Mushroom Incubator

By

Elizabeth Boyer

Cameron Fuller

Dylan Greenhagen

Final Report for ECE 445, Senior Design, Spring 2024

TA: Abhisheka Sekhar

1 May 2024

Project No. 6

Abstract

The following report details our Mushroom Incubator project for Senior Design in Spring 2024. Our incubator can assist in the mushroom-growing process for both amateurs and professionals through precise regulation of the environment within the enclosure. With high-level overviews, detailed engineering descriptions, and discussion of the overall build process, this report covers the team's work throughout the semester which culminated in a successful final product that met all of our requirements.

Contents

1. Introduction

1.1 Purpose

Many people want to grow mushrooms in their own homes to experiment with safe cooking recipes, rather than relying on risky seasonal foraging, expensive trips to the store, or time and labor-intensive DIY growing methods. However, living in remote areas, specific environments, or not having the experience makes growing your own mushrooms difficult, as well as dangerous. Without proper conditions and set-up, there are fire, electrical, and health risks.

The mushroom incubator has humidity and temperature sensors that monitor the internal temperature and humidity, and heating, and humidity systems to match user settings continuously. This includes a visual interface to display the current temperature and humidity within the environment. It is medium-sized (3.5 ft^3) and is able to grow several batches at a time, with more success and less risk than relying on a DIY mushroom setup.

1.2 Functionality

Our product includes:

- a control unit and user interface that allows the user to see the current conditions (temperature and humidity) inside the enclosure as well as give them the ability to change the setpoint values to fit their needs,
- a humidity sensing and control system that constantly regulates the humidity by utilizing a mist spray system to increase the humidity and a high-speed airflow fan system to decrease the humidity,
- a temperature sensing and control system that similarly regulates the temperature within the enclosure utilizes a Peltier module if the temperature is too low and a high-speed airflow fan system if the temperature is too high,
- an air quality control system that provides the enclosure with good, clean air at all times to both ensure the mushrooms get clean air as well as prevent any spores or particles from leaving the enclosure, and
- a lighting system that will turn on at 8 am and off at 6 pm, providing the mushrooms with the recommended 10 hours of light a day.

1.3 Subsystem Overview

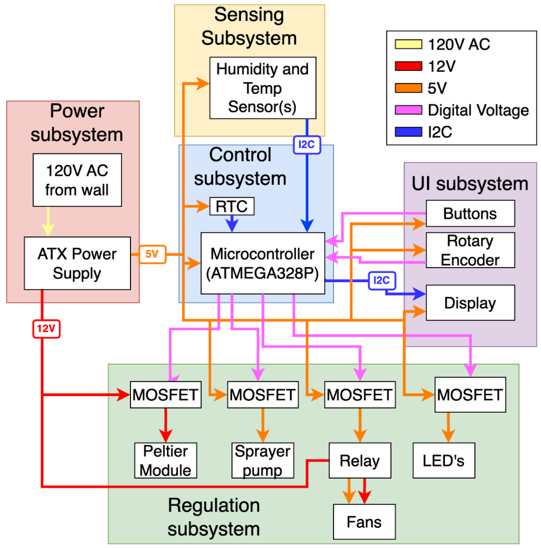

Figure 1: Full Block Diagram

1.3.1 Control Subsystem

The control subsystem deals with deciding the necessary temperature and humidity adjustments for each moment in time-based on the signals sent to it from the sensing subsystem and then changes the relevant digital voltage signals to the regulation system, namely the MOSFETs (metal-oxide-semiconductor field-effect transistor) connected to the Peltier module, sprayer pump, fans, and LED's (light-emitting diodes). Additionally, it sends information to the display through I2C (serial data transmission protocol) regarding the current and set temperature and humidity, which it receives from the sensing subsystem.

1.3.2 Power Subsystem

The power subsystem is responsible for providing and regulating the power sent to all the other subsystems. This receives 120 V ac from a wall socket and supplies both 12 V and 5 V power through the use of voltage regulators. 12 V is sent to the MOSFET for the Peltier module and to the relay for the fans, and 5 V is sent to the sensing subsystem, the UI (User-Interface) subsystem, the MOSFETS for the sprayer pump and LEDs, and the relay for the fans in the regulation subsystem.

1.3.3 Regulation Subsystem

The regulation subsystem handles the actual adjustments that are made to keep the project working as intended based on the signals received from the rest of the system. It includes the Peltier module, sprayer pump, fans, LEDs, and their respective MOSFETs and relay. It receives 12 V and 5 V from the power subsystem, and digital voltage signals from the control subsystem in order to control the heating and humidity elements.

1.3.4 Sensing Subsystem

The sensing subsystem is responsible for detecting the change in the temperature and humidity, and sending those signals to the microcontroller in the control subsystem via I2C. It is powered by the power subsystem.

1.3.5 UI Subsystem

The UI subsystem displays information about the current and set heating and humidity levels (switching between them by pressing the buttons), and allows the user to change the settings via rotary encoders. The buttons and rotary encoders communicate with the control subsystem via digital voltage, and the screen receives I2C data from the control subsystem about what to display. The display communicates with the control system using SPI (serial peripheral interface).

2. Design

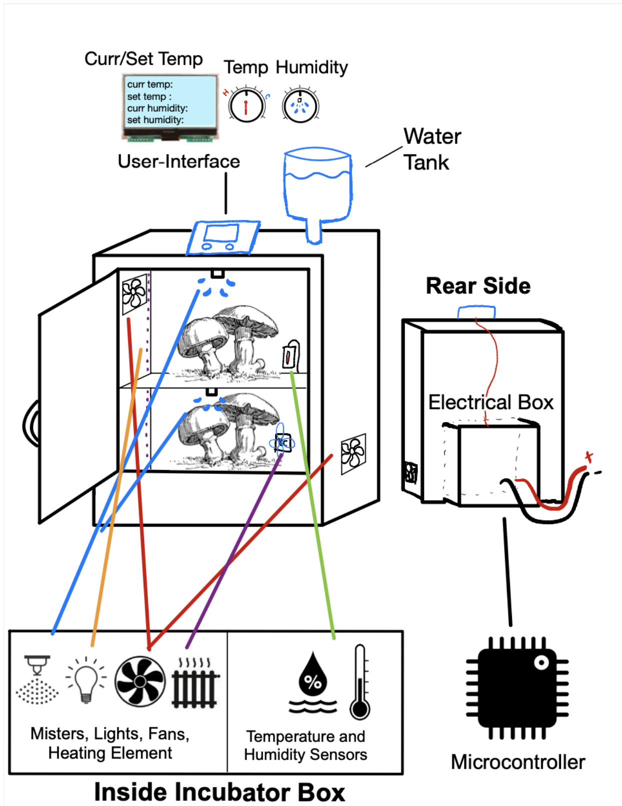

Figure 2: Detailed Visual Aid of the Project

2.1 Equations and Simulations

During the construction of our product, we had to ensure that the enclosure would be able to sufficiently be heated and cooled during operation. In regards to the heating component, we utilized Q = mcΔT and the following values:

- Air at room temp: c = 1005 J/kg * K (note: the specific heat of water vapor in humid air is negligible)
- Volume of the inside of our container: 0.099 m^3
- Density of room temp and pressure air: 1.204 kg/m^3=> Mass of air = 0.119 kg
- ΔT = 1 K

$$Q = mc\Delta T \; -> \; Q = (0.119)(1005)(1) = 119.60\,J \qquad \text{(Eq 1.1)}$$

Our Peltier module can reasonably supply 10 J/s (max is ~20 J/s), so this is about 12 seconds to raise all of the air in the container by 1 ℃ or 1.8 ℉, or just under 7 seconds to raise by 1 ℉, which is well under our projected time of 1 min/℉ even when accounting for ramp-up time.

Along with this, we also needed to estimate the time required to increase our humidity and the volume of water it would require. To do this we utilized a psychrometric chart(App. A.1) with our known volume of air (3.5 cubic ft), a starting humidity of 90%,, a desired change of 1%, and a temperature of 22℃, we found that the ratio of grams of water to grams of air is approximately 0.015. Utilizing a known density of air of 1.196 kg/m^3at 22 ℃…

$$0.015\,\frac{g_{water}}{g_{air}} = \frac{g_{water}}{d_{air} \cdot V_{air}} = \frac{g_{water}}{(1.196)(0.099)*1000} \;->\; g_{water} = 1.78\,g \qquad \text{(Eq. 1.2)}$$

Considering that our sprayer outputs far more water per spray, we knew we would be able to release at least 1.78 g of water within a minute to meet our projection.

To decrease the temperature and humidity, we relied on our airflow fan system. We utilized two equations to account for this, starting with Q = hc*A*dT and the following parameters:

- Q = volumetric flow rate in m^3/s
- RPM (Rotations per Minute) = 1600
- A (Area) = 120 mm * 120 mm = 0.0144 m^2

$$\begin{aligned} Q &= A * v \qquad \text{(Eq. 1.3)} \\ Q &= A * (2 * pi * r * RPM)/60 \\ Q &= .0144 * (2 * pi * .120 * 1600)/60 \\ Q &= 0.29\, m^3/min \end{aligned}$$

Considering our unit is only 0.099 m^3 to begin with, this means that our air cycling rate of 0.29 m^3/minute is well equipped to deal with both lowering temperature and humidity at a rate of at least 1 ℉ or 1 % per minute.

2.2 Design Alternatives

While revising our initial design, we thought of several alternatives and changed our design according to what we determined would be the best fit. In our initial design, we had a much larger incubator, sized at around 6 in. We decided to go with a smaller, more scalable design, in order to fit the timeline of our project while still providing a proof of concept for an expanded version. Additionally, we planned on having a mister for each shelf, mounted directly above each shelf. However, this sprinkler-like system would have required us to add complexity to our Printed Circuit Board (PCB) in the form of a water pump. Instead, the machine shop encouraged us to use more powerful, hand-held, electronic sprayers, which ended up simplifying the sprayer system. The addition of a more powerful air circulation system also allowed us to trim down from two sprayers to one, as we could be more certain that the humid air would circulate between each shelf.

2.3 Design Description and Justifications

2.3.1 Physical Incubator Design

Our project consists of a 15 in. x 15 in. x 27 in. locker that encloses our growing area. On the top of the locker is our electrical box, which houses our PCB, control subsystem, and UI subsystem. The placement of the electrical box on the outside of the locker is important to reduce any shorting, as well as give the user access to the controls without reducing the internal temperature or humidity.

Inside the locker are our sensing and regulation subsystems. We mounted an intake fan on the top left side, and an exhaust fan on the bottom right side. The placement of the fans is chosen due to thermodynamics. We want the inside of the incubator to be warmer than the outside typically, and heat rises. We intend to expel cold air the most, so we built our exhaust fan into the bottom shelf. This way, as fresh air comes in passively (we do not want to lower the temperature in this case), it will circulate through the incubator, and cooler air out.

The humidity and temperature sensor is mounted on the top back corner of the locker. We chose this location to be as far from the path of airflow as possible, to be certain that we measure the temperature and humidity in our box with as little interference from the external temperature and humidity as possible.

We mounted the misting sprayer (supplied with water from the upper water tank) in the path of the intake fan to bring humid air throughout the locker, rather than just on a single shelf.

Finally, we mounted lights above each shelf to be sure each shelf receives equal levels of light.

2.3.2 Control Subsystem

The control subsystem, which interfaces with each of the other subsystems, consists of a microcontroller and RTC module, in addition to several passive components for the microcontroller. The ATMEGA328P was an ideal choice for our microcontroller because of its simplicity and popularity while satisfying the requirements of our design, including its 2-wire Serial Interface and SPI capabilities. We chose the PCF8523 RTC breakout board from Adafruit because of its simplicity and low cost.

2.3.3 Power Subsystem

The power subsystem provides power to every other electrical component in the design. Initially, we planned on using separate power supply modules for 5V and 12V supplies. After experimenting with our components, measuring the actual current draw, and considering the complexity, we decided to replace the separate supplies with a single ATX power supply.

