdGVkSWQ9QTA1NzY0MzMyUFMzVjE1N0VIVVY1JmVuY3J5cHRlZEFkSWQ9QTAzMDM1MDMxR01XRjFNMkU3Q0VEJndpZGdldE5hbWU9c3BfYXRmJmFjdGlvbj1jbGlja1JlZGlyZWN0JmRvTm90TG9nQ2xpY2s9dHJ1ZQ&th=1. [Accessed: 05-Jul-2021].
43. OSI Optoelectronics, “Multi Element Photodiode Array,” *OSE Optoelectronics*, 2020 [Online]. Available: https://www.osioptoelectronics.com/standard-products/silicon-photodiodes/photodiode-arrays/multi-element-photodiode-array.aspx. [Accessed: 25-Jun-2021].
44. S. Writer, “How to Decide Between a Reed Switch or a Hall Switch,” Thomasnet® - Product Sourcing and Supplier Discovery Platform - Find North American Manufacturers, Suppliers, and Industrial Companies, 12-Oct-2018. [Online]. Available: https://www.thomasnet.com/insights/how-to-decide-between-a-reed-switch-or-a-hall-switch/. [Accessed: 30-Jun-2021].
45. J.M.K.C. Donev et al. (2018). Energy Education - Permeability of free space [Online]. Available: https://energyeducation.ca/encyclopedia/Permeability_of_free_space. [Accessed: July 21, 2021].
46. P. Bishop, “A tradeoff between microcontroller, DSP, FPGA and ASIC technologies,” *EETimes*, 25-Feb-2009. [Online]. Available: https://www.eetimes.com/a-tradeoff-between-microcontroller-dsp-fpga-and-asic-technologies/. [Accessed: 23-Jun-2021].
47. TronicsZone Editorial Staff, “FPGA Vs Microcontroller: When to Use What?,” *TronicsZone*, 23-Feb-2021. [Online]. Available: https://www.tronicszone.com/blog/fpga-vs-microcontroller/. [Accessed: 23-Jun-2021].
48. “RF Wireless World,” *Advantages of DSP | disadvantages of DSP*. [Online]. Available: https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-DSP.html. [Accessed: 24-Jun-2021].
49. M. Short, “Getting Started with the Raspberry Pi Zero Wireless,” Sparkfun. [Online]. Available: https://learn.sparkfun.com/tutorials/getting-started-with-the-raspberry-pi-zero-wireless/all. [Accessed: 04-Jul-2021].
50. A. Industries, “Raspberry Pi Zero W,” adafruit industries blog RSS. [Online]. Available: https://www.adafruit.com/product/3400?gclid=CjwKCAjwuIWHBhBDEiwACXQYsQuOmsvoJeAfSCp9c6R994S6nTL_el9GU4fg_l0NYBmI9rv9BqZZEBoCxwAQAvD_BwE. [Accessed: 04-Jul-2021].
51. J. Yiu and I. Johnson. (Mar. 2013). The Many Ways of Programming an ARM® Cortex®-M Microcontroller. [Online]. Available: https://community.arm.com/cfs-file/__key/telligent-evolution-components-attachments/01-1989-00-00-00-00-52-02/The-many-ways-of-programming-an-ARM-Cortex_2D00_M-microcontroller.pdf

