AUTOMATIC WATER BOTTLE FILLER

By

Abigail Mohan

Jakub Migus

Priyank Jain

Final Report for ECE 445, Senior Design, Spring 2023

TA: Nikhil Arora

03 May 2023

Project No. 21

Abstract

The premise of this report is to detail the design of an automatic water bottle filling station, from the idea behind the device all the way through its implementation and testing. This sensor-based system dispenses both pre-workout concentrate and water and automatically stops dispensing water when the users bottle is full (or half full, if that's what the user selects). This cost-effective, hands-free device can be used in gyms or other small businesses and has a wide range of applications.

Contents

1 Introduction

Water bottle filling stations have saved millions of plastic bottles from being wasted and have started to become common in most public buildings. While they are effective in public spaces where many people have access to them, they are not useful in homes or small businesses. Both small-scale and public water bottle fillers require constant user attention as well as manual start and stop. Furthermore, public water bottle fillers can cost upwards of $2500 per installation [2], so they are not an attractive solution for small buildings and homes. The only alternative for at-home use is a built-in fridge water dispenser, but these features add over $200 to the cost of the refrigerator. Because of the user attention and cost constraints, automatic water dispensing was previously unattainable for many.

Our solution is to create a cost-effective water dispenser with an automatic stop and start feature that detects when any sized bottle is present and stops filling when the bottle is almost full. It offers a hands-free attention-free solution to homeowners and small businesses. Additionally, the device will have a feature for gyms to dispense pre-workout to gym-goers and can be used to mix and dispense other concentrated drinks as well. This device will save people time and money and will create an easy selling point for gyms instead of distributing pre-packaged pre-workout drinks. In our device, the user will select between types of liquid dispensed and will select either “100% fill” or “50% fill”. For concentrated drinks such as pre-workout, a predefined percentage of liquid concentrate will first be dispensed followed by water. The engineering behind this device, as well as its design process, will be detailed in this report.

1.1 System Functionality

The device will use a sensor to determine the height of the bottle/container and calculate a “full” level in the microcontroller (~80% of the bottle, to prevent over filling). Then, water or pre-workout will begin dispensing and a time-of-flight based sensor will monitor the liquid level. The device will stop filling up the bottle/container when the desired liquid level is reached. For the pre-workout, 10% of the bottle’s height will be filled with concentrate followed by water until the desired level is reached. To achieve this functionality, three high-level requirements each representing a different key component of functionality were defined. These requirements, which the final design met and even exceeded are as follows:

1. Sensors determine the height of the bottle and actively monitor the level of water in the container within +/- 2 cm.
2. Controller can automatically turn on/off the pump based on sensor values of water level and bottle height.
3. Users can select between different drink types and can select a desired fractional volume of their bottle/container (eg. ½, 1 etc.). Users can safely leave the device unattended as it operates.

These high-level requirements encompass each subsystem’s individual performance (detailed in section 1.2) as well as how these subsystems connect to achieve the functionality outlined above.

1.2 Subsystems Overview

To meet the high-level requirements described above, the device's subsystems must also function properly. Our design involves five major subsystems: *power*, *control*, *user interface*, *liquid*, and *sensing*. A description of each subsystem can be read below:

- Power: To ensure easy usage, the device can simply be plugged into the wall. This subsystem ensures that the entire device is powered using a 120 VAC (wall voltage) to 12 VDC converter, and a 12 V to 5 V buck converter. The entire system operates on these two voltage lines.
- Control: The microcontroller reads data from the sensors and user interface, determines what action (if any) to take, and sends commands to the pumps and sensors.
- User Interface: How the user interacts with the system. Includes a master power switch, a 50%/100% fill toggle switch, a preworkout/water toggle switch, a manual water dispense button, and an automatic start button.
- Liquid: Includes the water and preworkout concentrate containers, dc submersible liquid pumps, tubing, and overvoltage protection circuitry.
- Sensing: This subsystem is what makes our device unique. It utilizes two separate sensors – a time of flight sensor mounted at the top of the housing to monitor the current water level and a custom LED/phototransistor array to optically measure the height of the bottle.

The subsystems described above must function on their own but also interconnect to each other in order for the comprehensive design to work. These connections can be visualized on the block diagram shown below:

