ECE-493 Team #11

Sustainable Fridge Final Report

Cameron Flores, Nestor Arellano, Matthew Phillips, Ahmed Jaber

Executive Summary

To prolong the shelf life of store bought produce the team built a project that monitors and adjusts the four main factors that affect the longevity of stored produce. The four factors consist of temperature, humidity, ethylene gas sensitivity, and ethylene gas production. To simplify the project the four most common factor combinations were selected as target environments to replicate.

The project was initiated as follows. To protect the desired environments a thermally insulated container would hold the four separate chambers. Additionally, a method of adjusting temperature and humidity was required. A refrigerator would provide 2/3 of the project's basic needs, a source of cold air and a thermally insulated container to maintain desired temperatures. The humidity was provided by a personal water bottle humidifier and a one-liter water container placed in each of the four chambers. Lastly, ethylene gas production and sensitivity were addressed by using online research to create each of the 4 groups that correspond to the four environments, the groups were populated by produce with similar temperature and humidity preferences and ethylene gas compatibilities.

To control the components, a Raspberry Pi microprocessor was implemented. The Raspberry pi would take temperature and humidity data from each of the four chambers using four individual sensors and determine which chamber requires a change of environment. To change temperature the Raspberry Pi turns on the refrigerator to decrease the temperature and turns off the refrigerator to allow the chambers to warm up if needed. Air delivery to a desired chamber was done through a collection of four solenoid valves to allow the air to travel from a central cold air hub through an air hose to the desired chamber. When a humidity adjustment is needed the Raspberry Pi can power on and off the humidifier within each chamber.

Component power delivery was achieved using multichannel relay boards, the 3.3V and 5V Raspberry Pi pins and a 12V DC power supply.

The software for this project was done completely in Python3 and consists of a main file that utilizes various import statements to reference other Python3 scripts saved on the Raspberry Pi that each add the necessary functionalities that allow this project to operate as desired.

To test this project a sample from each of the pre-determined groups will be tested in their corresponding chamber with their desired environments. The shelf life of the produce will be compared to that of produce stored in normal crisper drawers. It is important to note that the control groups tested in the normal crisper drawers were also separated using the groups designed for the project, to get the best performance from the stock refrigerator. The tests will compare the physical appearance and mass of the produce as it changes over time.

The sustainability aspect of the project comes from the attempt to reduce food waste, to allow more of the purchased produce to be consumed by the user. An additional benefit to come from this project and its potential reduction of food waste, is the potential to influence the user's diet in a healthier direction by having healthy food readily available for longer periods of time.

Table of Contents

1. Approach

1.1 Project Origin

The idea for a project to encourage healthier eating habits, prolong produce shelf life, and reduce food waste was originally proposed by Dr. Margaret Slavin and Dr. Nathalia Peixoto. The project was then added to a list of potential capstone projects for the ECE 492 and ECE 493 courses and presented to enrolled students by Dr. Piotr Pachowicz. The design and circuit were drawn and modeled by the team during ECE 492.

In ECE 493 the final design parameters and User interface requirements were proposed by Dr. Margaret Slavin. The necessary workspace for this project was provided by Dr. Peixoto. Parts acquisitions and assembly were made possible by contributions by the team members, Dr. Peixoto, Dr. Pachowicz, and Sarah D'alexander through the financial support of the Patriot Green Fund.

1.2 Project Solution

The solution to the project was to have 4 individual chambers to be used for the fridge. Each chamber would have the option to change its environment to any of the preset environments given, adjusting its temperature and humidity, or the user could turn off the chamber and not adjust the temperature/humidity.

For the preset environments, we had them selected based on the type of produce that would be best suited for the which temperature and humidity, the amount of ethylene it produces, and the effect ethylene has on that product. The results of which type of produce fits in which environment can be seen below in figure 1.

Recommended Groups

Group A: 34 (F) : 90%	Group B: 32 (F) : 90%	Group C: 44 (F) : 90%	Group D: 50 (F) : 90%
Acerola cherry	Celery	Bell pepper	Tomato
Artichoke	Mushrooms	Pineapple	Passion fruit
Rutabaga	Peas	Asparagus	
Tamarind	Arugula	Tomatillo	
Pear	Chard	Okra	
Avocado	Cilantro	Summer squash	
Apple	Collards	Papaya	
	Parsnip		
	Spinach		
	Watercress		

Figure 1: Food Test Groups

The chamber's temperature would be changed using the fridge's fan to blow cold air to the chambers from the freezer. The solenoid valves would then pass air if the chambers needed an environment change. The chamber's humidity would be changed thorough the mister located inside each chamber. When humidity needed to be added, the mister would start spraying a mist to raise the humidity in the environment. Once it reached the threshold humidity, the humidifier would be turned off until needed again. The environment would then be monitored by the temperature/humidity sensor inside each chamber.

The edits in the chambers and status of the chambers would be shown to the user using an LCD screen displaying current values and options for the user to edit themselves. These choices would be selected through push buttons, allowing the user multiple choices at a time in the menu. All the data displayed on the LCD would be taken from the Raspberry Pi, which would communicate with the electrical components and run the project.

1.3 Alternative Designs

There are several alternative designs that could be implemented. All that is required is an insulated compartment that can have cold air blowing into it and a mister. One straightforward design is just a transportable picnic cooler with a divider between two compartments. On one adjacent compartment there would be a mister controlled by an embedded system and on the other; some ice that the user would have to supply. Then whenever the air temperature rises above the desired level, there would be an air valve in between the two sides of the cooler that would open as necessary to cool the contents. As for draining out the excess water from the cooler, there could be a rack to keep the food off the bottom of the cooler and a standard water tap can let the user more easily drain the water. This alternative design could run off batteries and would be completely transportable.

As stated, any thermally insulated compartment could be modified accordingly. A refrigerator would not necessarily have to be divided up into such small sections and if there was only one type of food involved, could also be environmentally controlled in its entirety. In fact, this would make the design modification simpler because the fridge would be programmed to turn off and there would not be a need for the valves and the hoses.

A professionally manufactured sustainable fridge would have usable compartments built in and would optimize space. It would also use the standard water supply that goes to the fridge icemaker to fill the misters. These modifications could be used to make a product that can be marketed.

1.4 Contributions of each member

Cameron Flores: Project Manager, helped mainly in software, integrating the code files together as well as the User Interface code. Also helped the Hardware team.

Ahmed Jaber: Helped mainly in software, designing the needed algorithms and creating the needed helper code for internal functions. He also helped in conducting the control test and conducted the final test.

Matthew Phillips: Helped mainly in hardware, designing the needed hardware setup for air ducts, drains, wiring, and assembly.

Nestor Arellano: Helped mainly in hardware, performing the needed soldering and crimping for the electronic components, alongside the main hardware assembly. Also helped in conducting the control test.

2. Technical Section

2.1 Hardware:

For the hardware seen below, our main processing module is the Raspberry Pi 3B+. This will be the main processing unit to control all the electrical components and make them perform their tasks at the required times. One of these components is the DHT22 Temperature/Humidity sensor, its purpose is to monitor environments. It will check the chamber environment when asked by the Raspberry Pi. To conserve power, relays shut off the DHT22's between tests. When the temperature or humidity is not in the correct range, the Raspberry Pi will close the assigned relays to that chamber to either adjust humidity with a humidifier or adjust temperature by opening a solenoid valve. These are connected through soldering on a PCB Board and the Raspberry Pi GPIO pins.

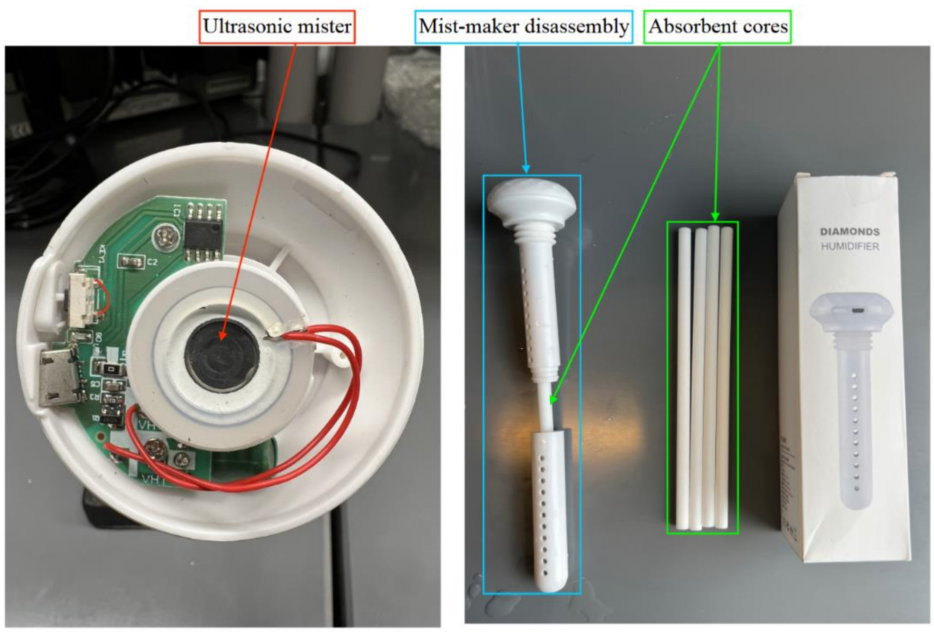


Figure 1: Mister Design (Left) and Mister Parts (Right)

Since the misters conveniently only required the 5 volts that our Raspberry Pi 3B+ supplies, we connected those 5 volts to the common of each of the 4 relays that controlled our misters. By doing this we were able to turn on and off the 5-volt source with 3-volt GPIO pins. To avoid using too much amperage at one time, we were careful to program the Pi so there is only one mister running at a time.