2.3.4 Regulation Subsystem

The regulation subsystem handles the actual adjustments that are made to keep the project working as intended based on the signals received from the control subsystem. One important change we made during our design process is how the fan speed is controlled. At first, our fans could be controlled using a Pulse-Width Modulation (PWM) signal input to vary their rotational speed and therefore airflow. The form factor of these initial fans was not ideal for our project; we wanted airflow in the same direction as the rotational axis of the blades, but the output of these was perpendicular to that axis. We found fans that better suited our needs but discovered they had no speed control. To resolve this, we decided to switch the power supply for the fan between 5V and 12V. Our revised circuit does this using a single-pole double-throw relay to switch the connection for the fan power. For the LEDs, we chose the white color option for the user's visual convenience, and because precise color frequencies are not crucial for the mushrooms' growth. We chose the FQP30N06L MOSFET because it is compatible with logic levels, so we could use these as switches with digital signals from our microcontroller. Omron's G2RL-1-E-DC5 relay was chosen for logic-level compatibility, and we used 1N4001 diodes to eliminate flyback from the relay coil and the sprayer pump's motor. Resistor values were chosen to be appropriate as pull-ups and gate resistors.

2.3.5 Sensing Subsystem

The sensing subsystem, measuring current temperature and humidity inside the enclosure, consists of a single sensor. We chose Adafruit's SHT30 sensor because it can measure both temperature and humidity, reducing cost and parts compared to using two different sensors, and because it has an enclosed shell to prevent damage if water from the sprayer comes in contact with the components.

2.3.6 UI Subsystem

The visible UI subsystem consisted of an LCD (liquid crystal display) screen that allowed the user to see current temperature and humidity as well as use two rotary encoders and two knobs to adjust the setpoint values for both metrics. We separated the temperature and humidity controls to have their own rotary encoder and button in order to provide a clearer, less confusing user interface. For our LCD screen, we chose the ST7789 ADA3787 for several reasons. First, it can be powered by 5V or 3.3V, which gave us flexibility in other areas of the design. Second, it was compatible with existing Arduino libraries, which simplified the software portion of our project. Rotary encoders were chosen over potentiometers to allow for 360° rotation that depended upon the change from the initial position, rather than the position itself. This made the UI more intuitive and allowed an initial temperature and humidity to be set every time.

2.3.7 Printed Circuit Board Design

We designed and ordered a PCB for the complex electrical circuits in our project. The PCB is centered around our microcontroller, but has electrical connections to each subsystem. We chose to use the through-hole package of our microcontroller rather than the surface mount version, so that we could use an IC socket and transfer the chip between different versions of the board. We also chose to use an external 16MHz crystal oscillator, rather than the chip's internal oscillator, for better timing precision. In the first version, the PCB had all the necessary components laid out arbitrarily in order to test functionality. Once assembled, we encountered no issues with this version in terms of programming, outputs, digital communication, power consumption, or anything else requiring major redesign. Version two had the relay for fan speed control, screw terminals and a power jack for simplifying wire connections to external components, and an improved layout for an intuitive user interface. The testing procedure for each version involved soldering on necessary components, uploading simple test programs, and observing or measuring outputs with a multimeter or oscilloscope. We encountered no major issues in the process of designing, programming, and using the PCB.

2.4 Subsystem Diagrams and Schematics

Reference Appendix C for individual subsystem schematics as well as full PCB schematic

2.4.1 Control Subsystem

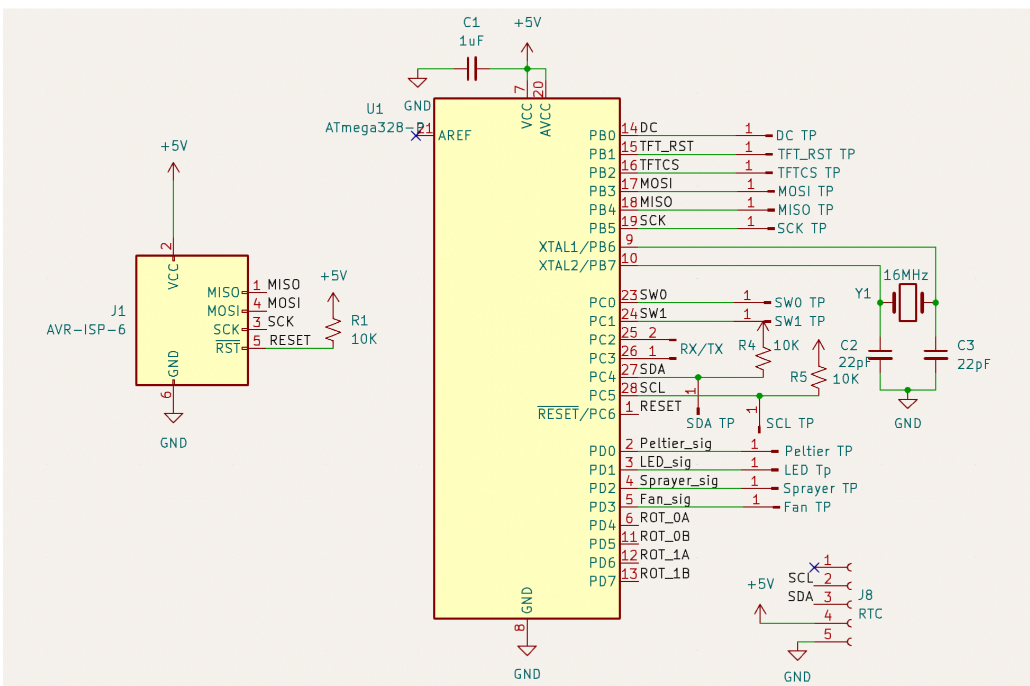

Figure 3: Control Subsystem Schematic

2.4.2 Power Subsystem

Due to the fact that we ended up using a simple ATX power supply, we don't have an individual electrical schematic for this component. It is however a component of each of the other schematics, as it provides the 5V and 12V power supply to each and every subsystem.