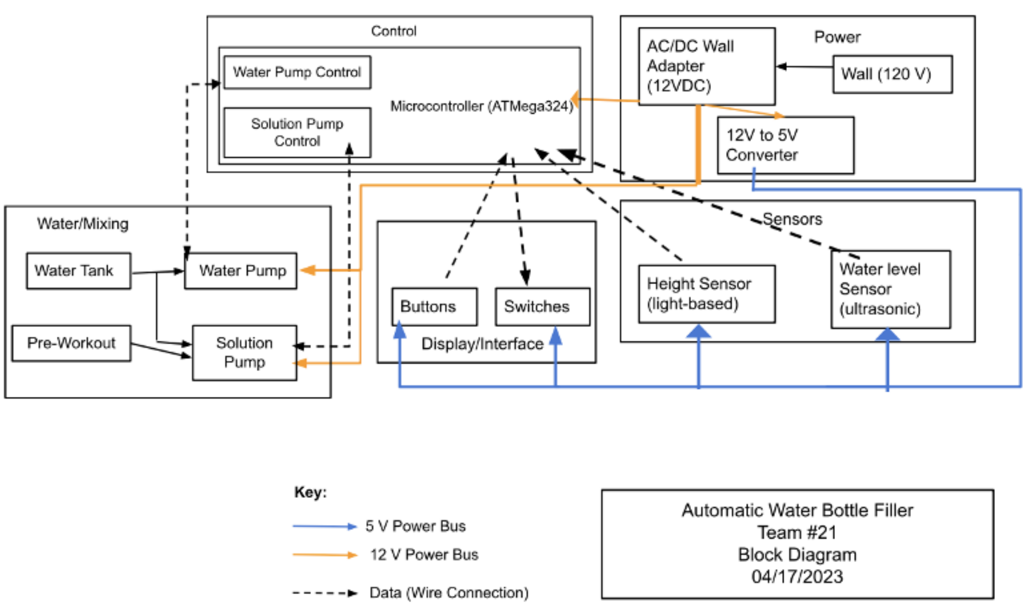

Figure 1: High-Level Block Diagram

2 Design

In this section we will be going over the design process. We will split this up by going over each subsystem individually while discussing how they each connect to one another. Figure 2 shows the full schematic of our design.

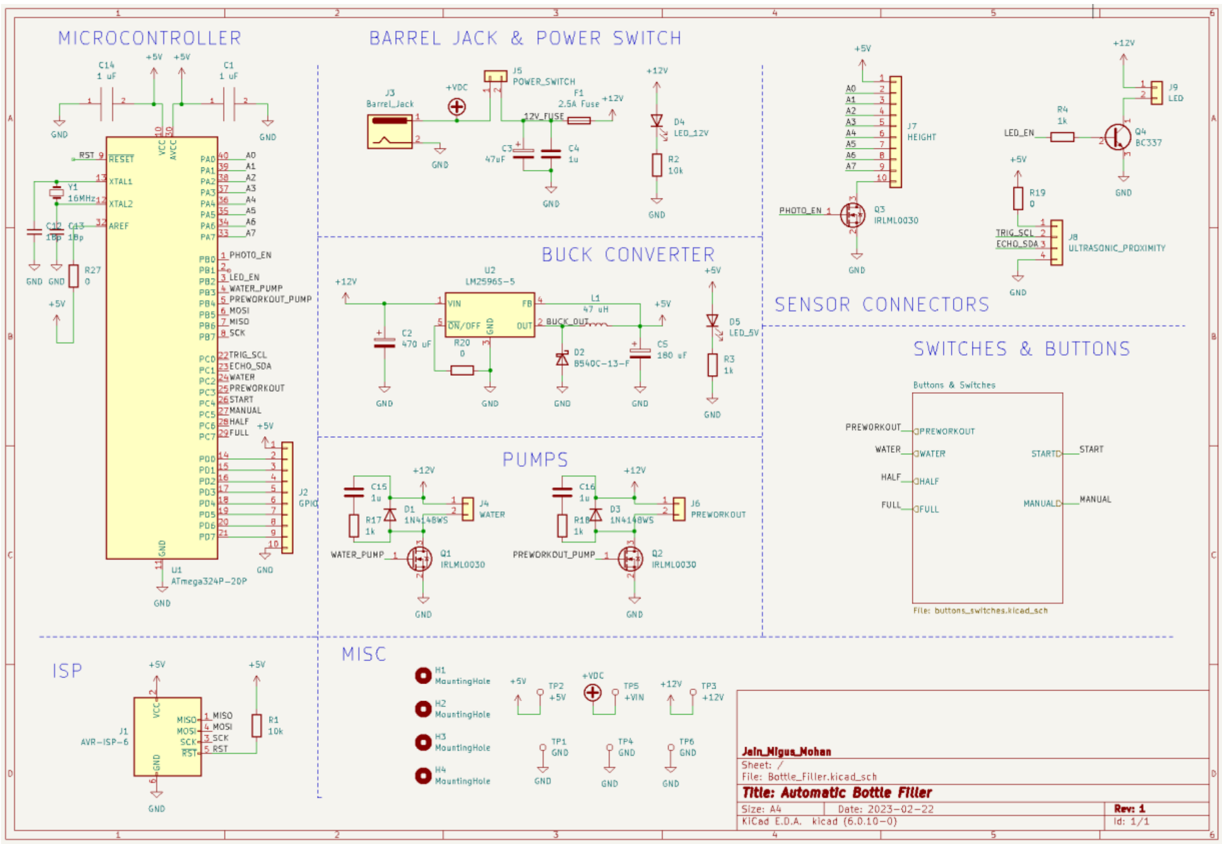

Figure 2: High-Level Block Diagram

2.1 Power Subsystem

The power subsystem handles power delivery to the entire device efficiently. For user convenience and simplicity, the device is wall powered. To stay within voltage ratings for the different components receiving power, we utilize two separate power rails: 12V rail (to the pumps and LED's), 5V rail (to the various sensors, buttons, switches, and microcontroller). To accomplish this, we utilize a basic 12V AC-DC wall adapter along with the LM2596S-5 buck converter. There is also a main power switch which will also act as an emergency shutoff. The power input is rated for 12V and a maximum of 2.5A with a fuse. The power subsystem's schematic is shown below:

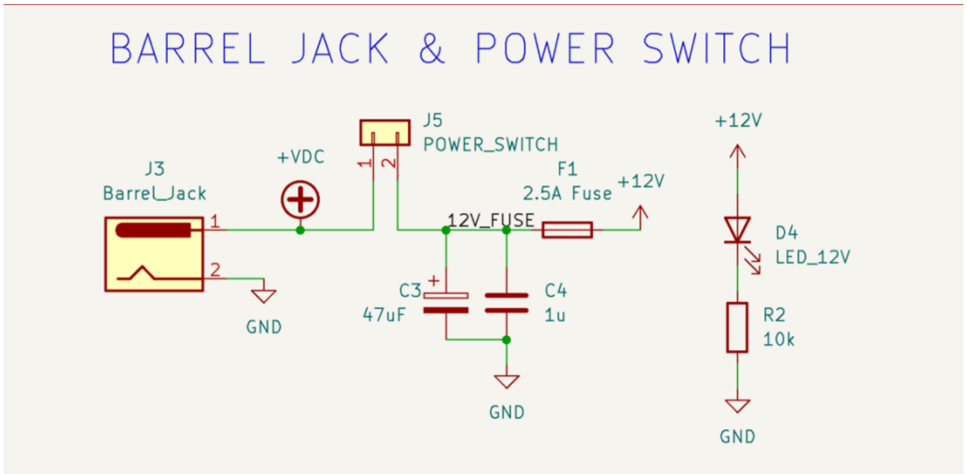

Figure 2: Power Subsystem Schematic

2.2 Control Subsystem

The control subsystem accepts user and sensor inputs and drives the pumps. The microcontroller receives settings and start/stop commands from the microcontroller. The biggest design decision that was made in the control subsystem was the selection of a controller. Because our sensing and user interface components required a high number of I/O pins, we chose the ATMega324P. This controller is relatively inexpensive, has 28 I/O pins, and can be programmed via USB or ISP. Additionally, it includes 8 analog pins which are used by the height sensor. The microcontroller drives the height sensor and calculates the height of the container (water bottle). Then, the device enables the correct pump to dispense liquid while reading from the water level sensor. When the microcontroller determines the correct amount of liquid has been dispensed, it shuts off the pump and waits for the next user input. The microcontroller enables the pumps through N-channel MOSFET's due to the microcontroller's I/O pin current and voltage limit. The analog ADC pins will be used for the phototransistor height sensor, and the GPIO breakout connector will be used for testing of new features in the future. All sensors and buttons/switches are routed to I/O pins on the ATmega324P.

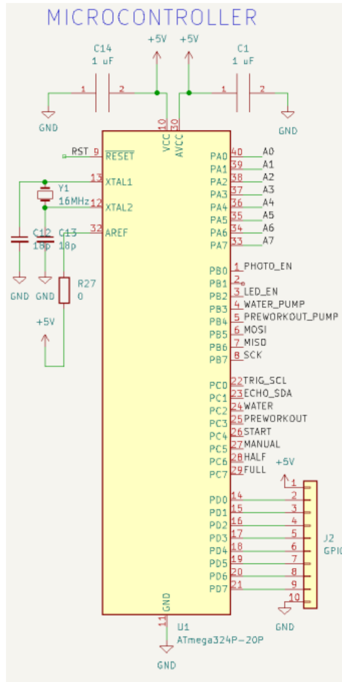

Figure 3: Control Subsystem Schematic

2.3 User Interface Subsystem

The user interface subsystem handles user inputs and preferences. It allows the user to select pre-workout or water, fully filled or half filled, automatic dispensing, and manually dispensing (similar to a traditional water fountain). Finally, there is a main on/off switch to supply power to the device. This subsystem consists of 3 switches (power on/off, preworkout/water, 50%/100%) and 2 buttons (manual dispense, automatic start). The buttons communicate with the microcontroller. Additionally, debounce circuits are included to allow for interrupts in the software registering a single press or flip. All switches and buttons are IP67 waterproof and will be mounted on the enclosure.