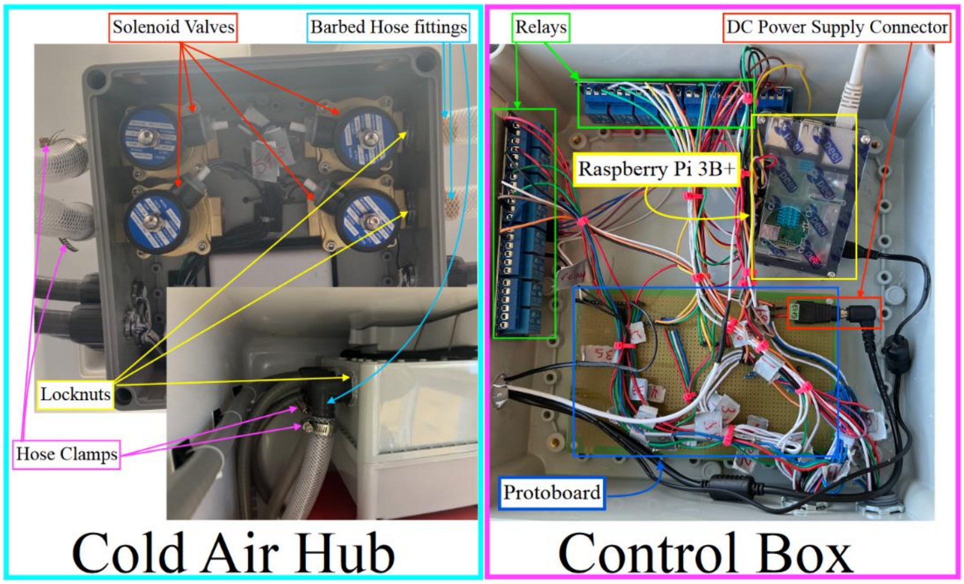


Figure 2: Cold Air Hub (Blue), Control (Purple) Layout

The cold air hub in the sustainable fridge collects the cold air that an unmodified fridge blows into the lower chamber. Once the cold air is contained, four solenoid valves control air flow to the four coolers in the bottom of our fridge. All solenoid valves are normally closed which means that they require 12 volts to enable the air to flow through them. The valves are required to be closed most of the time, this lowers power consumption.

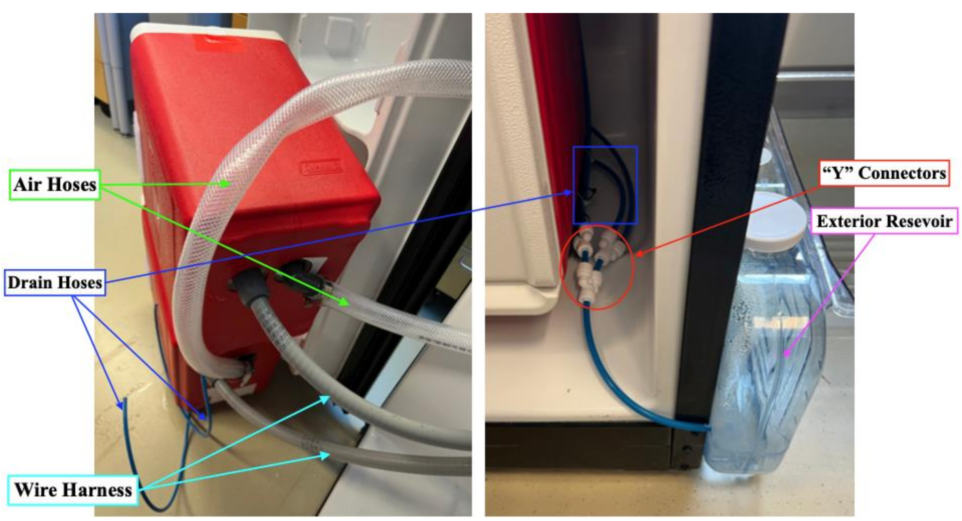


Figure 3: Chamber Characteristics + Tubing

Using a system biased on gravity the sustainable fridge drains excess water from its coolers. To collect the extra water, holes were drilled into the bottom right corner of each cooler where a small hose was then inserted and the gap around the hose was sealed with silicone sealant. It was necessary to slightly prop up the front of the coolers so the water would run to the back. From the coolers, the hoses lead the water to a clear container mounted outside of the fridge.

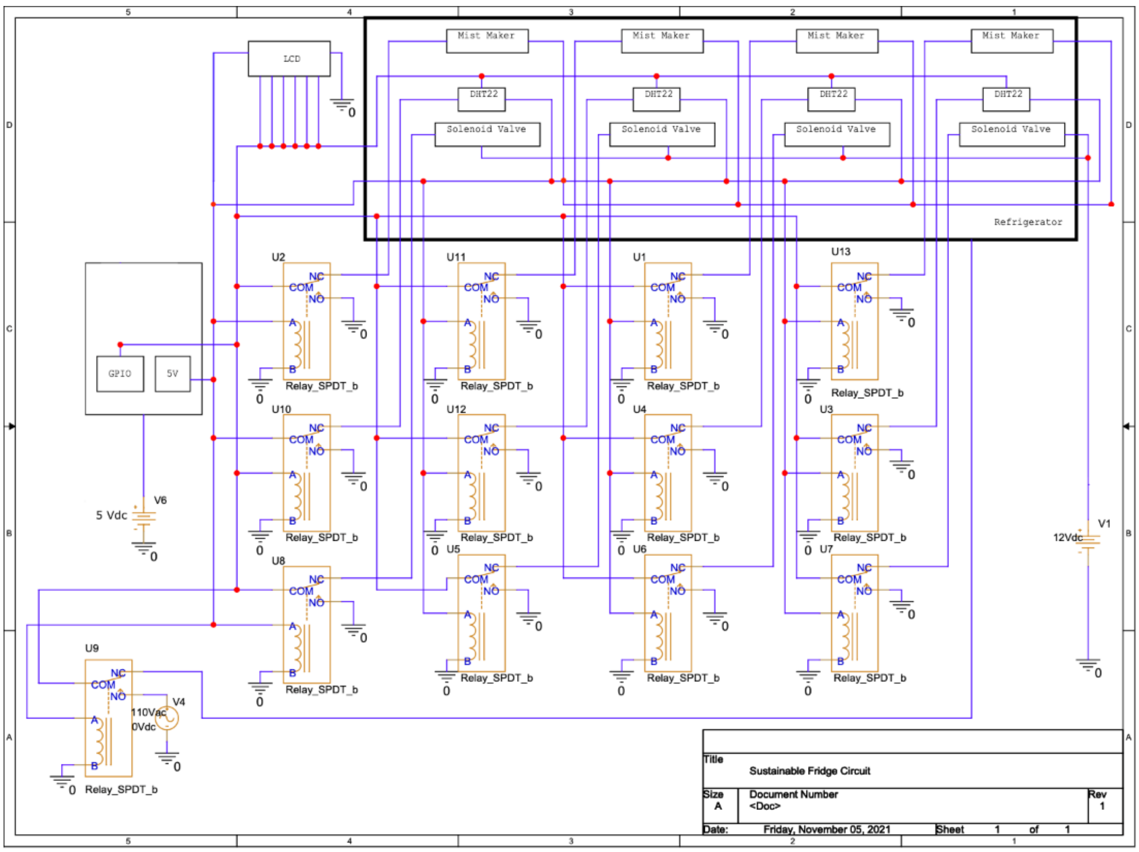


Figure 4: Project Circuit Diagram

2.2 Software:

Code was first written in separate files, each with a designated functionality. After that, we integrated the files and tested them extensively until we reached the proper functionality for the main testing.

First, we made a main file, that performs the essential functions for the system. This file calls functions from the other files to form two modes of operation, user control mode and checking chamber mode.

At startup, all pinouts would be initialized and components like the LCD will be set up. The fridge would then start in user control mode, where the user can decide to change the characteristics of the 4 chambers such as turning it on and off, see the status of the fridge and being able to change the environments by selecting pre-selected environments or manually selecting temperature and/or humidity. This mode uses a combination of the menu file, button file, temperature file, relay file to functionally work.

In the second mode, checking chamber mode, after no inputs for 30 seconds, the fridge will go start checking the chambers temperatures. The raspberry pi then compares the difference of those individual chambers to their target temperatures. It then goes and checks to see if a user made an environment change before going to the checking chamber mode. If a change has been made by a user, then that chamber goes in front of the line before the non-user changed modes. If no change by a user has been made, then the chamber with the greatest difference in its target temperature and its initial temperature goes first and the smallest difference goes last. Then in the process of changing an environment, it will change one environment one at a time by adjusting the temperature and humidity to its right level by opening the solenoid valve to the chamber to allow cold air to come through and/or a mister spraying water in the chamber. Once both temperature and humidity have been checked it will go to the next chamber needing an adjustment change. Once all chambers are checked it will go to the beginning of the list and check to see when a chamber needs to be changed again. If the user also made a change to turn off the chamber, the chamber will still be in the list, it will just be skipped in changing the environments in the chambers. This mode uses a combination of the relay file, listing file, environment file, menu file, and temperature file.

While checking the chambers, the sustainable fridge will update the user on the LCD screen with each chamber's temperature, humidity, and a notification to see if water needs to be added to the chamber or not. If the chamber is off, it will only display "Off" to the chamber on the LCD screen and a signal to see if the water level needed to be refilled or not. Seen below is figure 5 showing what the LCD screen can display.

Figure 5: LCD Display in Chamber Check mode

For the menu file, its main functions are to initialize the LCD screen and display messages for the user to read. For the file, we used open-source code to add i2c functionality as the main mode of communication with the LCD screen. This allows us to use the I2c driver pre-installed to the LCD to display its messages without using multiple GPIO pins [1]. For the user control mode, it will display based on a mealy state machine with a number combination from the button file being its inputs to determine what will be displayed at what time. For the checking chamber mode, it will show each chamber's temperature, humidity, and notification if it needs to fill its water level based on what the "environment" file sends out display. The display in this mode will update the current value of every chamber's environment every two minutes when in the process of changing an environment or looking through each environment until a change needs to be made in a specific chamber. Seen below in figure 6 is showing how the menu displays are set up.

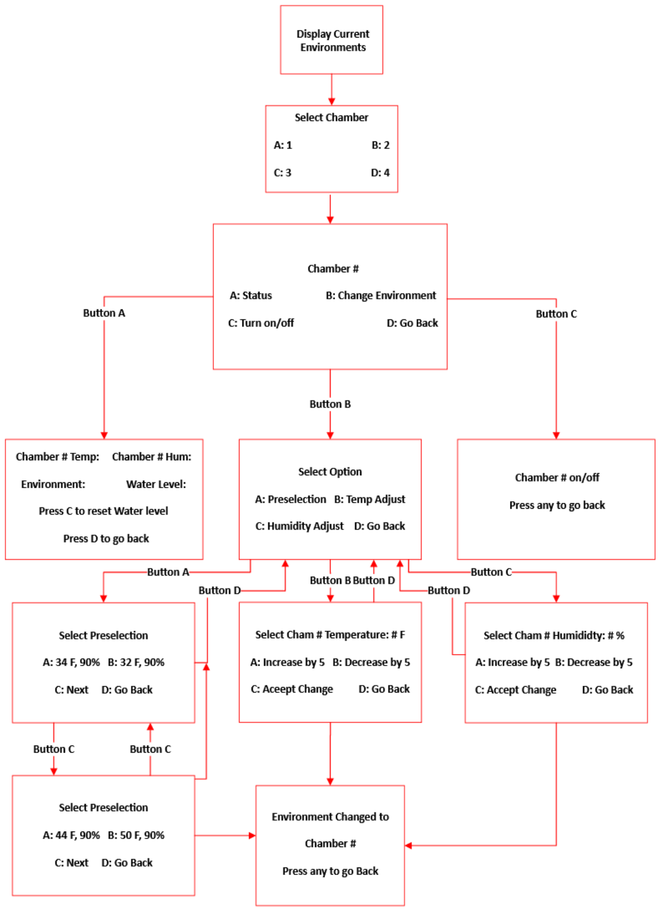


Figure 6: Menu Flowchart

In the button file, we implemented the code needed to handle button presses, as a conversion from GPIO signals to the corresponding buttons. The button file would work communicating back and forth with the menu file as the outputs of the menu file will be used as inputs for the button file and the outputs of the button file would communicate with the menu file as inputs. This allows at specific points in the mealy machine what of the 4 button inputs would do what. The outputs from the button file are user inputs that would be used to communicate with the menu file in either user control mode or checking chamber mode. For completion, we also had printouts displayed to the terminal indicating which button was pressed as soon as it occurred to assure the correct button was pressed.