52. A. Nguyen, “Bluetooth 1.0 vs 2.0 vs 3.0 vs 4.0 vs 5.0 - How They Compare | Symmetry Blog,” *SymmetryElectronics.com*, 18-Apr-2018. [Online]. Available: https://www.semiconductorstore.com/blog/2018/Bluetooth-1-0-vs-2-0-vs-3-0-vs-4-0-vs-5-0-How-They-Differ-Symmetry-Blog/3147/. [Accessed: 25-Jun-2021].
53. Bates, Matthew. (2013) “Build Your Own Induction Charger,” *Nuts & Volts Magazine*. [Online]. Available: https://www.nutsvolts.com/magazine/article/august2013_Bates. [Accessed: 05-Jun-2021].
54. “EZ-SERIAL™: EZ-BLE MODULE FIRMWARE PLATFORM,” Cypress.com, 21-Jun-2021. [Online]. Available: https://www.cypress.com/documentation/software-and-drivers/ez-serial-ez-ble-module-firmware-platform. [Accessed: 06-Jul-2021].
55. “Microchip Bluetooth Data - Apps on Google Play,” Google, 02-Jun-2021. [Online]. Available: https://play.google.com/store/apps/details?id=com.microchip.bluetooth.data. [Accessed: 07-Jul-2021].
56. J. Katsandres, “Bluetooth Low Energy -It Starts with Advertising,” Bluetooth® Technology Website, 15-Feb-2017. [Online]. Available: https://www.bluetooth.com/blog/bluetooth-low-energy-it-starts-with-advertising/. [Accessed: 16-Jul-2021].
57. E. Pena and M. G. Legaspi, “UART: A Hardware Communication Protocol Understanding Universal Asynchronous Receiver/Transmitter,” Analog Dialogue, Dec-2020. [Online]. Available: https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html. [Accessed: 07-Jul-2021].
58. “macOS Big Sur is compatible with these computers,” Apple Support, 17-Nov-2020. [Online]. Available: https://support.apple.com/en-us/HT211238. [Accessed: 25-Jun-2021].
59. “Meet Android Studio,” Android Developers. [Online]. Available: https://developer.android.com/studio/intro. [Accessed: 25-Jun-2021].
60. “Top Programming Languages for Android App Development,” GeeksforGeeks, 04-Mar-2021. [Online]. Available: https://www.geeksforgeeks.org/top-programming-languages-for-android-app-development/. [Accessed: 25-Jun-2021].
61. “Lesson: Object-Oriented Programming Concepts,” Oracle: The Java™ Tutorials; Learning the Java Language. [Online]. Available: https://docs.oracle.com/javase/tutorial/java/concepts/index.html. [Accessed: 19-Jul-2021].
62. Ellisys Bluetooth Video 5: Generic Attribute Profile (GATT). Ellisys, 05-Jun-2018. *YouTube*. [Online] Available: https://www.youtube.com/watch?v=eHqtiCMe4NA. [Accessed: 17-July-2021].
63. C. Y. Ong, “The Ultimate Guide to Android Bluetooth Low Energy,” Punch Through, 15-May-2020. [Online]. Available: https://punchthrough.com/android-ble-guide/. [Accessed: 17-Jul-2021].
64. “Documentation: ScanFilter,” Android Developers, 24-Feb-2021. [Online]. Available: https://developer.android.com/reference/android/bluetooth/le/ScanFilter. [Accessed: 19-Jul-2021].
65. “Asynchronous programming techniques,” Kotlin, 03-Jun-2021. [Online]. Available: https://kotlinlang.org/docs/async-programming.html. [Accessed: 20-Jul-2021].
66. L. Klint, "Azure DevOps vs GitHub: Comparing Microsoft's DevOps Tools," A Cloud Guru, 20-Jan-2021. [Online]. Available: https://acloudguru.com/blog/engineering/azure-devops-vs-github-comparing-microsofts-devops-twins. [Accessed: 21-Jul-2021].
67. P. A. Ladanyi and S. M. Morrison, “Ultraviolet Bactericidal Irradiation of Ice,” *Applied Microbiology*, vol. 16, no. 3, pp. 463–467, 1968.

68. UFO Battery, "Does Cold Weather Affect Lithium-ion Battery?," UFO BATTERY, 18-Dec-2019. [Online]. Available: https://www.ufo-battery.com/does-cold-weather-affect-lithium-ion-battery. [Accessed: 19-Jul-2021].
69. "RoHS Directive," European Commission. [Online]. Available: https://ec.europa.eu/environment/topics/waste-and-recycling/rohs-directive_en. [Accessed: 09-Jul-2021].
70. "RoHs Lead Free," Custom PCB Prototype Manufacturer. [Online]. Available: https://www.pcbway.com/pcb_prototype/RoHs_Lead_Free.html. [Accessed: 09-Jul-2021].
71. "Board of Directors," Bluetooth® Technology Website. [Online]. Available: https://www.bluetooth.com/about-us/board-of-directors/. [Accessed: 09-Jul-2021].