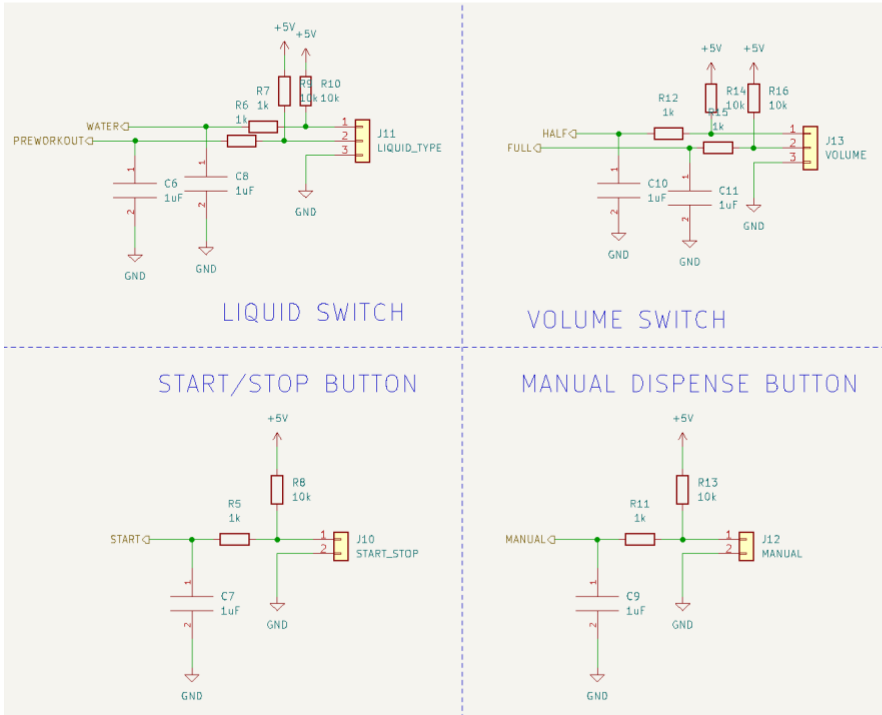

Figure 4: User Interface Subsystem Schematic

2.4 Liquid Subsystem

The water subsystem consists of 1 water tank (for our demo we used a 5 gallon bucket), 1 pre workout tank, and 2 pumps. The pumps are 12V brushless DC pumps with max power draw of 3.6 W. An enable signal from the microcontroller turns on a MOSFET which allows conduction to the correct pump. Since these are inductive machines, a freewheeling diode is placed across the pumps to avoid stress on the MOSFET. The pumps are submersible and will be placed in their respective tanks and be connected to the PCB via jumpers. The pumps will draw liquid from the tank and will pump to the outlet nozzle.

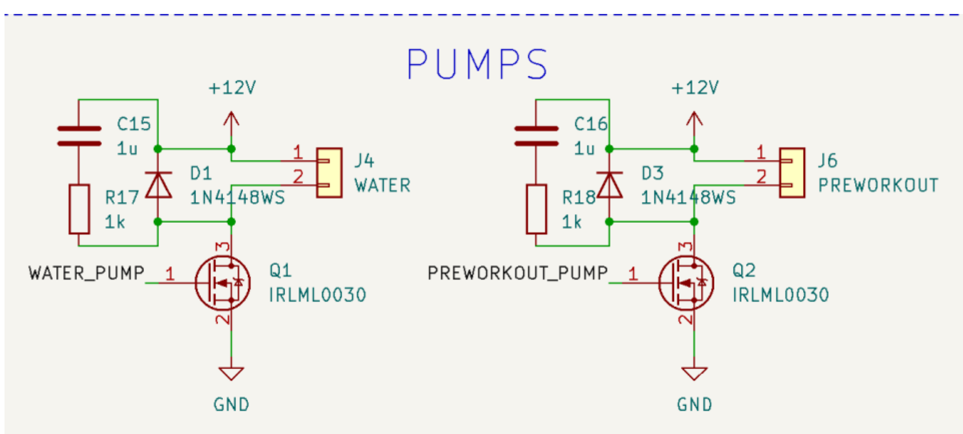

Figure 6: Pump drive circuits

2.5 Sensing Subsystem

The sensor subsystem consists of a laser proximity (water level) sensor and an LED-phototransistor (height) sensor. The water level sensor monitors the water level of the bottle from the top of the device and accurately relays the water level to the control unit through I2C. The height sensing is managed by parallel photodiode and LED arrays. During operation, the LED's are simultaneously turned on during the height sensing period. Each LED is positioned across a phototransistor on the other side of the water bottle. Phototransistors in the path of light increase in current flow, increasing the voltage across the resistor. These voltages are passed through a non-inverting amplifier and are read by the microcontroller through the analog pins. The height of the bottle is then estimated based on trigonometric calculations using average water bottle size and maximum light angle from the highest LED and the gradient of Photodiode currents. Blue LED's are used because they were determined during testing to have the highest sensitivity, and brightness is managed using series resistance. The LEDs are powered with 12V, while the phototransistors and proximity sensor are powered with 5V. The LED/phototransistor sensors are enabled and disabled using digital signals to the respective MOSFET and BJT. The proximity sensor is constantly powered due to low power consumption.