In the Environment file, the main purpose of this file was to change the temperature and/or humidity in each chamber and update the water level for each mister when in use. This is done by taking the results from the listing file and choosing from the list which chamber will be checked first, and which chamber will be checked last. It will check both temperature and humidity first and see if it is inside an appropriate range of temperature and humidity. If so, it will move on to the next. If not, it will focus on adjusting the temperature and humidity. For temperature, it will open the solenoid valve, to allow cold air to flow into the chamber and will close once inside the chamber it is at the appropriate temperature. By adjusting the humidity, it will turn on the mister and spray mist out until it is at the appropriate humidity level. Once then it will turn off the mister. When adjusting the environment, it will update the status of the chamber every two minutes. Once then it will go to the next chamber in the list that is called and will repeat the list once it reaches the end.

When using the mister, and when it is done reaching the correct humidity level it will then calculate the amount of water left in the jar. The equation is shown below, once the jar gets around 90mL or 10% of the water level, it will keep the mister off and skip the requirement to reach appropriate humidity level. This is to avoid damaging the mister long-term and any unnecessary power consumption. The environment file communicates with the menu file, relay file and temperature file. Below in figure 7 is the equation to determine the water loss during a misting spray.

$$Water\ Container\ Volume = 946.4\ mL$$

$$Spray\ Rate\ of\ Misters = 35\frac{mL}{hour} = .58\ \frac{mL}{min}$$

$$Apporpriate\ Amount\ of\ Water = 940\ ml \geq x \geq 900\ mL$$

$$Minutes\ of\ spraying = b$$

$$Remaining\ Water\ in\ Jar: W(b) = x - .58 * b$$

Figure 7: Equation to Determine Water Loss

The environment file updates a text file "output.txt" only when needed to adjust the temperature and humidity in a specific chamber to appropriate levels and the time it took to do that change.

In the Listing file, the purpose of the file was to get the current temperatures and humidity in each chamber and sort out the chambers for the environment file to decide which chamber would be looked at first. The first thing it does is get the current temperature, both in Fahrenheit and Celsius, and humidity. Then we move that data over to part where it sorts out the code. It will look at each chamber's temperature and compare each preset temperature. As stated before, it places the first chamber with the greatest difference in temperature and the least temperature difference last. As stated before, it will check if a user has changed the environment and will then put those chambers first in the list no matter what.

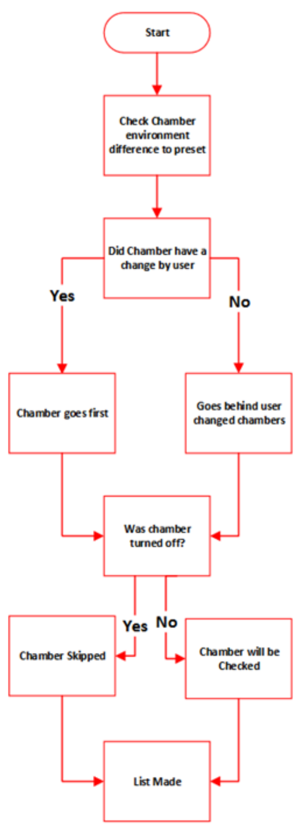


Figure 8: Sorting Chamber Flowchart

In the temperature file, the main use was to read temperature and humidity from the chambers. For this We used open-source code from Adafruit to properly read the data off the temperature-humidity sensors, and format them into a format ready whenever called upon by the other files [2]. When using the temperature sensors, there is a risk of two things occurring. One is the chance of an error value coming from the sensor due to a bad reading. The second is that the reading comes off incorrect due to the signal being sent back to the Pi may be off due to hardware issues. If either situation were to occur, it would read the environment again 2 more times. If it cannot read a reasonable value after 3 times, it will turn off the sensor and turn it back on again to fix the issue. It will repeat this until we get an appropriate value.

In the relay file, its purpose was to initialize relay ports and turn on and off relays when asked by other files.

3. Experimentation

3.1 Experiment overview:

We split produce into four main groups in our testing; Group 1 includes Apple, Pear, and Avocado; Group 2 includes Kale, Spinach, and Celery; Group 3 includes Bell Pepper, and Squash; Group 4 includes Tomato. Our testing process involves placing produce (one of each type) in a fridge then measuring mass and taking a picture of the produce every day for at least three weeks. For our control, we used the refrigerators at home which had an internal temperature of 42 degrees and humidity of 11%, storing each group in a separate built-in crisper. Since most refrigerators (as found in our homes) only had two crisper drawers, we split the testing groups among two team members, Nestor (Groups 1 and 2) and Ahmed (Groups 3 and 4).

As soon as our Sustainable fridge was ready for testing, we bought fresh produce belonging to the four groups. We then conducted another test for the experimental group which used the Sustainable Fridge with each group in a corresponding crisper, for another three weeks.

For the second test we were to test the power consumption of both the control fridge and the sustainable fridge over a 21-day period in kWh. This data would be converted to see the cost to run the fridge in the 21-day period and over a year. This data would be recorded by using a power meter plug to record the kWh. Then using this data, we would see if overall the energy consumption of the fridge overall was less expensive than buying fruits and/or vegetables

A third test was to see the sensor data when in the process of recognizing the change in a temperature or humidity and seeing how long it would take to return to a set environment. The time being recorded during the 21-day period with the 4 test groups in the chambers.