22 Appendix A: Datasheets

1) MS-2431-3 Press-Fit Sensor Datasheet
2) SMCO5 Magnet Datasheet
3) TVP5150AM1 Ultralow-Power NTSC/PAL/SECAM Video Decoder datasheet (Rev. E) (ti.com)
4) ATmega48A, ATmega48PA, ATmega88A, ATmega88PA, ATmega168A, ATmega1688PA, ATmega328, ATmega328P datasheet (microchip.com)
5) MSP430G2x53, MSP430G2x13 Mixed Signal Microcontroller datasheet (Rev. J) (ti.com)
6) MSP430FR698x(1), MSP430FR598x(1) Mixed-Signal Microcontrollers datasheet (Rev. D) (ti.com)
7) MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers datasheet (Rev. P) (ti.com)
8) CYBLE-013025-00 BLE Module Datasheet
9) RN4870/71 Data Sheet
10) EYSPBNZUA Data Sheet
11) EZ-Serial WICED BLE Firmware Platform User Guide
12) LIPO785060 2500mAh 3.7V Datasheet
13) MSP430FR58xx, MSP430FR59xx, and MSP430FR6xx Family User's Guide (Rev. P) (ti.com)
14) APFA2507LSURKSYKZGKC(Ver.3A) (kingbrightusa.com)
15) AV02-0587EN DS ASMT-CB00 05May2010.pdf (broadcom.com)
16) pts840.pdf (ckswitches.com)
17) MIC 5365/6 High-Performance Single 150mA LDO Datasheet
18) EZ-BLE™ MODULE ARDUINO EVALUATION BOARD

23 Appendix B: Copyrights and Permission

Permission for Figure 11: Electromagnetic Spectrum Figure taken from [17]:

Re[2]: Permission to use the Electromagnetic Spectrum Image

VT

Valery Tuchin <tuchinvv@mail.ru>
Sun 7/25/2021 10:47 PM
To: Neeil Nailesh Gandhi

Dear Neeil Gandhi,
Thank you for interest.
I will be more than happy.
Please, use as you wish.
Valery Tuchin

Valery Tuchin, Prof., corr.-member of the RAS, PhD., Dr. Sc.,

Head of Optics and Biophotonics Chair, Director of Science Medical Center, Saratov State University;

Supervisor of Lab. of Biophotonics, National Research Tomsk State University;

Supervisor of Lab. of FemtoMedicine, ITMO University;

Head of Lab. of Laser Diagnostics of Technical and Living Systems,

Institute of Precision Mechanics and Control of the RAS, Russia

Cell phone: + 7- 904-241-97-10

tuchinvv@mail.ru

> Воскресенье, 25 июля 2021, 19:26 +03:00 от Neeil Nailesh Gandhi <neeilgandhi23@knights.ucf.edu>:
>
> Greetings,
> I am Neeil Gandhi from University of Central Florida. For my Senior Design Project research paper, I wanted to ask for your permission to use your image of the Electromagnetic Spectrum of the light from the "Tissue Optics and Photonics: Biological Tissue Structures" paper you have published.
> Hope it is alright from your side to use this image for our paper. Proper citation will be given to it.
>
> Thank you,
> Neeil Gandhi
> University of Central Florida (CREOL)

Permission for Figure 12 and Figure 14 parts (a) and (b):

MH
Mitchel Hansen <mitchel.hansen@aquisense.com>
Mon 7/26/2021 6:51 PM
To: Neeil Nailesh Gandhi

Hello Neeil,

My name is Mitch Hansen with AquiSense Technologies. We are fine with you using the below referenced images from the article.
Do you mind sharing the article when it is finished? We like to see publications that reference our sources.

Best,

Mitch Hansen
Marketing Manager | AquiSense Technologies
www.aquisense.com | Tel: +1 859 869 4700

...

NG
Neeil Nailesh Gandhi
Sun 7/25/2021 10:15 PM
To: info@aquisense.com

Greetings,
I am Neeil Gandhi from University of Central Florida. For my Senior Design Project research paper, I wanted to ask for your permission to use your images - Figures 2a and 2b. From left to right: (a) Early UV-C LED
device in TO packaging. (b) Typical UV-C LED device in SMD packaging and Figure 4. Selected examples of UV-C LED water disinfection systems from your paper "UV-C LED Devices and Systems: Current and Future State" which your writers have published.
Hope it is alright from your side to use these images for our paper. Proper citation will be given to it.