2.4.3 Regulation Subsystem

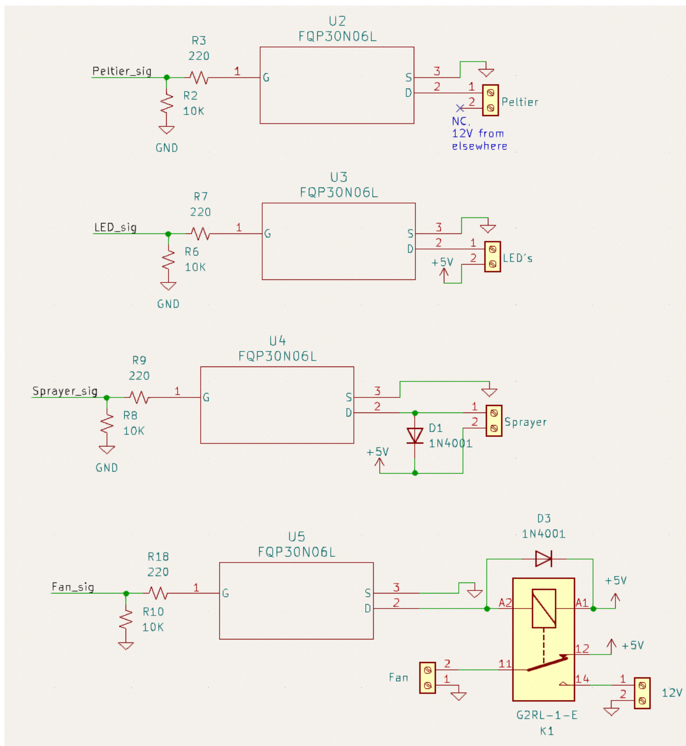

Figure 4: Regulation Subsystem Schematic

2.4.4 Sensing Subsystem

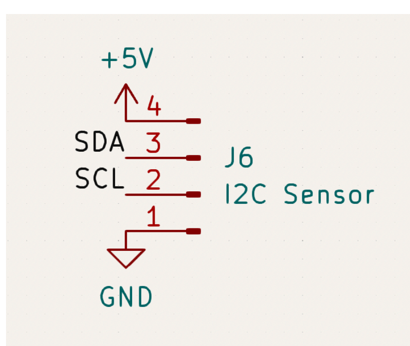

Figure 5: Sensing Subsystem Schematic

2.4.5 UI Subsystem

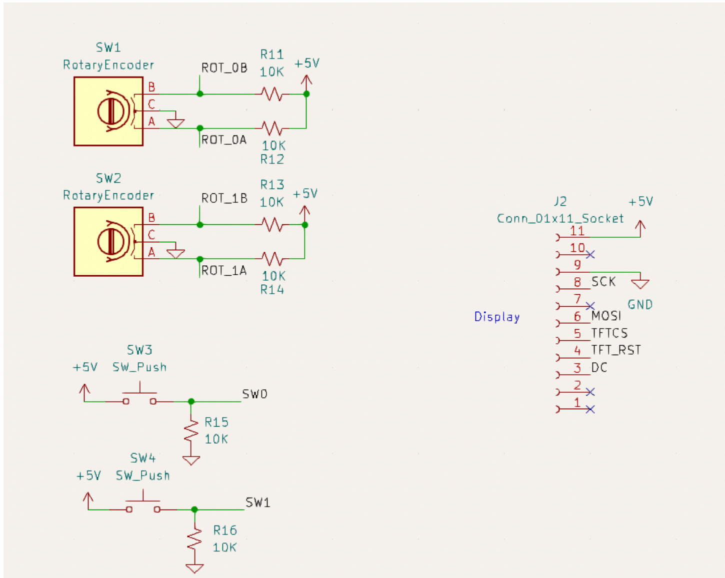

Figure 6: UI Subsystem Schematic

3. Cost and Schedule

3.1 Cost Analysis

3.1.1 Purchased Parts/Materials List

See Table 6 in Appendix A.

3.1.2 Student Wages/Hours of Development

According to the Grainger College of Engineering, the average Electrical Engineering graduate from UIUC makes on average roughly $87,000, or roughly $41.83/hr. Taking that into account for our student wages and assuming each member spends a minimum of 1 hour per day working on the project, we can find the total student wage.

$$T = \$/hr * students * hrs/day * days$$

$$T = 41.83 * 3 * 1 * 68$$

$$T = \$8533.32$$

Thus, our student wage equates to $8533.32 for this project.

3.1.3 External Resources

Outside of the parts we were required to order mentioned in 3.1.1, we also had access to other resources. Such resources include:

- Machine Shop Resources: For this project we required the assistance of the ECEB machine shop, specifically for the actual locker structure that our projects involves. On top of that, we also worked with the machine shop to create an insulating layer on the inside to prevent the escape of any humidity as well as to keep any outside influences on the temperature to a minimum.
- Senior Design Lab Resources: For the creation of our PCB specifically, we require a soldering iron, the solder itself, an oscilloscope, and a multimeter, all of which are provided by the Senior Design Lab.
- Supply Shop Resources: For our UI subsystem, we need knobs and rotary encoders, both of which can be found in the ECE supply shop for no cost. Other items we got from the supply shop included a power button and a TEC heat sink.
- Personal items: Our team already had the RTC module and battery on hand as well as the ATX power supply, so we did not need to purchase these.

3.1.4 Total Cost

In total, with all the purchases and work hours added up, the total cost for our project comes out to be:

$$T = C_{parts} + C_{hours}$$

$$T = 130.25 + 8533.32$$

$$T = \$8663.57$$

In total, the project will cost $8663.57 to complete from both cost of parts and wages.

3.2 Schedule

For the detailed schedule, reference Appendix B.

4. Requirements and Verification

Refer to Appendix A for all Tables mentioned below.

4.1 Control Subsystem R&V

To determine whether or not the control subsystem reacted to user inputs (Table 1, row 1), we used our verification method. We pressed the temperature button and turned the rotary encoder to set the temperature 5 ℉ higher. As we did, the screen updated with the new temperature in less than one second. The same test was performed with setting humidity. Both were within our tolerance ranges.

To determine whether or not the control subsystem read the accurate time of day from the RTC (Table 1, row 2), we used our verification method. We turned our system on, read the time on the LCD screen, compared it with our phones and watches, and found that it was accurate within 1 minute. We waited two minutes and checked again, and the RTC was still accurate within 1 minute, which was within our tolerance range.

4.2 Power Subsystem R&V

To determine if our power supply could provide the necessary power to the system at 5 V, we performed our verification test (Table 2, row 1). After measuring the voltage drop, we found that it was on average between 4.7 V and 5.3 V, which was our tolerance range.