3.2 Data Collected from Experiments

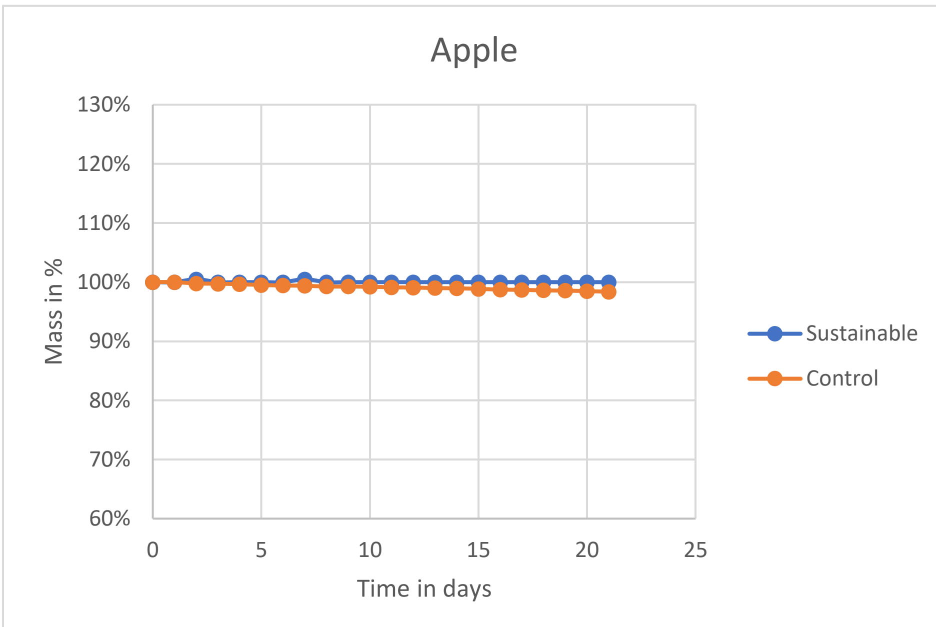


Figure 9.1: Control Fridge vs Sustainable Fridge: Apple Mass

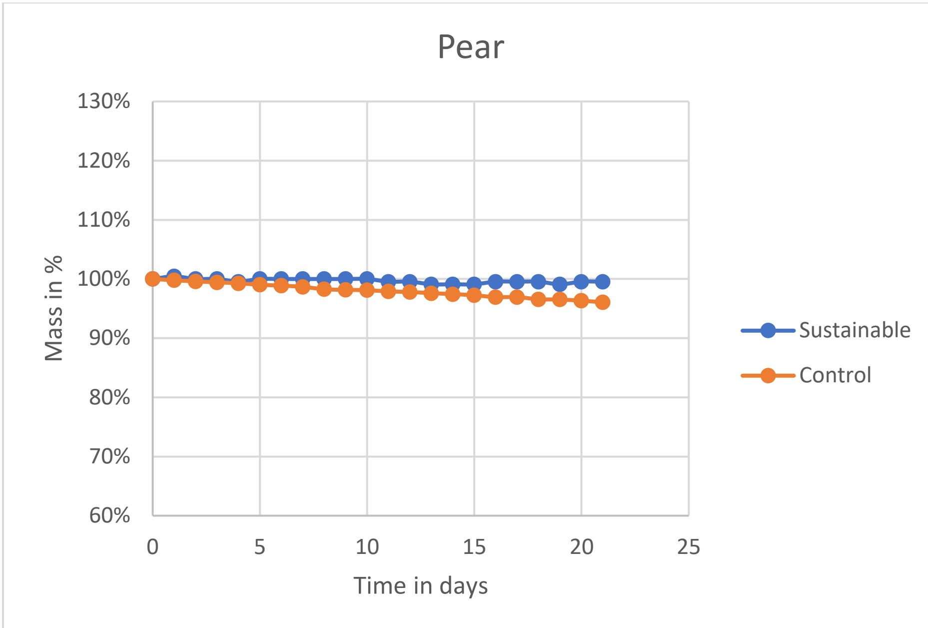


Figure 9.2: Control Fridge vs Sustainable Fridge: Pear Mass

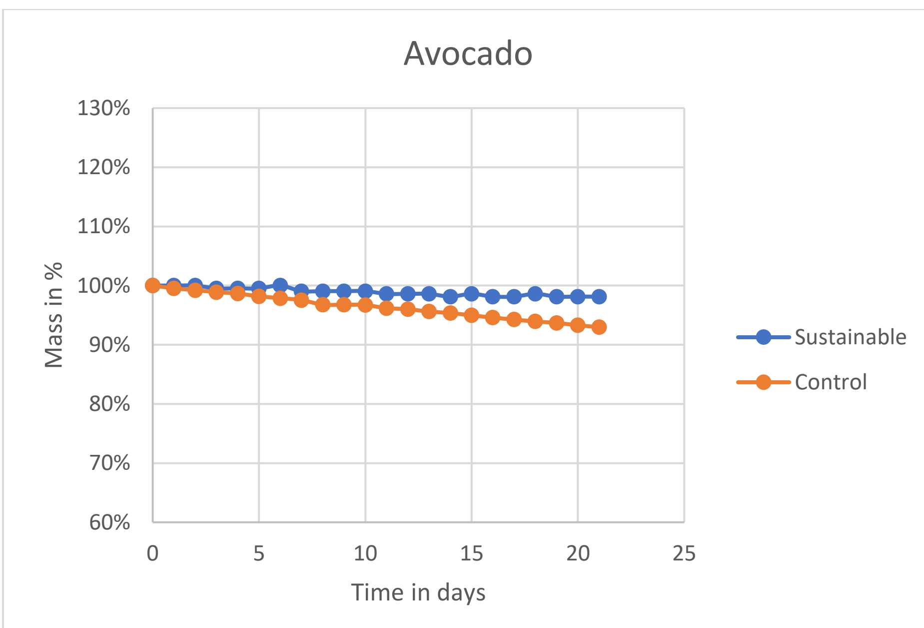


Figure 9.3: Control Fridge vs Sustainable Fridge: Avocado Mass

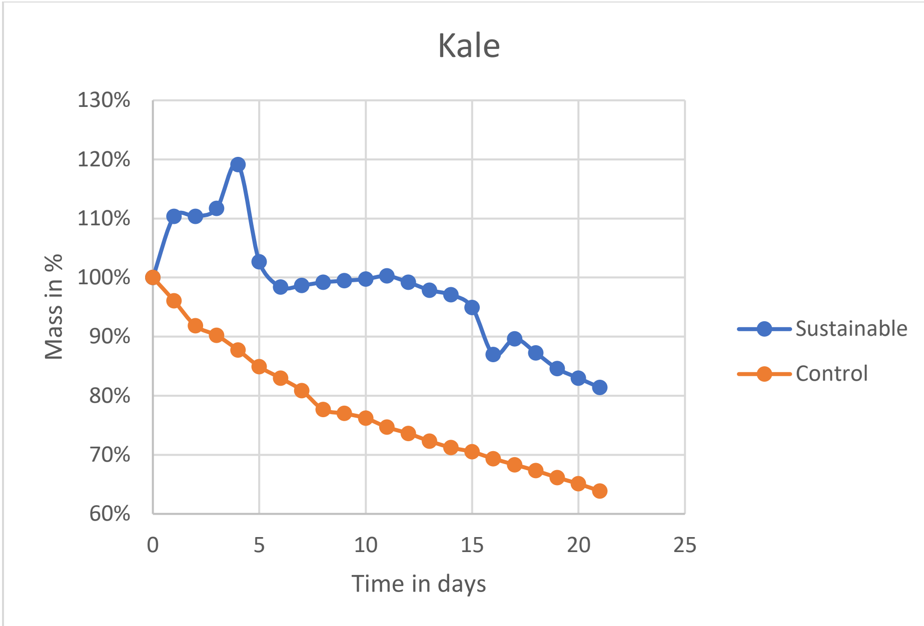


Figure 9.4: Control Fridge vs Sustainable Fridge: Kale Mass

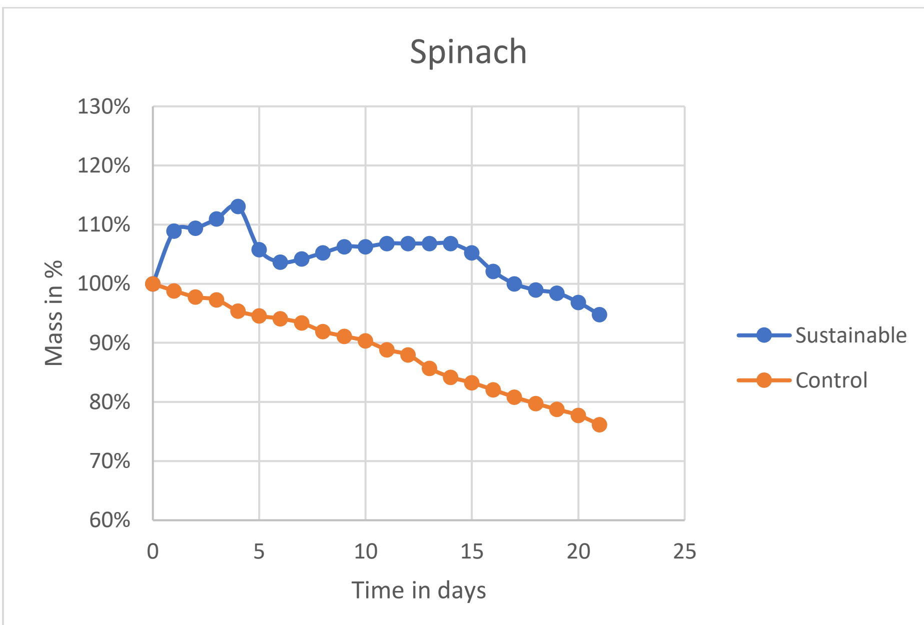


Figure 9.5: Control Fridge vs Sustainable Fridge: Spinach Mass

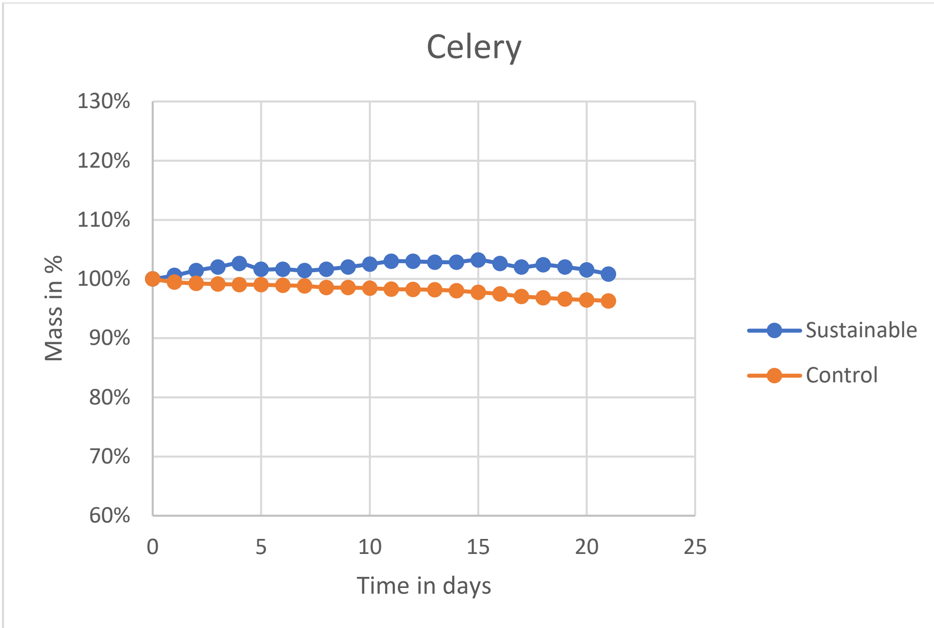


Figure 9.6: Control Fridge vs Sustainable Fridge: Celery Mass

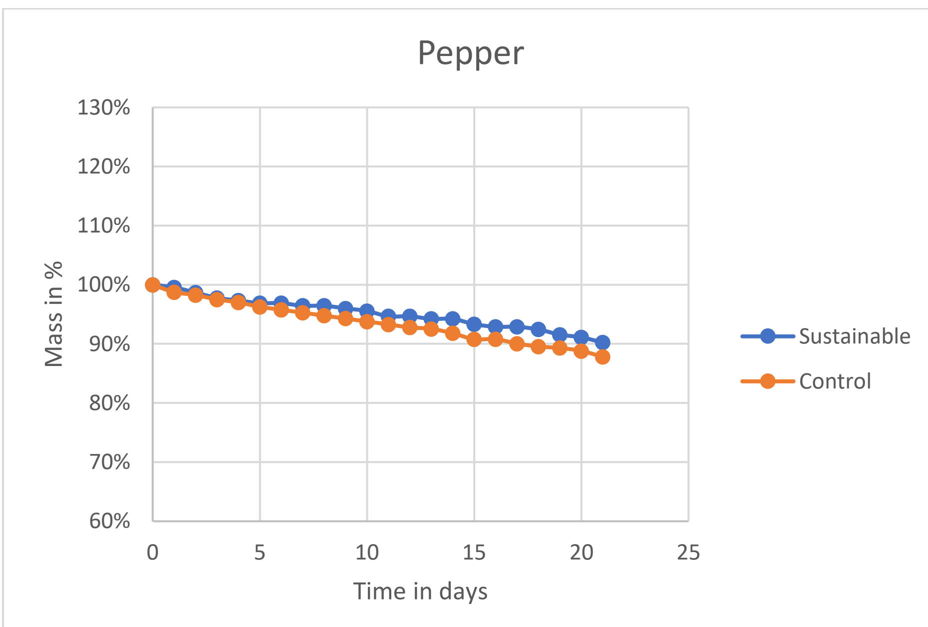


Figure 9.7: Control Fridge vs Sustainable Fridge: Pepper Mass

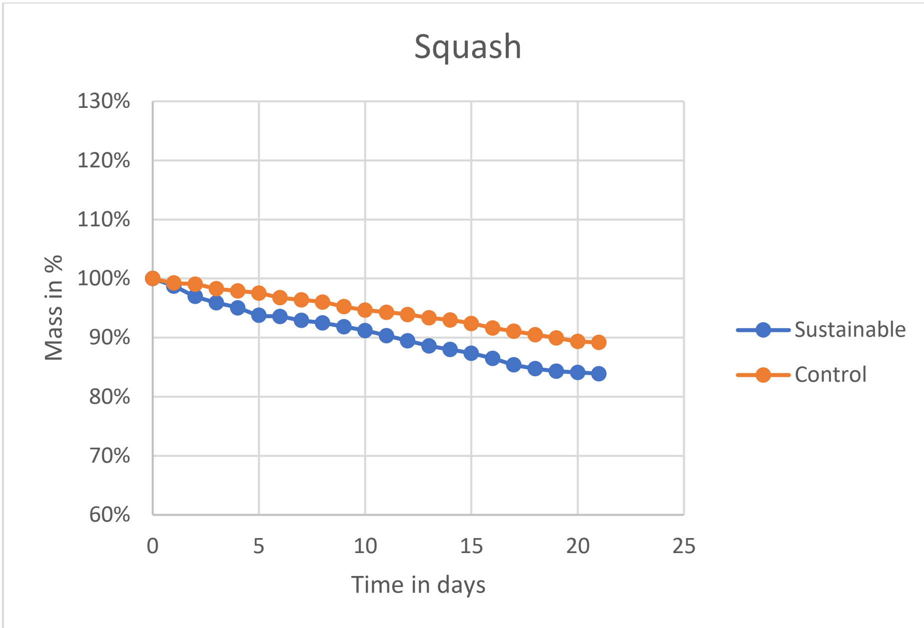


Figure 9.8: Control Fridge vs Sustainable Fridge: Squash Mass