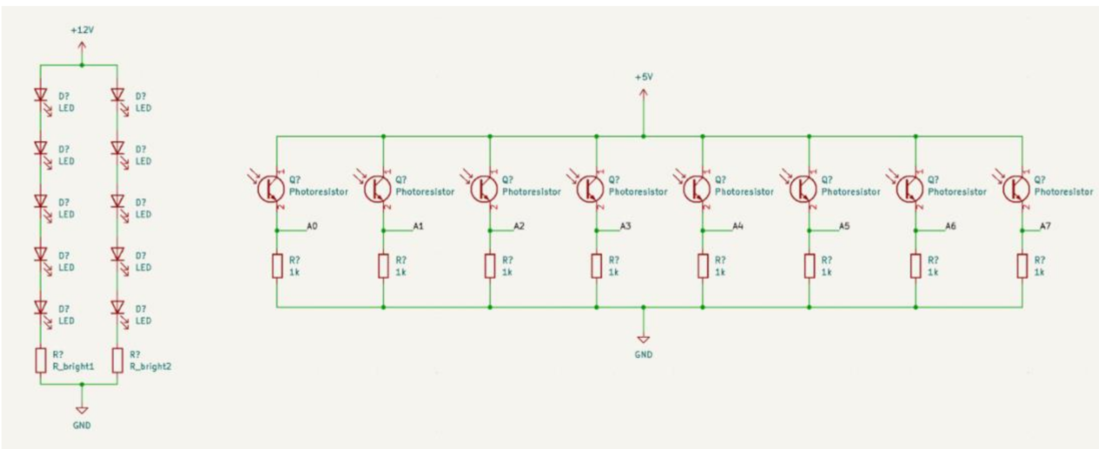

Figure 5: Sensing Subsystem Schematic

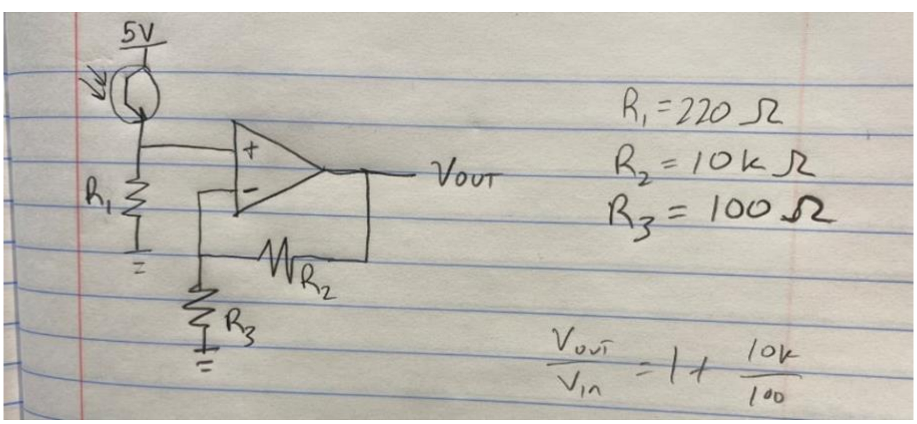

Figure 8: Non-inverting Amplifier for Phototransistor

3 Design Verification

In this section we will be going over how we verified the results of our design. In Appendix A, there is the Requirements and Verification Table. The results below are the verifications for each of subsystems of our project.

3.1 Power Subsystem

We will be going over the requirements that we created for the Power Subsystem. These requirements focus on the power delivered to the PCB and other components of the project to get the 12V and 5V rail.

3.1.1 Requirement 1

The power subsystem supplies 12 V +/- 0.5% from the wall input and 5V +/- 5.0% from the buck converter. We were able to partially achieve this requirement. Our 120 AC to 12 DC converter supplied 12.533 V as seen in Figure 8. This error of 4.442% is outside the desired 0.5% range. The 12V to 5V Buck converter gives us an output of 4.985V as seen in Figure 9. This gives a relative error of 0.3%.

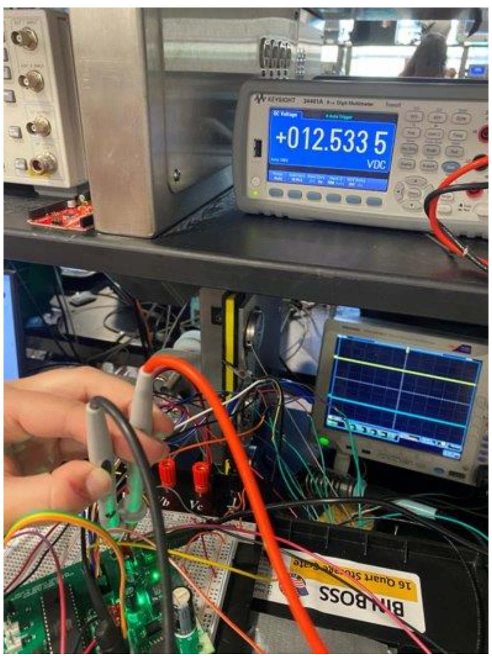

Figure 8: 120V AC to 12V DC output

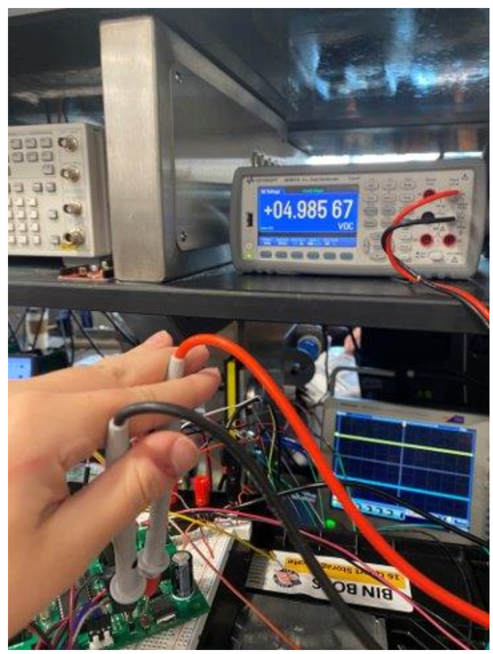

Figure 9: 12V to 5V DC Buck Convertor Output

3.1.2 Requirement 2

The power subsystem delivers up to a 2.5A current while keeping the voltage levels constant. This requirement was met because the system drew a maximum of 300mA during all tests. The 2.5-amp fuse was tested for operation as well.

3.2 User Interface Subsystem

We will be going over the requirements that we created for the User Interface Subsystem. These requirements focus on the user's ability to tell the dictate and change the functionality of the Automatic Water Bottle Filler.

3.2.1 Requirement 1

The User Interface subsystem successfully operates on the input settings. **This functionality is shown in the video link at the bottom of the references ([3]).**

3.2.2 Requirement 2

The User Interface subsystem requires the On/Off switch to allow no power flow when Off and allows power flow to the device when On. This functionality was achieved. If we flip the switch off in the middle of filling up a bottle, the pumps will shut off and the whole system will reset.