To determine if our power supply could provide the necessary power to the system at 12 V, we performed our verification test (Table 2, row 2). After measuring the voltage drop, we found that it was on average between 11.7 V and 12.3 V, which was our tolerance range.

4.3 Regulation Subsystem R&V

To determine if our fans and heater responded appropriately together, we performed our verification test (Table 3, row 1). After setting the temperature as described and waiting for one minute, we verified that the temperature was raised by 1 ℉. We continued the test by starting the internal temperature at 77 ℉ and setting the desired temperature as described. After one minute, the temperature dropped by roughly 6 ℉. This was within our tolerance range. These are recorded in Figure 4 and Figure 5.

To determine if our fans and mister responded appropriately together, we performed our verification test (Table 3, row 2). After setting the humidity as described and waiting for one minute, we verified that the humidity was raised by 9%. Then, after setting the humidity down to 70% and waiting two minutes, the humidity dropped by 4%. This was within our tolerance range. This increase test can be seen in Figure 6, and the decrease test in Figure 7

4.4 Sensing Subsystem R&V

To determine if our sensing subsystem was accurate, we performed our verification test (Table 4). After checking the thermometer, our temperature sensor was 0.2 °F off, which was within our tolerance range. After checking the humidity detector, our humidity sensor was within 0.4 % humidity, which was within our tolerance range.

4.5 UI Subsystem R&V

To determine if our UI accurately responded to user inputs and promptly outputted the expected value, we performed our verification test (Table 5). After pressing the temperature and humidity buttons, the screen updated to the “Set T/H” menu. After rotating the rotary encoder, the UI displayed the change in less than 1 second, which was within our tolerance range.

5. Conclusion

5.1 Accomplishments and Uncertainties

Our product was able to fully accomplish the goals that we laid out for it at the beginning of the project. We were able to pass all five of our desired functionality requirements and had it working in time for the demo. Our PCB was fully functional and did not require major modifications. We had no major issues with any of the subsystems, and they all passed their respective requirements.

5.3 Future Work and Alternatives

Looking forward, there are a few areas that we could use improvement if we desire to continue our work on the product. They are as follows:

Improved Waterproofing/Insulation: While our product did work during the demo, we still did have some issues with insulation and preventing water leaks. We had originally planned on creating a polyethylene film coating on the inside of the locker to better trap the humidity and temperature, as well as to prevent leaks, but ended up scrapping the idea due to time constraints. This would help the unit to be even more efficient in both its temperature and humidity controlling abilities.

More Stylized Appearance: While not a functionality issue, our final product did look somewhat haphazard due to the short amount of time we actually had to work with the enclosure itself. Given more time, we could add to the appearance of the box in order to give it a more professional look.

Improve misting sprayer: Our misting sprayer, while fully functional, ended up being more of a light jet spray as opposed to the mist we wanted it to be at the beginning of the project. Due to the fact that it was functional, we didn't see a reason to change the component out simply to have to retest and change our already working model. Given time, we could either edit the current mister to force a finer spray or create a new misting system with a different nozzle that would provide the spray that we want. This would help to reduce the amount of water used as well as lower the risk of leakage as there would no longer be as much excess water.

5.4 Ethical Considerations

In relation to the IEEE Code of Ethics,

- Our product does comply with ethical design and sustainable development practices and disclose factors that might endanger the public or environment. In relation to our project, our growing environment does not in any way harm the surrounding environment.
- All of our claims and estimates are accurate and realistic, and we accepted honest feedback and criticism from our TA to make our project as precise as possible.

- We only used equipment if we were trained or experienced with said equipment.

Additionally, we do not condone the use of our product for the purposes of growing illegal substances as stated by US federal regulations.

6. References

[1] “Grow mushrooms you can’t find anywhere,” Shrooly, https://shrooly.com/#faq (accessed Feb. 8, 2024).

[2] “IEEE code of Ethics,” IEEE, https://www.ieee.org/about/corporate/governance/p7-8.html (accessed Feb. 22, 2024).

[3] “Ultimate Mushroom Growing & Incubator kit,” Midwest Grow Kits, https://www.midwestgrowkits.com/Ultimate-Mushroom-Growing-and-Incubator-Kit (accessed Feb. 8, 2024).

[4] ATMEGA328PB - Microchip Technology, https://www.microchip.com/en-us/product/atmega328pb (accessed Feb. 9, 2024).

[5] VGS-35C-15 Cui Inc. | mouser, https://www.mouser.com/datasheet/2/670/vgs_35w-2474743.pdf (accessed Feb. 21, 2024).