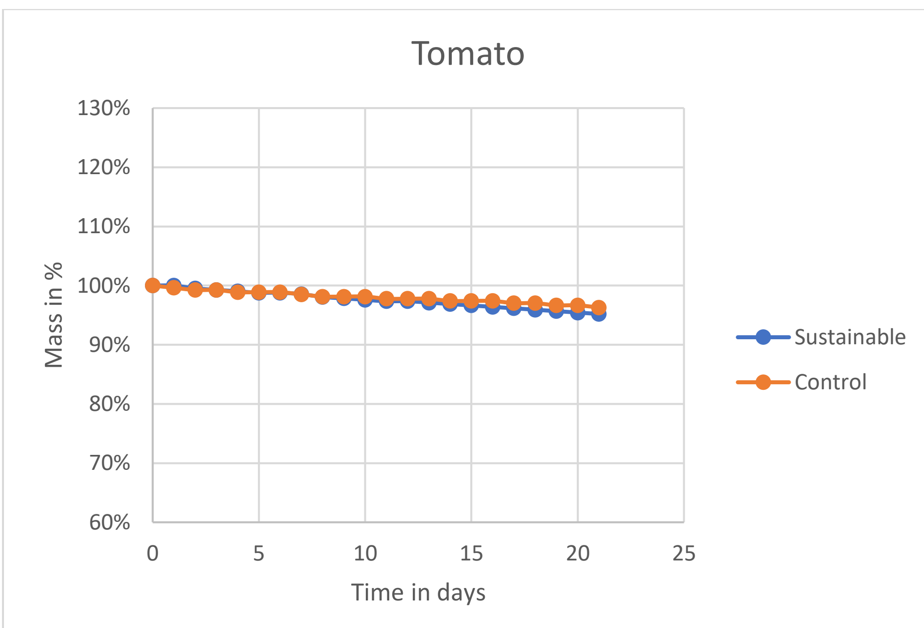


Figure 9.9: Control Fridge vs Sustainable Fridge: Tomato Mass

Change of Mass in % from Day 0 to Day 21 in Fridge

Control Test Pictures:

Top Left is Day 1, Bottom Right is Day 28 (Group 1 & 2)/Day 25 (Group 3 & 4)

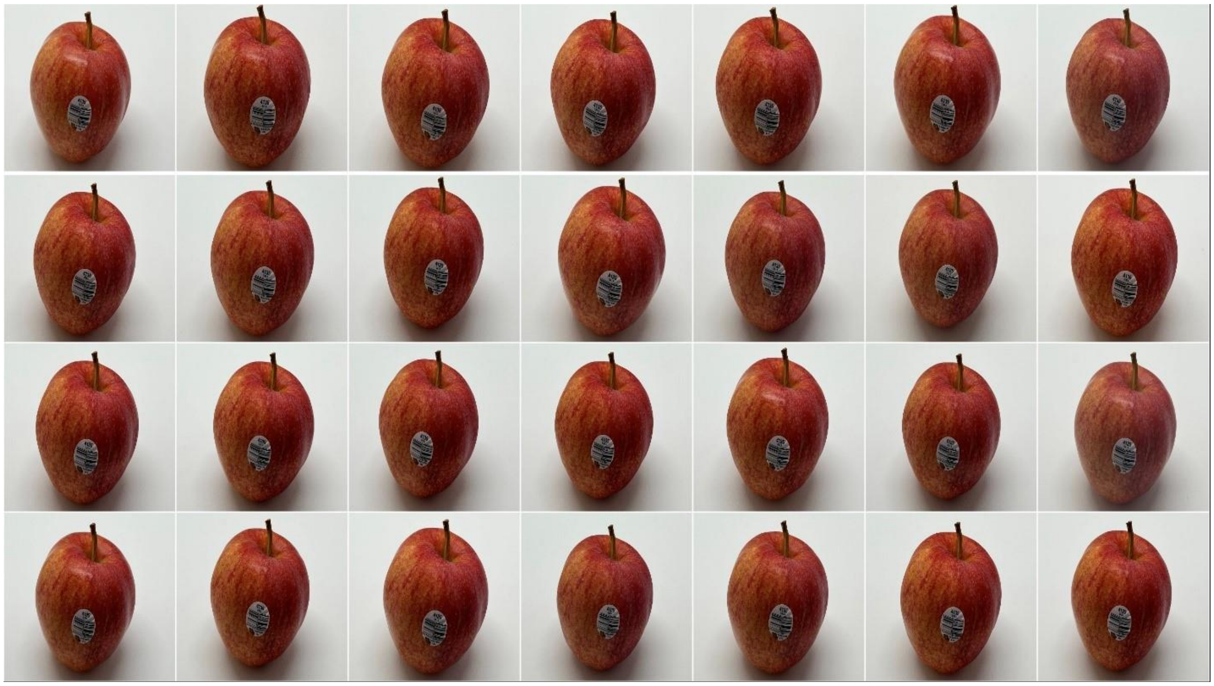


Figure 10.1: Control Fridge Apple Pictures

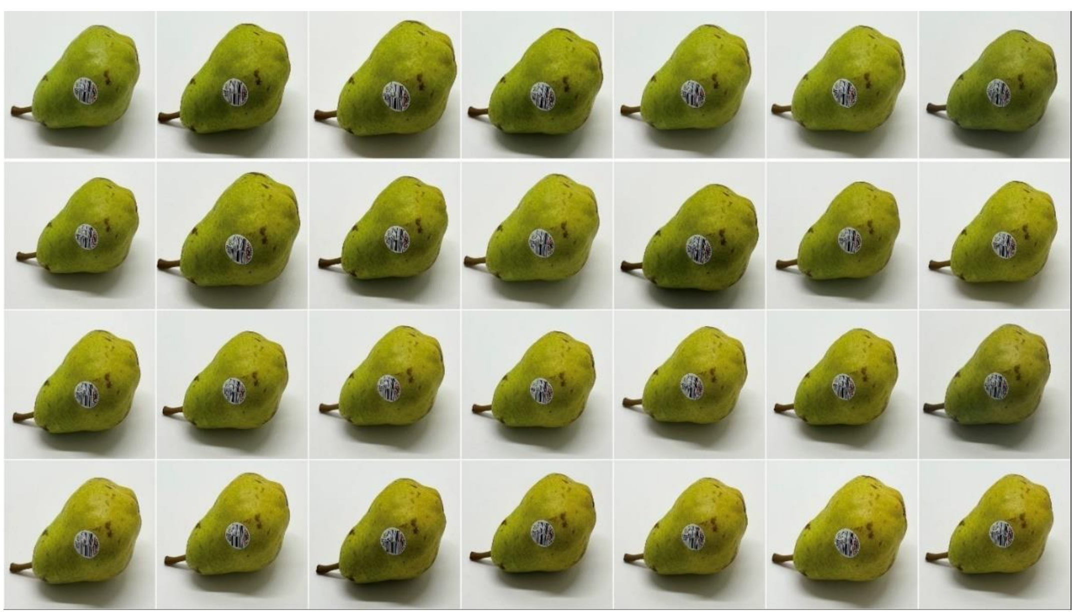


Figure 10.2: Control Fridge Pear Pictures

Figure 10.3: Control Fridge Avocado Pictures

Figure 10.4: Control Fridge Spinach Pictures

Figure 10.5: Control Fridge Kale Pictures

Figure 10.6: Control Fridge Celery Pictures

Figure 10.7: Control Fridge Squash Pictures

Figure 10.8: Control Fridge Bell Pepper Pictures

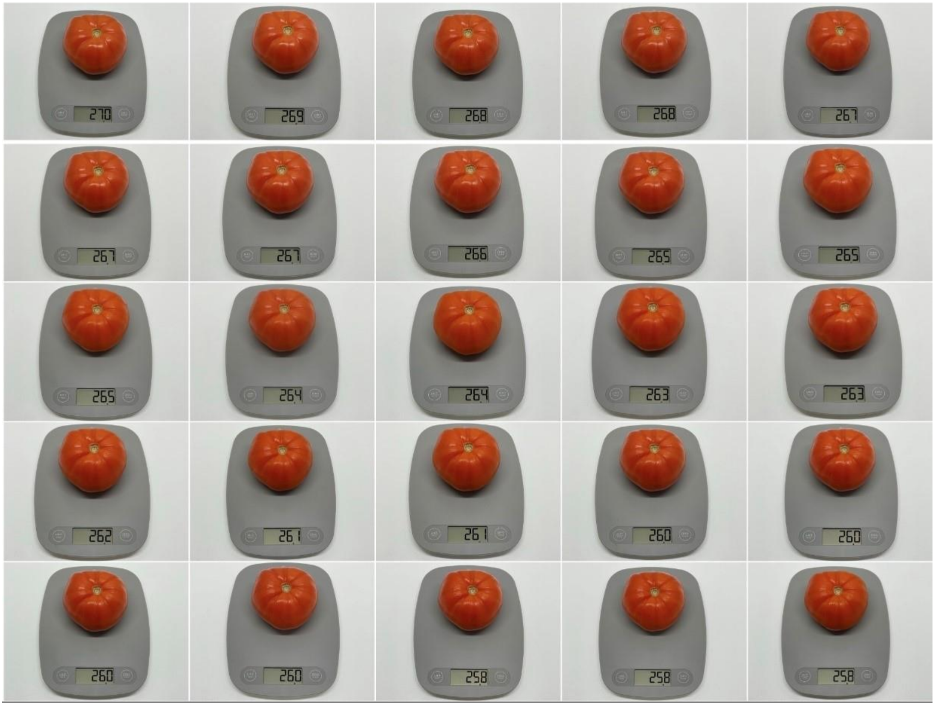


Figure 10.9: Control Fridge Tomato Pictures

Progressive photos taken throughout the experiment for each of the produce

Final Test Pictures:

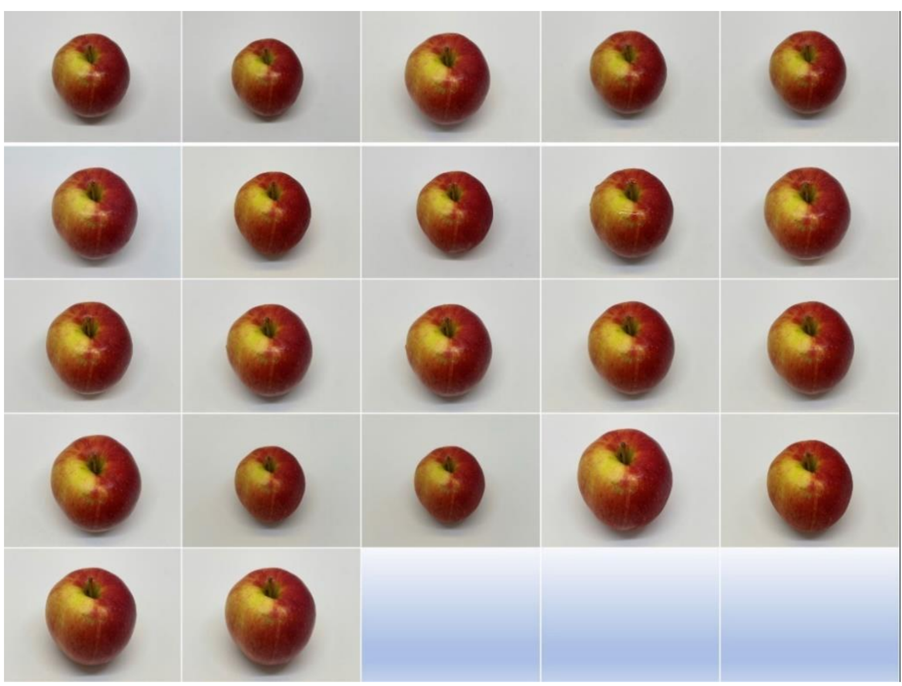


Figure 11.1: Sustainable Fridge Apple Pictures

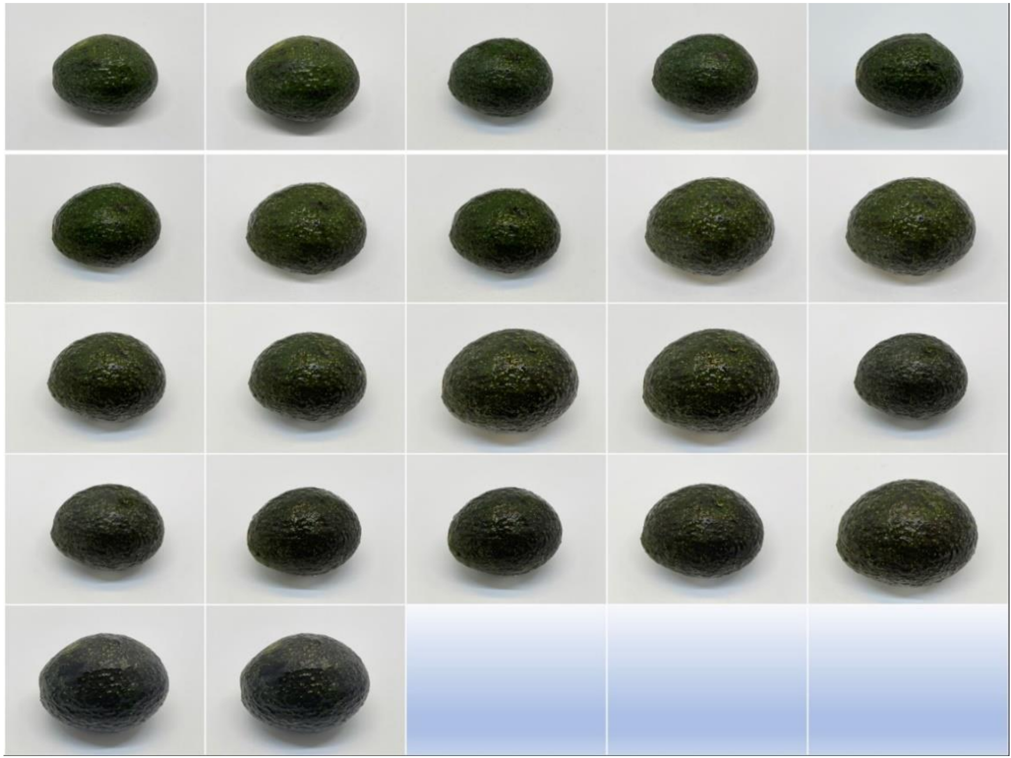


Figure 11.2: Sustainable Fridge Avocado Pictures

Figure 11.3: Sustainable Fridge Celery Pictures

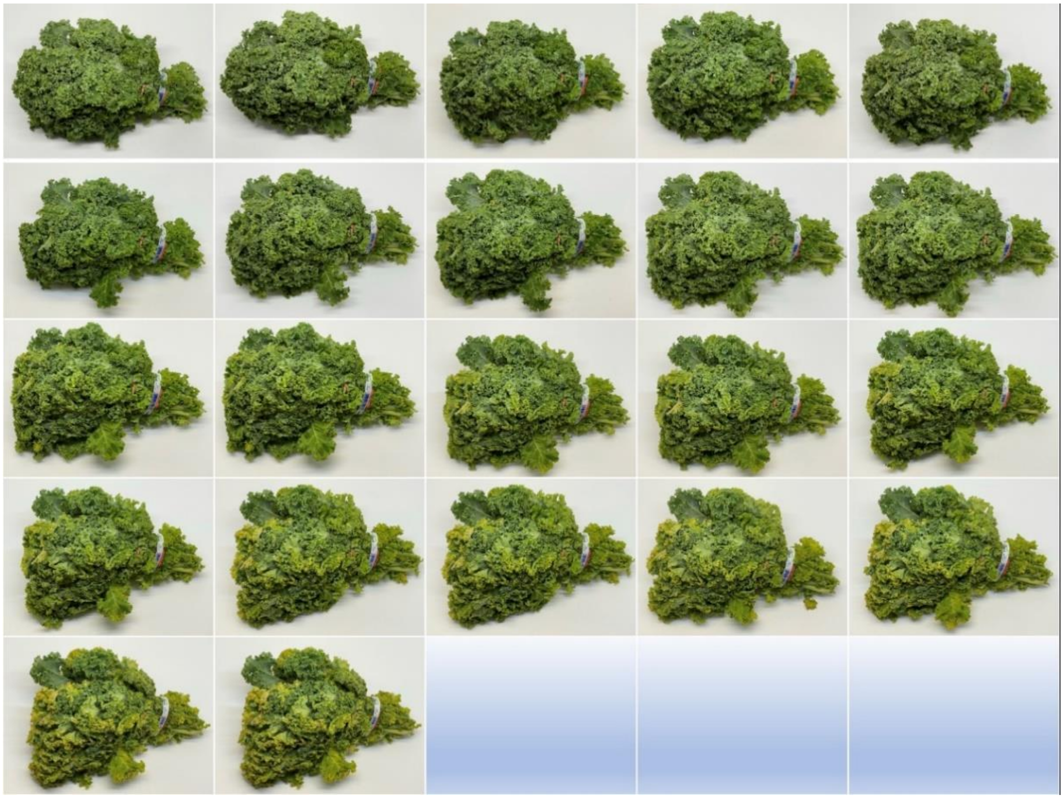


Figure 11.4: Sustainable Fridge Kale Pictures

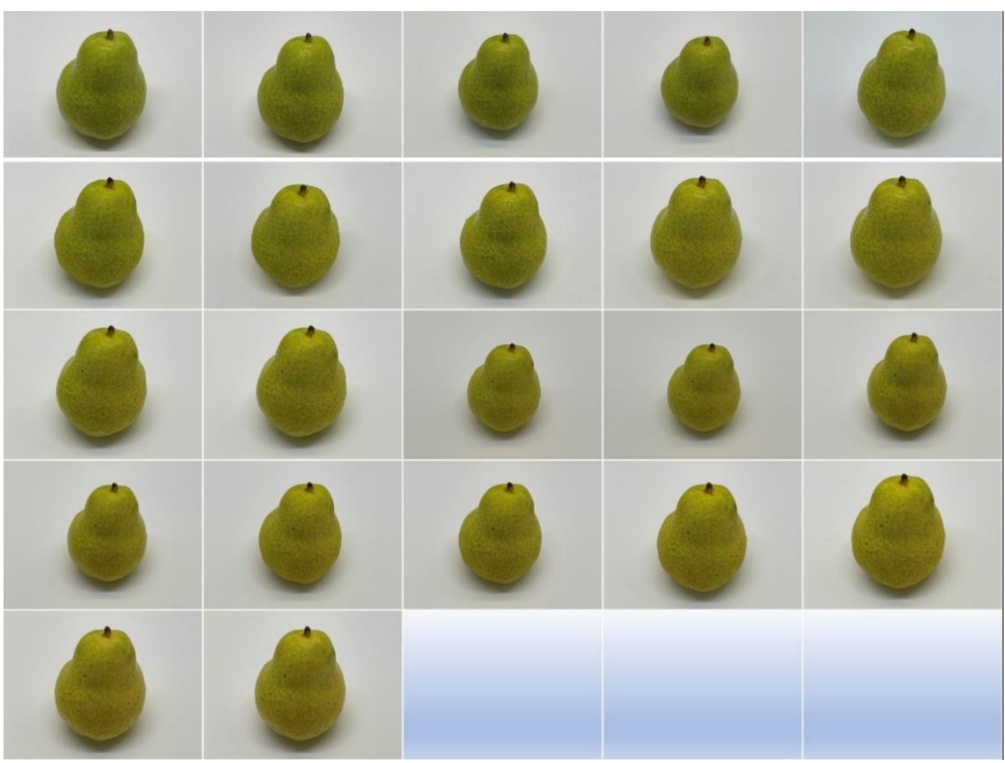


Figure 11.5: Sustainable Fridge Pear Pictures

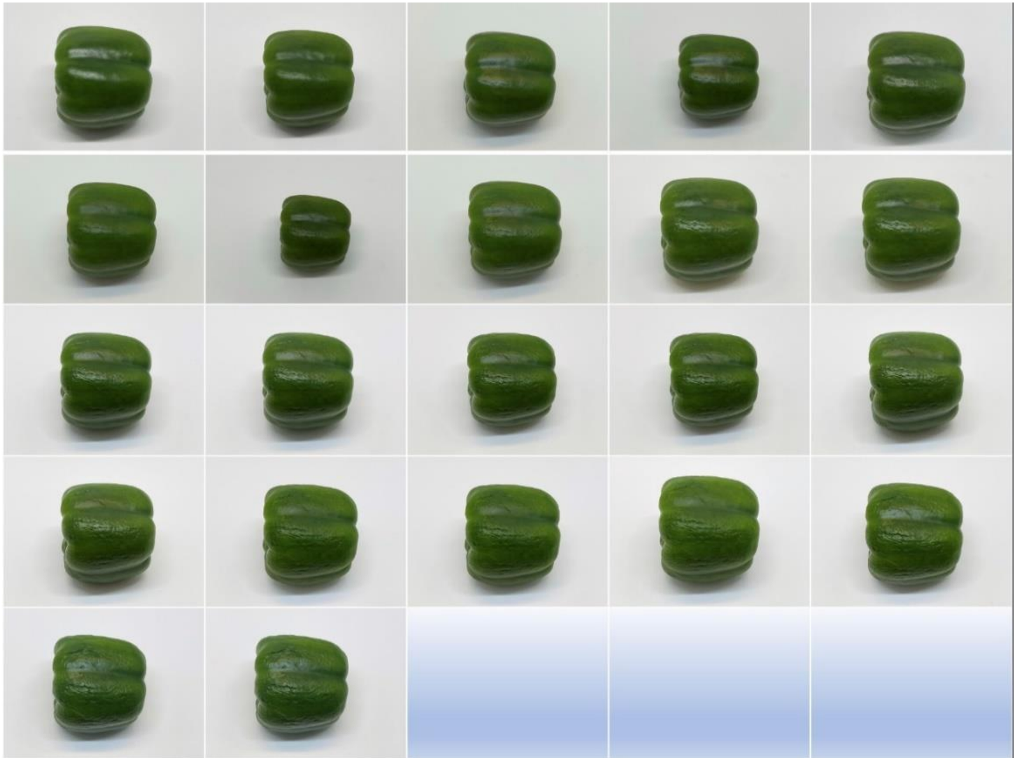


Figure 11.6: Sustainable Fridge Pepper Pictures

Figure 11.7: Sustainable Fridge Spinach Pictures

Figure 11.8: Sustainable Fridge Squash Pictures

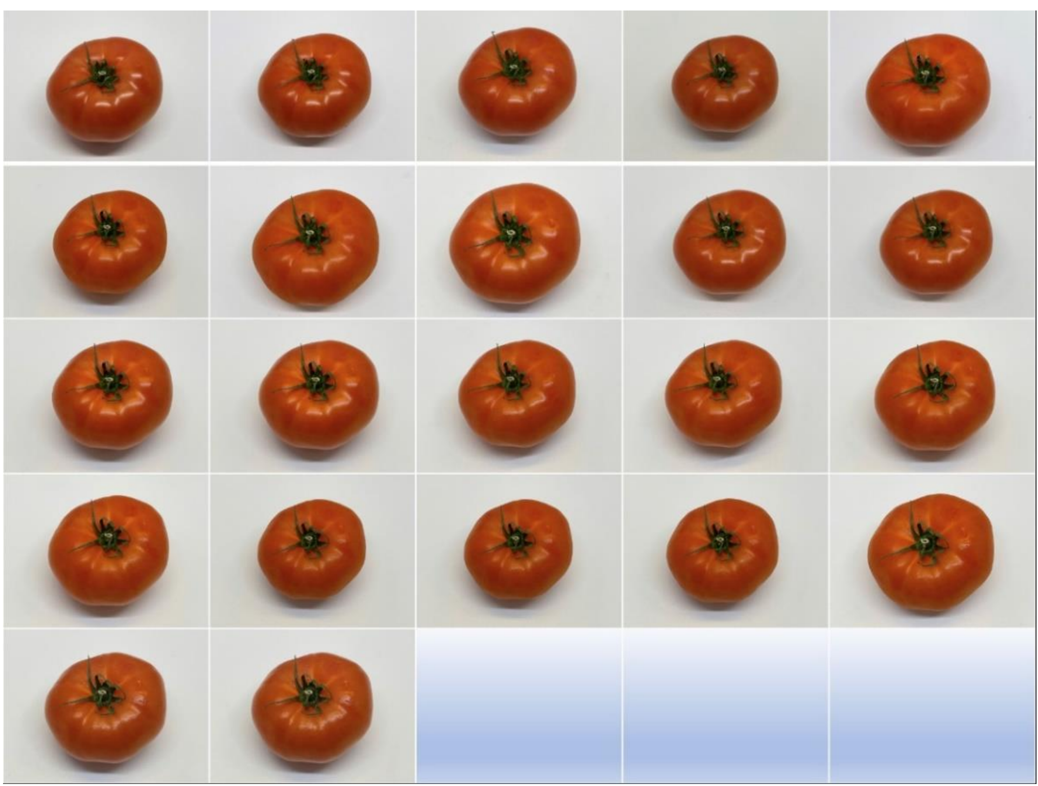


Figure 11.9: Sustainable Fridge Tomato Pictures

Progressive photos taken throughout the experiment for each of the produce

4. Experimentation validation

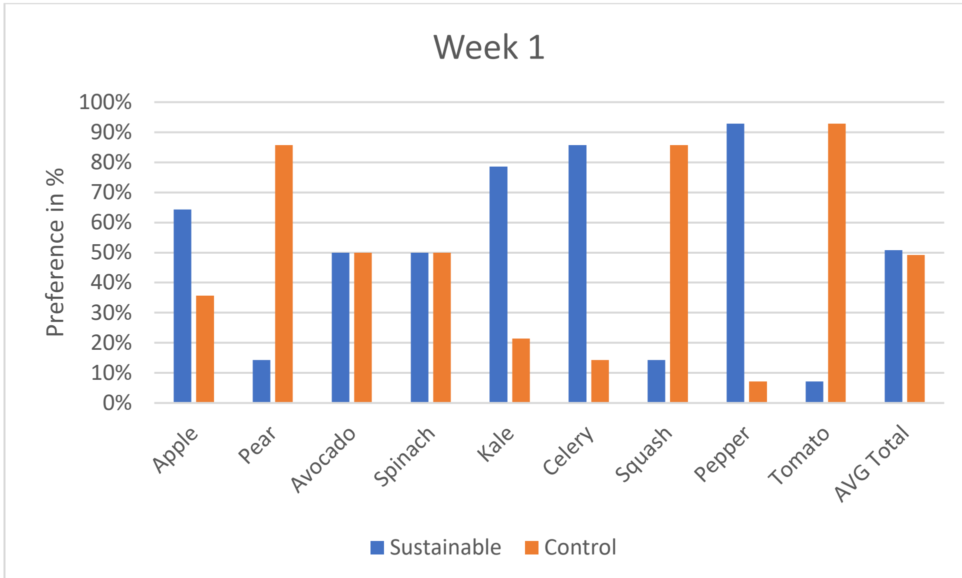