Thank you,
Neeil Gandhi
University of Central Florida (CREOL)

Permission for Figure 13 and Figure 14 part (c):

AR
Alessandro Reda <alessandro.reda@universal-science.eu>
Mon 7/26/2021 1:15 PM
To: Neeil Nailesh Gandhi
Cc: Marketing <marketing@universal-science.eu>

Dear Mr. Gandhi,

nice to meet you. I work in the Marketing & Communication department of Universal Science.

First of all thanks so much for sharing with us your intentions, I really appreciated.

Secondly, it sounds good for us, since pictures wont' be used for marketing goals.
If you give to it proper citations will be definitely OK.

I can suggest you to use this link from our online catalogue:
https://catalogue.redilt.it/product-category/uv-lighting/
Here you can find all pictures of our made in Italy UVC LED solutions.

Do you think it's possible to share just with me the final result before publishing? It would be very useful.
I hope not to ask too much.

I stay at your disposal for any kind of doubts or information about us.
You can count on our cooperation if needed.

Best regards,

Alessandro Reda
Marketing & Communication Manager

UNIVERSAL SCIENCE | RED integrating lighting technologies

Tel: +390294324.517
Mob: +393346253842
Email: alessandro.reda@universal-science.eu

Website | LinkedIn

Saremo chiusi per le vacanze estive dal 9 al 20 agosto
We will be closed for Summer holidays from August 9th to 20th

...

NG
Neeil Nailesh Gandhi
Sun 7/25/2021 10:20 PM
To: info@universal-science.eu
Cc: sales@universal-science.eu; ufficiotecnico@universal-science.eu; marketing@universal-science.eu

Greetings,
I am Neeil Gandhi from University of Central Florida. For my Senior Design Project research paper, I wanted to ask for your permission to use your images of the Single UVC LED Module and SpecialRED images from "UVC 275nm LED modules, services and strengths of Universal Science" website.
Hope it is alright from your side to use this image for our paper. Proper citations will be given to it.

Thank you,
Neeil Gandhi
University of Central Florida (CREOL)

Permission for Figure 15:

Permission to use an Image from your research publication

NG Neeil Nailesh Gandhi
Sun 7/25/2021 10:42 PM
To: Karl G. Linden <karl.linden@colorado.edu>

Great, thank you so much for the permission.

Regards,
Neeil

...

Reply | Forward

KL Karl G. Linden <karl.linden@colorado.edu>
Sun 7/25/2021 10:40 PM
To: Neeil Nailesh Gandhi

Thanks Neeil
It is OK by me as long as its cited – if you are publishing it then you may need permission from the journal.

Thanks for asking

Best Regards,
Karl Linden

====================
Karl G. Linden, Ph.D., BCEEM
Professor of Environmental Engineering | Mortenson Professor in Sustainable Development
Civil, Environmental, and Architectural Engineering | University of Colorado Boulder
Phone: (303) 492-4798 | E-mail: karl.linden@colorado.edu | Web: http://www.colorado.edu/klinden | Twitter: @waterprof
PI: USAID Sustainable WASH Systems Learning Partnership (www.globalwaters.org/SWS)
Past-President: Association of Environmental Engineering and Science Professors (AEESP.org)
Google Scholar Link: scholar.google.com/citations?user=uAS7KNUAAAAJ

"Do unto those downstream as you would have those upstream do unto you"
Wendell Berry; poet, writer, environmental activist, critic, farmer

...

24 Appendix C: Project Spending

Project Spending

Component	Cost
Device Housing	
Bottle	$13.99
Total:	**$13.99**
Bluetooth Module	
Cypress Bluetooth CYBLE-013025-EVAL Board (+tax & ship)	$54.51
Total:	**$54.51**
Sanitization	
LED 275 nm (including sales tax and shipping)	$11.67
LED 265 nm (including sales tax and shipping)	$8.83
Bacteria Test Kit (2)	$7.01
Total:	**$27.51**
Spectrometer	
Prism	$19.98
Photodiode	$14.58
Laser diode	$16.15
Servomotor	$11.99
Mirrors	$8.99
Lens	$18.83
Shipping + Tax	$15.75
Total:	**$106.27**
Embedded Hardware	
Microcontrollers (MSP430FR5989IRGC x6) from TI	$38.62
Total:	**$38.62**

Table 53: Project Spending