Appendix A: Figures and Tables

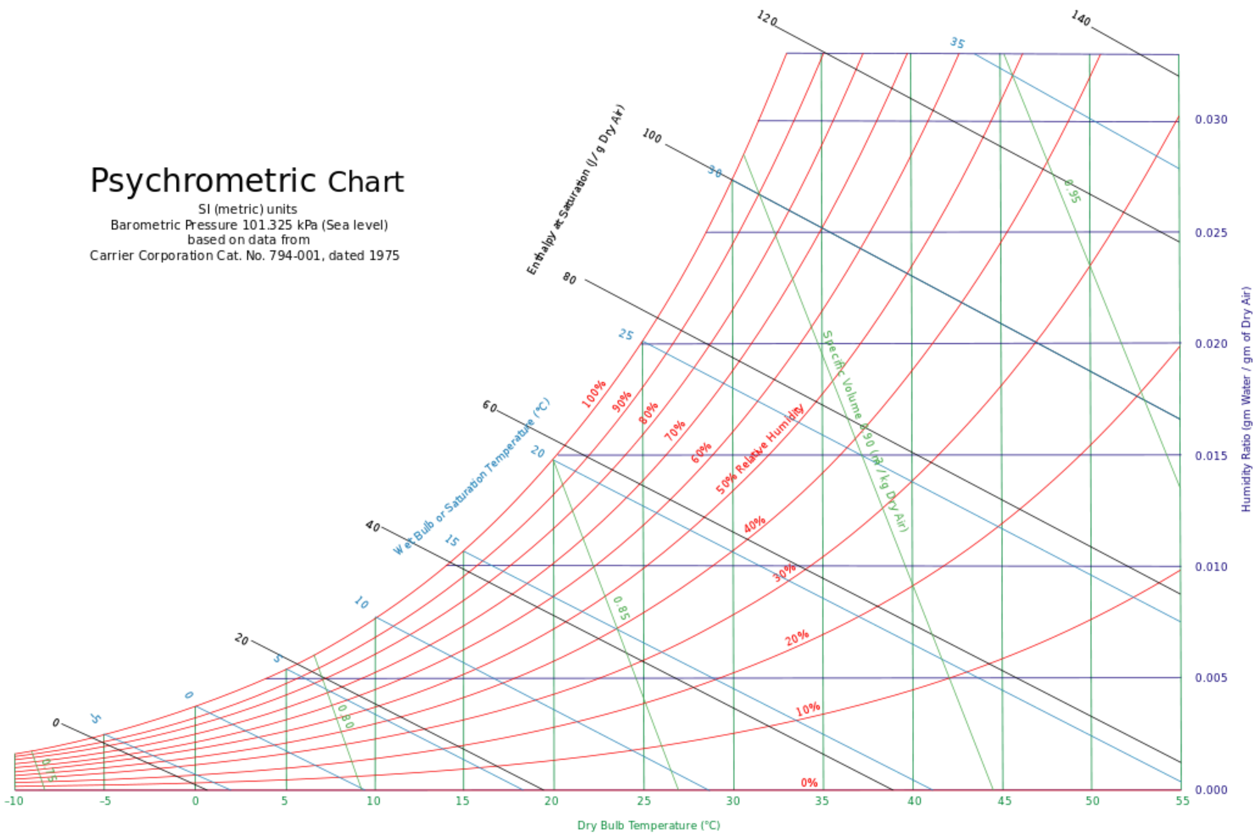

Figure 3: Psychrometric Chart

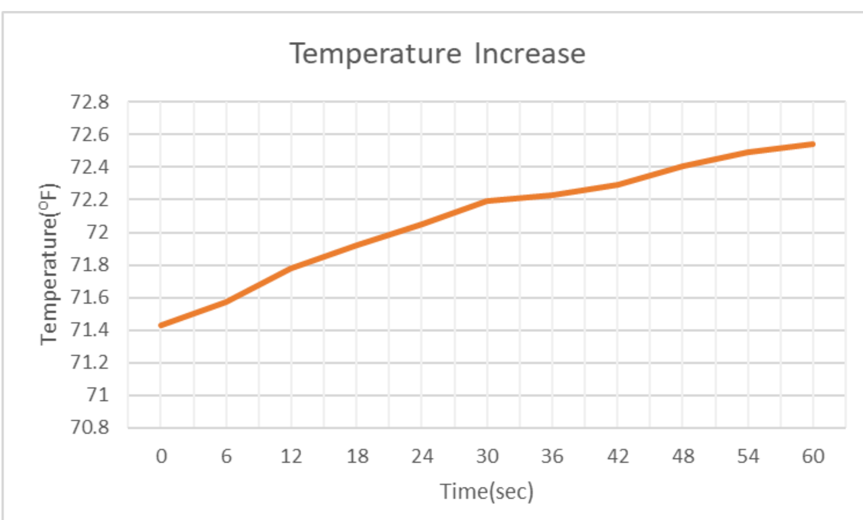

Figure 4: Temperature Increase Test

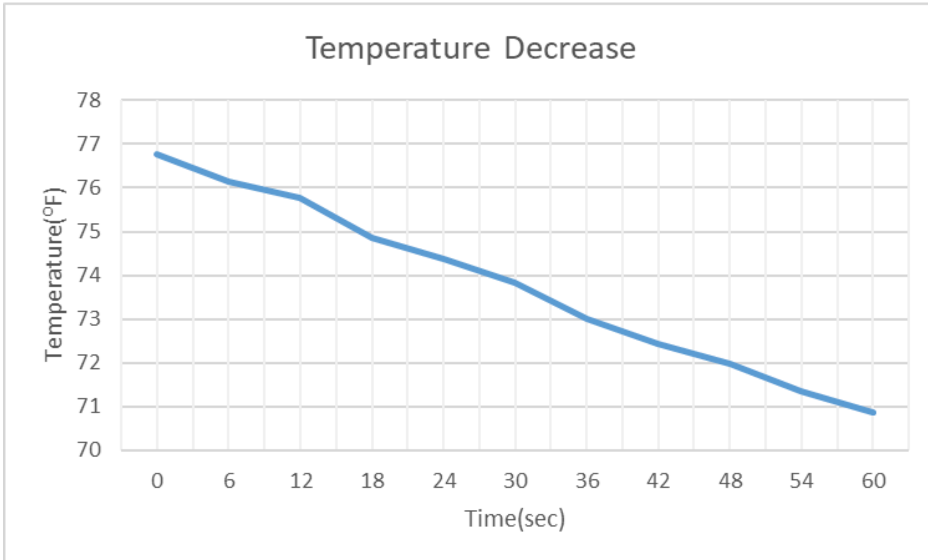

Figure 5: Temperature Decrease Test

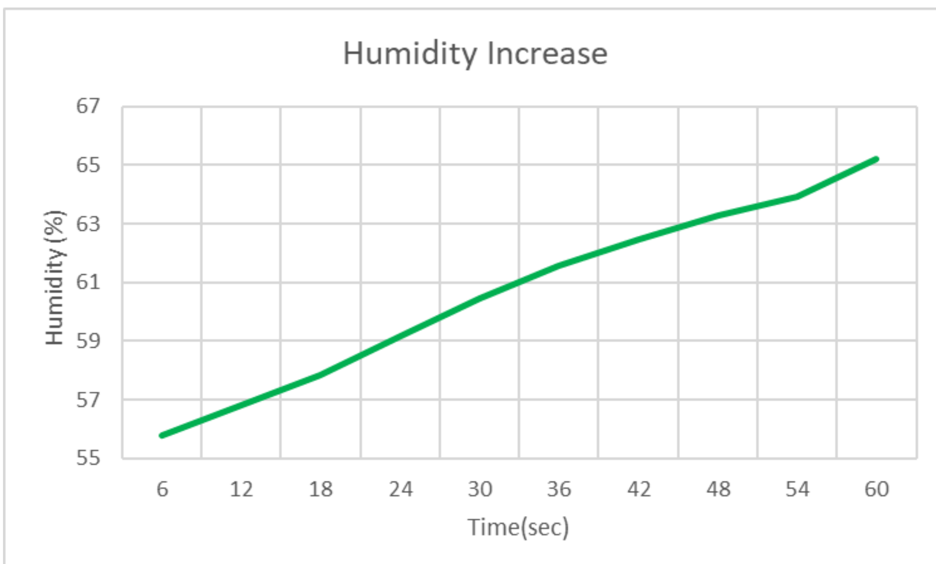

Figure 6: Humidity Increase Over Time Test

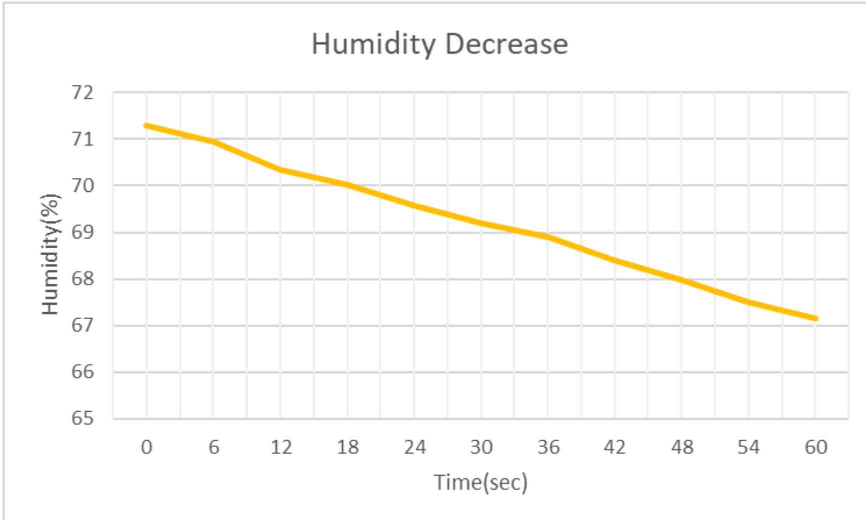

Figure 7: Humidity Decrease Test

Requirements	Verification
The user can control the set points for the temperature, humidity, or using buttons or knobs.	Turn a knob or press a button. Verify that the corresponding set point value on the display changes within 10 seconds.
The display shows accurate time-of-day.	Observe the time displayed. Verify that it is within 2 minutes of the time on an independent synchronized clock, such as an iPhone.

Table 1: Control Subsystem R&V Table

Requirements	Verification
The system must provide a steady and stable 5V(±0.3V) power supply that can source a maximum of 3.6A	Connect wires to the 5V power supply. Ensure the node is on, then measure the voltage between the power and ground and ensure it is between 4.7V and 5.3V.
The system must also provide a steady and stable 12V(±0.3V) power supply that can source a maximum of 5A	Connect wires to the 12V supply. Ensure the node is active, then measure the voltage between power and ground and ensure it is between 11.7V and 12.3V.

Table 2: Power Subsystem R&V Table

Requirements	Verification
The temperature sensing and control demo will show that the heater and fans respond appropriately: If the measured temperature is above the set point, the heater turns off and the fans run at high speed, reducing the temperature towards the set point at a rate of at least 1℉/minute If the measured temperature is below the set point, heater turns on and the fans run at low speed, increasing the temperature towards the set point at a rate of at least 1℉ /minute.	Increasing temperature: starting with air in enclosure at room temperature (~72℉), set the desired temperature to 80℉ using the interface buttons or knobs. Wait 1 minute and verify that the measured temperature has increased by at least 1℉. Decreasing temperature: starting with air in the enclosure at least 5℉ above room temperature, set the desired temperature to 70℉ using the buttons or knobs. Wait 1 minute and verify that the measured temperature has decrease by at least 1℉.