3.3 Control Subsystem

The Control Subsystem requirements center around successful operation and reliability of the system.

3.3.1 Requirement 1

The Control subsystem runs the functionality corresponding to a button when it is pressed. We achieved this functionality as we were able to manually fill up a water bottle or automatically fill up the water bottle depending on what button the user clicks.

3.3.2 Requirement 2

The Control subsystem receives water bottle height and current water level data from sensors within +/- 1 cm. We were able to achieve this requirement. You can see the results from testing done in Table 1

Table 1: Water Level Sensor Testing

Conditions/Bottle Type	Actual Water Level	Measured Water Level Range	Accuracy
No bottle	316 mm	308 mm - 320 mm	4 mm - 8 mm
Orange Bottle	161 mm	153 mm - 170 mm	8 mm - 9 mm
Orange Bottle	202 mm	196 mm - 212 mm	6 mm - 10 mm
Orange Bottle	225 mm	213 mm - 236 mm	11 mm - 12 mm
Orange Bottle	254 mm	240 mm - 260 mm	6 mm - 14 mm
Orange Bottle	312 mm	300 mm - 316 mm	4 mm - 12 mm
Metal Bottle	111 mm	106 mm - 113 mm	2 mm - 5 mm
Metal Bottle	156 mm	148 mm - 162 mm	6 mm - 8 mm
Metal Bottle	179 mm	173 mm - 186 mm	6 mm - 7 mm
Metal Bottle	215 mm	210 mm - 222 mm	5 mm - 7 mm
Metal Bottle	314 mm	308 mm - 317 mm	3 mm - 6 mm

3.3.3 Requirement 3

The Control subsystem calculates the "full" water level within +/- 1 cm. **You can see that we achieved this functionality by going to the video link at the bottom of the references ([3]).**

3.4 Liquid Subsystem

We will be going over the requirements that we created for the Liquid Subsystem. The liquid subsystem focuses on everything related to the pumps and anything that must do the with the liquid we are moving in the throughout our system.

3.4.1 Requirement 1

The Liquid subsystem maintains that the tanks and pumps remain leak free while keeping liquid away from electronics. We were successful with this requirement. This is also a requirement that we can continue to work on as it is a potential safety hazard.

3.4.2 Requirement 2

The Liquid Subsystem instructs pumps to draw liquid from the desired tank and turn on and off when instructed by the microcontroller. We were successful with achieving this functionality**. The video at the bottom of the references ([3]) shows that we were successful.**

3.5 Sensing Subsystem

We will be going over the requirements that we created for the Sensing Subsystem. The sensing subsystem primarily focuses on the Adafruit Proximity Sensor 3317 and the phototransistor and led module that we created.

3.5.1 Requirement 1

The Sensing subsystem determines the water bottle's height accurately within +/- 1 cm. We achieved this as our phototransistors are spread at 0.75 inches apart, and we calibrated our algorithm to determine the height at ¼ increments between each transistor. Our final accuracy was within +/- 0.5cm.

3.5.2 Requirement 2

The Sensing subsystem determines the current water level (within +/- 1 cm). We were able to achieve this. Refer to table 1 to see the accuracy of our results for the water level sensor.

3.5.3 Requirement 3

The Sensing subsystems stops filling the bottle if the bottle is removed. We were able to achieve this. We also required that the system can't start any pumps unless a bottle is present and detectable in the correct location.

4 Cost

4.1 Parts

The total list of parts used in the construction of the Automatic Water Bottle Filler is shown below. It's worth noting that some components (miscellaneous connectors, LED's, and resistors) were sourced from ECEB 2072 (design lab) and therefore are not included in total costs.

Table 2: Parts Costs

Part	Manufacturer	Retail Cost ($)	Quantity	Actual Cost ($)
AC-DC 12V Adapter	N/A (Lab-Sourced)	$0	1	$0
Buck Converter LM2596SX	Texas Instruments	$6.73	1	$6.73
Buck Inductor PE-54038SNLT	Pulse Electronics	$3.98	1	$3.98
Buck Input Cap EEU-FR1H471	Panasonic	$1.43	1	$1.43
Buck Output Cap EEU-FR1V181B	Panasonic	$0.68	1	$.68
Buck Diode B540C-13-F	Diodes Incorporated	$0.52	1	$.52
Microcontroller ATMEGA324P-20PU	Microchip Technology	$7.03	1	$7.03
Proximity Sensor 3317	Adafruit	$15.25	1	$15.25
Pumps	Amazon	$10.24	2	$20.48
LED Mixed Pack	N/A (Lab-Sourced)	$0	16	$0
Phototransistor BPW85A	Vishay	$0.70	8	$5.60
Toggle Switch M2013SD3W01	NKK Switches	$5.57	2	$11.14
Push Button EH12NMB3ZX	CIT Relay and Switch	$4.48	2	$8.96
Power Switch JWMW11RA1A	NKK Switches	$9.27	1	$9.27
Barrel Jack Connecter	Wurth	$1.02	1	$1.02
Diode 1N4148WSF-7	Diodes Incorporated	$0.18	2	$0.36
BJT BCW33LT3G	Onsemi	$0.22	1	$0.22
MOSFET IRLML2502	UWM	$0.51	3	$1.53
47uF Aluminum Electrolytic Capacitor	Panasonic	$0.44	1	$0.44
1k Ohm Resistor	Panasonic	$0.10	7	$0.70
10k Ohm Resistor	Panasonic	$0.10	7	$0.10
1uF Ceramic Capacitor	YAGEO	$0.24	8	$1.92
Proto-boards	Adafruit	$2.25	6	$13.50
DIP Socket for Microcontroller	3M	$1.32	1	$1.32
Oscillator	ECS	$0.61	1	$0.61
18 pF Capacitor	KEMET	$0.10	4	$0.40
0 Ohm Resistors	Bourns	$0.10	10	$0.10
1x2 Connector	Molex	$0.21	6	$1.26
1x3 Connector	Molex	$0.21	2	$0.42
2x3 Connector	Molex	$1.39	1	$1.39
2x5 Connector	Molex	$1.50	2	$3.00
5 Gal Bucket	Lowes	$4.48	1	$4.48
3/8 in. Tubing	Eastrans	$11.39	1	$11.39
Total				**$135.23**