Figure 12.1: Survey Results Week 1

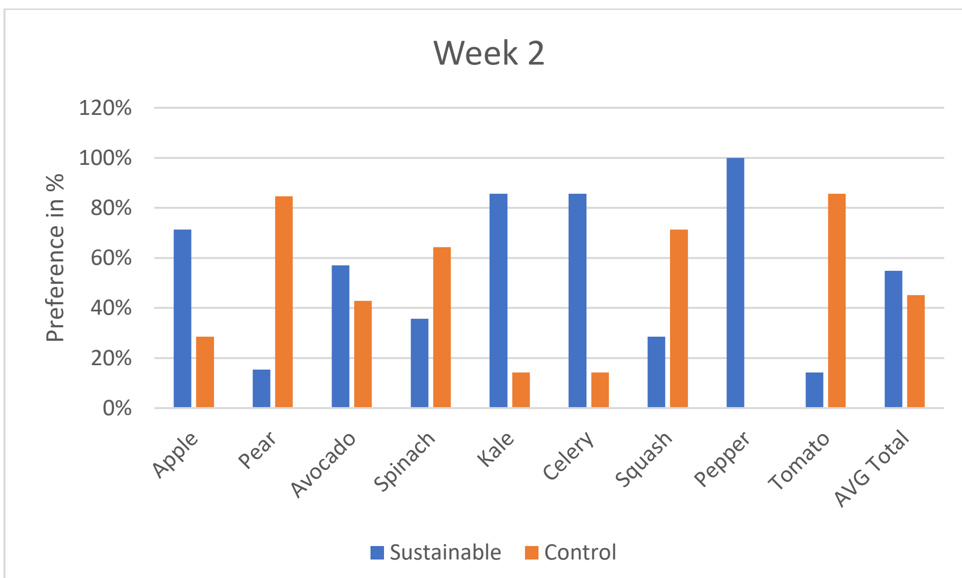


Figure 12.2: Survey Results Week 2

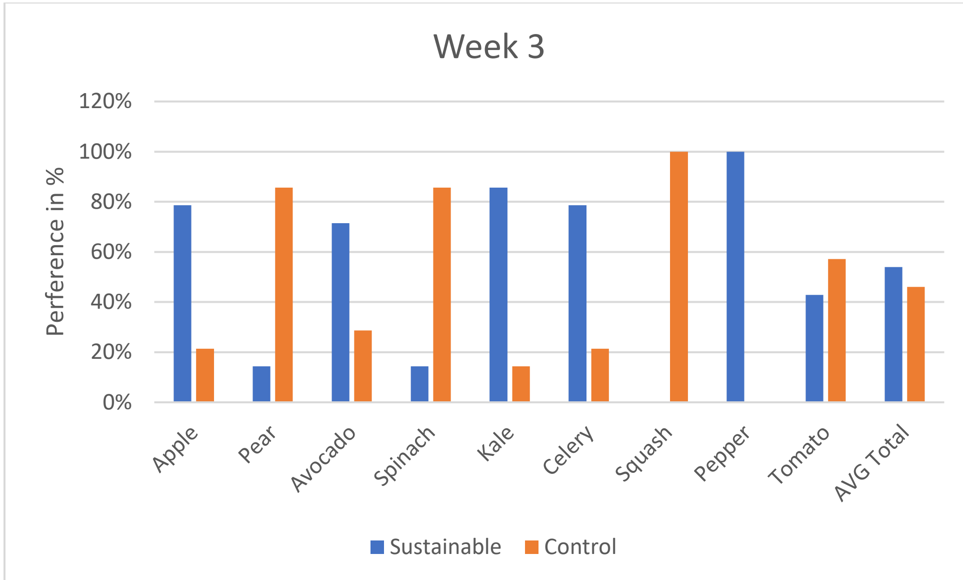


Figure 12.3: Survey Results Week 3

Using the results of the ongoing experimental test a survey was conducted in which volunteers were given the choice between two pictures of various produce at varying instances during the testing periods. Volunteers choose the picture that is most appealing to them. The survey was kept as short as possible to encourage participation. Additionally, the mass measurements taken during testing were plotted with both the control and experimental data on the same graph to better illustrate any differences in performance. Next, power consumption over a 21-day period was compared and the prices compared to fresh produce.