The humidity sensing and control demo will show that the misting system and fans respond appropriately: If the measured humidity is above the set point, the misting system will turn off(if not already) and the fans will run at high speed, cycling air and reducing the humidity at a rate of 1% every 2 minutes.	Increasing humidity: Starting with the door open (humidity < 50%), then close the locker and set the desired humidity to 70%. Wait 1 minute and then check to ensure that the measured humidity has gone up by at least 1%. Decreasing humidity: begin with the setpoint humidity at roughly 70%, then change the value to 50%. Wait 2 minutes, then check to

If the measured humidity is below the set point, the misting system will turn on and the fans will run at low speed, increasing the humidity at a rate of 1%/minute	ensure that the measured humidity has gone down at least 1%.

Table 3: Regulation Subsystem R&V Table

Requirements	**Verification**
This subsystem senses the temperature and humidity inside the enclosure, and sends it to the control subsystem. The sensors must both be able to withstand a temperature of 70-85°F and a humidity between 70-95% for prolonged periods. It must be supplied 5V +/- 0.5V at .98mA.	Sensing Temperature Change: Place a thermometer in the locker and wait for it to produce a steady reading. Ensure that the thermometer value and the sensor values are within 0.5°F of each other. Sensing Humidity Change: Using a humidity detector, record the actual humidity value within the locker. Ensure that the recorded value and the sensor are within 1% humidity of each other.

Table 4: Sensing Subsystem R&V Table

Requirements	**Verification**
When a knob is turned for either temperature or humidity, the updated set temperature or humidity is sent to the control subsystem. Additionally, the current and set temperature should be visible on the display.	Take note of the current setpoint temperature/humidity shown on the UI display. Turn either the temperature or humidity knob and observe whether or not the shown value changes along with the knob.