4.2 Labor

There are two basic costs for this project: parts/equipment and labor. From the final parts list above, we expect our parts cost to be $135.23 before shipping and taxes. Factoring in a tax rate of 10% (adds $13.52) and a shipping rate of 5% (adds $6.76), the total cost for the materials comes out to $155.51. For labor costs we expect a base pay of $35/hr, and each team member plans to spend 10 hrs/week on the project for 15 weeks, and finally an overhead factor of 3.

$$Labor\ Cost = \#\ team\ members * pay\ rate * \#\frac{hrs}{week} * \#weeks * overhead$$
$$Labor\ Cost = 3 * \frac{\$35}{hr} * \frac{10hrs}{week} * 15\ weeks * 3$$
$$Labor\ Cost = \$47{,}250$$

From these calculations, the projected total cost will be:

$$Total\ Cost = Parts\ Cost + Labor\ Cost$$
$$Total\ Cost = \$135.23 + \$47{,}250$$
$$Total\ Cost = \$47{,}385.23$$

Finally, as we look forward to our device's hypothetical functionality, we note that the sensor-based water bottle filling stations already in use (such as the Elkay EZH2O) typically retail for upwards of $1,400 each [2]. Our device, which costs $135.23 in parts and approximately $200 to assemble, can save a potential buyer upwards of $1,075, and therefore makes our product highly marketable.

5 Conclusion

5.1 Accomplishments

Following a semester of work developing this project, we are pleased to say we achieved full functionality while operating entirely off our custom printed circuit board (PCB). Referencing Appendix A, all subsystem requirements were met (with the exception of part of requirement #1, which will be discussed in 5.2). Furthermore, each high-level requirement was met, as evidenced by successful demo and videos demonstrating functionality.

5.2 Uncertainties

As shown in Appendix A, we had half a subsystem requirement not met – our voltage from the wall was expected to be 12 V +/- .5%, but in reality, was 12.5 V (~5%). In discussions the team agrees that .5% is far too precise an expectation, especially for an off-the-shelf component we knew little about. This component was sourced in the lab, so it's possible it had some pre-existing damage or malfunctions. To address this, we would ideally purchase a better voltage converter. However, since everything operating on our 12 V bus was rated for higher than 12 V, this was not an issue and the device still functioned exactly as expected.

5.3 Ethical and Safety Considerations

Since this device involves electrified components near water, ethics and safety must be considered. A malfunctioning device or misuse of the device (whether intentional or accidental) could result in electric shocks, and significant harm to the user. Referring to the IEEE code of ethics 7.8.I.1 "to hold paramount the safety, health, and welfare of the public … and to disclose promptly factors that might endanger the public or the environment" [1], we are obligated to ensure that the circuitry is protected from any possible contact with water, as well as include warning of electrical shock on the device.

In the prototype demonstrated, there is a warning label regarding electric shocks and saying that the device should only be maintenance by qualified personnel. In Section 5.3, we will elaborate further for how the final, marketable produce would include extensive waterproofing and custom-designed housing to ensure all electrified components are kept away from water. This final product would ensure user safety, thus ensuring our design meets all ethics and safety standards.

5.4 Future Work

As our final device is simply a "proof of concept", there are many possible ways to extend and improve upon this project. First and foremost – housing/waterproofing. We have designed a custom plastic housing with separate enclosures for the water components and the circuitry but given time and budget constraints were unable to construct it. In said housing, the height sensing arrays are entirely covered except for small spaces for the actual LED's and photodiodes to protrude. This design allows for a high level of safety and improves the overall look of the product, therefore making it more marketable. A computer-generated image of the housing designed is shown below:

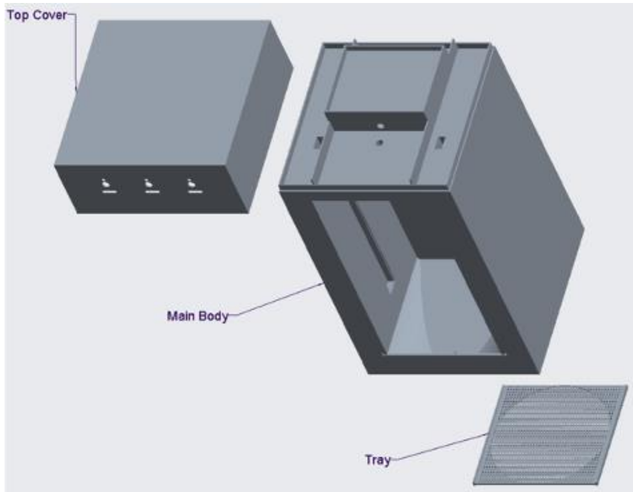

Figure 6 : Housing Design

This product has many different possible applications and high scalability – it could replace existing fountain soda machines in restaurants, or even make drinks in bars. There is virtually no limit to the applications of this device. For some applications, we have suggested replacing the pumps with actuated valves while tying into the building's main water supply. The device still functions the same, but does not require a water tank, thus simplifying installation. Finally, to improve on accuracy and accommodate more bottle sizes, we could extend the height sensing arrays to the bottom of the container and even increase them higher. These are just a few different ways we've come up with to build upon our initial proof of concept, and some team members have even stated commitments to work on this project recreationally in the future.

References

[1] IEEE, “IEEE Code of Ethics,” *ieee.org*, Jun. 2020. https://www.ieee.org/about/corporate/governance/p7-8.html

[2] “Elkay EZH2O Enhanced Wall Mounted Water Bottle Refilling Station, Filtered, Light Gray,” *www.globalindustrial.com*. https://www.globalindustrial.com/p/elkay-ezh2o-lzs8wslp-next-generation-water-bottle-refilling-station-wall-mount-gray?infoParam.campaignId=T9F&gclid=CjwKCAiA9NGfBhBvEiwAq5vSyxEEj8tDfP3LoiNkCLjFMynFxnp1qLHNcXlYZXI7ZDsf-y7xndoBDBoCziQQAvD_BwE (accessed Feb. 22, 2023).

[3] Functionality Video Link: https://youtu.be/XlIvqZJuLNE

Appendix A Requirement and Verification Table

Table 3: System Requirements and Verifications

Requirement	Verification	Verification status (Y or N)
1. The wall adapter delivers 12 V +/- .5% (necessary to stay within voltage ratings) throughout operation of the device a. The buck converter delivers 5 V +/- 5%	- Power on the device. - Place voltage probes on the 12V and 5V rails. - Run the device through the sensing/dispensing process. Verify that the voltage across the rails does not deviate from the desired voltage by more than 0.5% and 5% respectively.	12 V – N 5 V – Y
2. The subsystem delivers up to 2.5A of current while maintaining voltage levels.	- Connect a 5 Ohm resistor across 12V and ground, and place a voltage probe on the 12V rail to verify voltage stays within the desired range. - Connect a 2 Ohm resistor across 12V and ground, and repeat the previous step.	Y
3. The user interface must relay the user inputs to the device and perform the task selected by the user.	- Plug in the device to the wall, then turn on the main power switch. Set switches to 100% full and water initially. - Push the "manual dispense" button. Verify that water is dispensed while the button is pressed. - Push the "auto start" button. Verify that the device begins sensing and dispenses water until the water bottle's height is reached. - Flip % full switch to 50% and hit "auto start". Verify that the device stops dispensing water when the bottle is half full. - Flip preworkout/water switch to preworkout and press manual dispense. Verify that preworkout, not water, is dispensed.	Y
4. The On/Off switch allows no power to flow when Off, and powers the device when On.	- Plug in the device, with the power switch flipped to "off". - Press the manual dispense button. Nothing should happen. - Flip the power switch to "on". - Press the manual dispense button. Water should be dispensed.	Y
5. When a button is pressed, the controller runs the functionality corresponding to that button. a. EX: If manual dispense water is pressed, the controller turns the water pump on.	- Plug in the device and hold down the manual dispense water button,the pump should turn on and water should be dispensed. When the button is released, the pump should stop and water should stop being dispensed.	Y
6. Microcontroller Receives water bottle height and current water level data from sensors within +/- 1 cm	- Measure the water bottle height with a ruler. - Look at the value recorded in the controller, compare the actual water bottle height with this value. Is it within +/- 1 cm?	Y

7. Microcontroller calculates the "full" water level within +/- 1 cm. a. Shuts off the pumps when the current water level reaches the "full" level determined.	- Measure the water bottle height, then calculate 80% of that height ("full" level). - Run the water dispenser, measure the final height of the water dispensed. Is it within 1 cm of the "full" level calculated?	Y
8. Tanks and pumps remain leak free, keeping liquid away from electronics.	- Measure water bottle height with a ruler. - Measure the value the sensors determine. Is it within +/- 1 cm?	Y
9. Pumps draw liquid from the desired tank and turn on and off when instructed by the microcontroller.	- Place a cup with some random amount of water on the tray, with a ruler held near it (allows observation of water level in real time). - Monitor the value recorded in the controller, compare the actual water height with this value. Is it within 1 cm?	Y
10. Determines the water bottle's height accurately (within +/- 1 cm).	- Place a bottle under the dispenser and select automatic dispensing mode. - Before dispensing is finished, remove the bottle. - Does the pump stop within 1 second?	Y
11. Determines the current water level (within +/- 1 cm).	- Toggle switch to "preworkout" and click dispense. Preworkout should dispense. - Toggle switch to "water" and click dispense. Water should dispense. - Liquid should stop dispensing when instructed by the control system. There should be no residual liquid dispensing afterwards.	Y
12. Stops filling bottle if bottle is removed.	- Observe the device over time of use. There should be no leaking water inside the enclosure, and there should be no splashing water around the working area of the device.	Y