Table 5: UI Subsystem R&V Table

1 x Temp and Humidity Sensor	$8.95
1 x ATmega328P-PU	$2.89

1 x Peltier Module	$30.00
2 x LED strips	$15.80(7.90 per one)
1 x Sprayer	$8.98
1 x Filter	$9.99
2 x Fans	$16.72(8.36 per one)
1 x Display	$9.95
1x Relay	$3.51
1 x 5V Power Supply	$13.42
Total Cost:	$120.21

Table 6: Individual Component Prices and Cost Total

Appendix B: Detailed Schedule

Week	**Goals**
Feb 18th	Cameron: Design Doc Dylan: Design Doc Elizabeth: Make changes on Project Proposal, Design Doc Assignments Due: Finish Design Doc by 2/22
Feb 25th	Cameron: Design schematic for our PCB, start PCB layout Dylan: Research code associated with the purchased sensor Elizabeth: Github, Research existing UI firmware, start designing interface Assignments Due: Design Review 2/26 and PCB Review 3/1
March 3th	Cameron: Finish PCB layout, submit by 3/5, order parts for PCB Dylan: Begin researching firmware for subsystem interaction Elizabeth: Designing Interface Assignments Due: Order PCB 3/5 and Teamwork Evaluation 3/6
March 10th	Cameron: Plan and reflect Dylan: Enjoy Break Elizabeth: Happy Spring Break Assignments Due: N/A
March 17th	Cameron: Assemble PCB Dylan: Continue research until board is assembled, then begin testing Elizabeth: Finish 1st Round UI, Assignments Due: Second Round of PCB Orders 3/19
March 24th	Cameron: Test PCB with off-board parts Dylan: Test PCB with off-board parts Elizabeth: Test PCB with off-board parts Assignments Due: Individual Progress Report due 3/27
March 31st	Cameron: Documentation, PCB revisions if necessary Dylan:Continue debugging firmsoft/software and assemble full model Elizabeth: Integrate features into model (Fans, UI) Assignments Due: N/A
April 7	Cameron: Finish assembly and begin testing

	Dylan:Finish assembly and begin testing Elizabeth: Testing on full model (Fans, UI) Assignments Due:N/A
April 14th	Cameron: Prepare Plots for Mock Demo (Lighting, control unit) Dylan: Prepare Plots for Mock Demo(Sensing) Elizabeth: Prepare Plots for Mock Demo (Fans, UI) Assignments Due: Mock Demo and Team Contract Fulfillment
April 21st	Cameron: Make necessary changes, prepare for Final Demo/presentation Dylan: Make necessary changes, prepare for Final Demo/presentation Elizabeth: Make necessary changes for Final Demo, practice and prepare presentation Assignments Due: Final Demo and Mock Presentation
April 28th	Cameron: Present and work on final paper Dylan:Present and work on final paper Elizabeth: Prepare final Presentation/ Paper Assignments Due: Final Presentation and Final Paper due 5/1

Appendix C: PCB Diagram

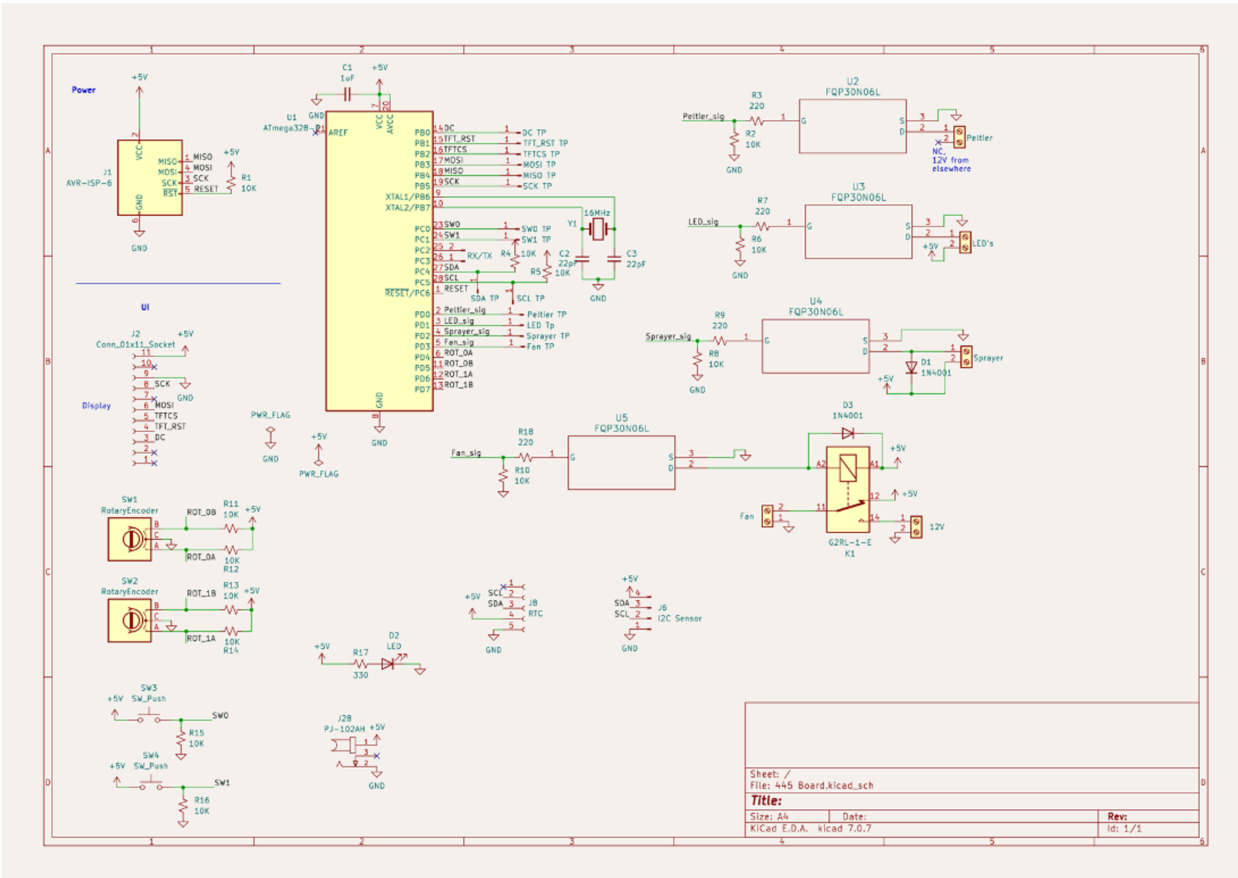

Figure 7: Full PCB Schematic