Power Usage Control Test

Fridge	Fridge Run Time (Days:Hours:Minutes)	Total Power Used (kWh)	Uncertainty (kWh)
Control Fridge	21:09:03	35.56	0.005
Sustaintable Fridge	21:06:29	35.09	0.005

Figure 13.1: Power Usage Control Test

Power Costs State

Fridge	Average Retail Energy Price in Virginia ($/kWh)	Total Cost over 21 day span ($)	Cost per Year ($)
Control Fridge	0.0916	3.26	56.61
Sustaintable Fridge	0.0916	3.21	55.87

**Data on $/kWh for State found from eia.gov [3]

Figure 13.2: Power Costs State

Power Costs National

Fridge	Average Retail Energy Price United States ($/kWh)	Total Cost over 21 day span ($)	Cost per Year ($)
Control Fridge	0.1059	3.77	65.45
Sustaintable Fridge	0.1059	3.72	64.59

**Data on $/kWh for State found from eia.gov [3]

Figure 13.3: Power Costs National

The data shown below is the annual cost to run the fridge compared to the average annual cost of fresh fruits and vegetables for homes with a food budget of $3,935 [4].

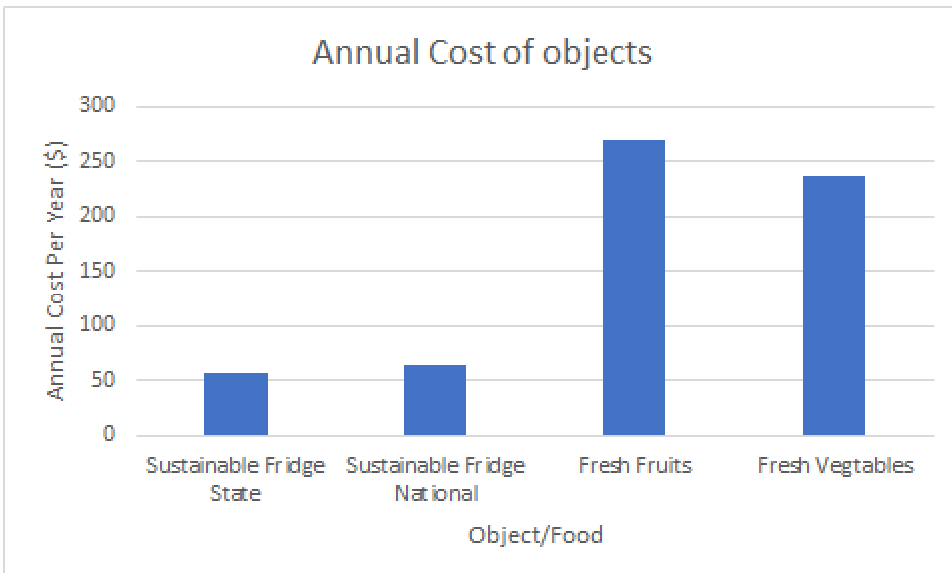


Figure 13.4: Annual Costs of objects

Finally, the next data is for the amount of time it took to change an environment when starting at an appropriate environment level.

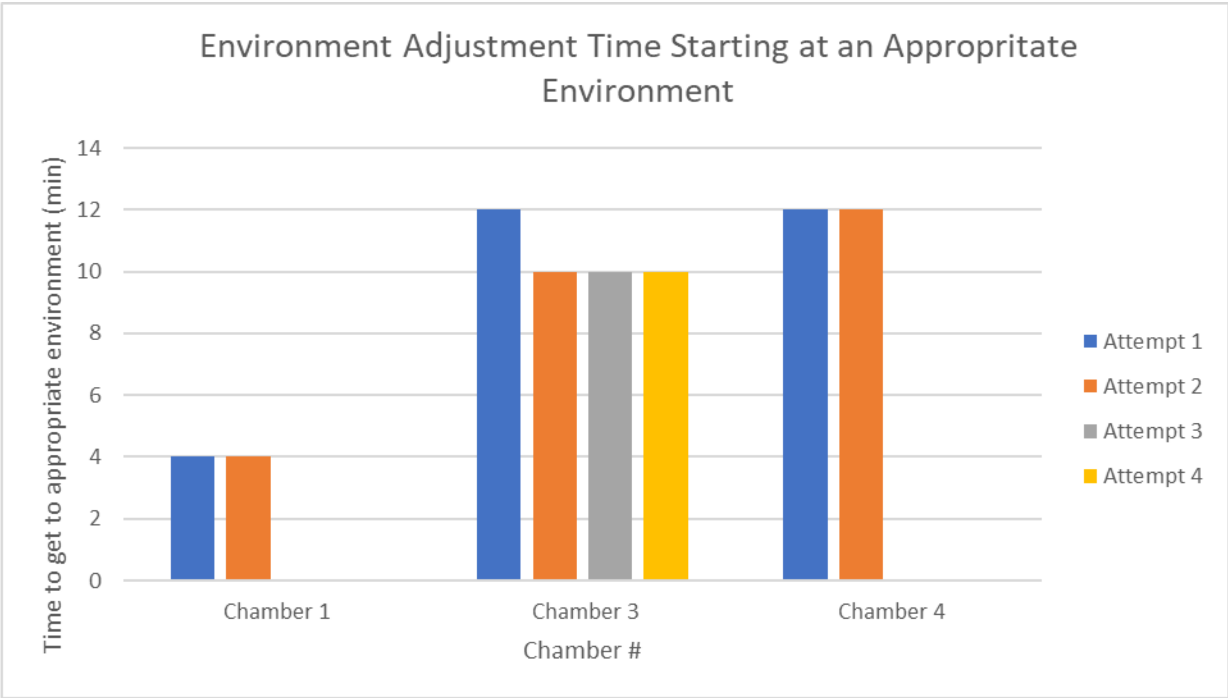


Figure 14: Time it takes environment Change to appropriate environments when starting an acceptable value

4.1 Final Conclusions

Final conclusions were determined with a high emphasis on the collected photos. Due to the humid conditions within the finalized chambers mass comparison was not the ideal comparison since the produce in the experimental chambers were wet at the time of data collection. Photos better illustrate the difference in freshness between the control and the experimental produce.

The survey results seem to reflect the mass data correspondingly. For example, we see a preference from the respondents on the sustainable avocado over the control especially in week

3, where there was also an improvement in the mass of the sustainable over the control. We notice a similar trend over the apple, kale, celery, and pepper. On the other hand, we notice a preference from the respondents on the control squash over the sustainable especially in week 3, which also reflects the lack of improvement of the sustainable squash mass over the control, similarly with the tomato. However, outliers to this comparison between the survey and the data results are also noticed, such as the spinach, where there was a noticeable improvement of the sustainable over the control in terms of change in mass. Whereas the survey respondents preferred the control spinach over the sustainable, similarly with the pear.

We believe that this conflict in our results is due to the small sample size of one produce item per group per test, and the variables of the initial conditions. Moreover, it is important to note that the produce used in both tests may have not begun their testing period at the exact same "age" since it is impossible to know how long each item had already been in commercial storage before it was bought for these tests. For this reason, it is recommended to redo these tests using fresh picked produce either from a home garden or a farmers' market to formulate a more precise conclusion.

Results show that there is little difference when comparing the control fridge and the sustainable fridge's power consumption. This leads the overall cost, both state and national, to be almost the same. The reason we believe that this is a possibility is two reasons. One is due to how the software is designed to set up to only run at most, a solenoid valve, a humidifier, and a temperature sensor when in chamber check mode. This is due to avoiding overdrawing current through the Raspberry Pi. Another reason for this is due to the thermometer of the fridge itself. When reaching the appropriate temperature, the motor of the fridge circulates cold air throughout the fridge and will be off until the fridge is needed to cool again. During this period of the motor off, there will be less power being drawn through the fridge.

When comparing the data, we see that the annual cost to run a Sustainable Fridge both in Virginia and National, it is cheaper to than the average annual cost of produce on a regular basis with a budget of around $3,935 [4]. This means that having the Sustainable Fridge long-term can lead to cheaper costs at the grocery store.

With the time required to adjust the environment we see that Chamber 1 had the least amount of time to adjust its environment, and Chambers 3 and 4 took the longest. With Chamber 4 it had the most consistent data, with it taking about 12 minutes to adjust its environment. Chamber 3 took about an average of 10.5 minutes to adjust its environment. Chamber 1 took around 4 minutes to adjust its environment. What we can possibly infer is that the volume taken by the produce causes a change in how much time it takes to change environment, with chamber 4 having the most of its interior filled up by produce. Chamber 3 had the second most produce inside the chamber, with chamber 1 and chamber 2 having the least amount of space filled up. We believe that the increase of volume has an increase on the time to change the environment when already at an appropriate level.

5. Project Implementation + Problems

5.1 Reason for the project:

The reason for this project was to help produce lasting longer. This would benefit people who must buy produce on a regular basis because it means they would spend less on produce. Another purpose was to have an indirect effect on the user’s eating habits. If the product were able to stay fresh and visually appealing longer than in other fridges, it could lead to people eating healthier.

5.2 Potential Use of the project:

This project could be marketed as a product to be an add-on for fan-blower fridges. This would allow the feature of keeping produce fresh longer to normal fridges. This would, as a result, be a cost-effective option to turn a standard fridge into a sustainable fridge when compared to modern fridges. The purchasers of this will be two types of people, one would be consumers working in the restaurant business needing a cost-effective option to sustain produce. The second group would be people that want to eat more fruit and vegetables. For those that have a more vegetarian-type diet, this would be a way for the consumer to modify their fridge to help sustain fruits and vegetables and save money in the long term.

5.3 Costs:

Approximate Manufacuter Cost			
Item	Quanitity	Cost for 1 ($)	Total Cost ($)
Chambers	4	27.99	111.96
Raspberry Pi	1	35	35
Raspberry Pi Case	1	15.99	15.99
Soilenoid Valves	4	39.99	159.96
Temp/Hum Sensors (6 pack)	1	28.99	28.99
LCD Screen (2 pack)	1	12.99	12.99
Junction Box 1	2	49	98
Junction Box 2	1	9	9
Humidifiers	4	11.99	47.96
Water Jars (6 pack)	1	20	20
8-Channel Relays	2	19.99	39.98
Wire Counduit 1	1	21.99	21.99
Wire Counduit 2	1	16.72	16.72
Push Buttons (5 pack)	1	8.99	8.99
12V 8A Power Supply	1	12.99	12.99
Relay Power Strip	1	57.75	57.75
Brass connectors	4	8.96	35.84
Food Grade Plastic Tubing	1	12.99	12.99
Water Tubing Fitting Connectors	1	11.99	11.99
Water Container	1	19.97	19.97
Zipties (100 pack)	1	4.99	4.99
Total Cost			784.05

Figure 15: Approximate Manufacturer Cost

5.4 Alternatives to the implemented design:

An alternative to the design, if we were unable to obtain a fridge at a fair cost, was to do the same project, but with a freezer. The reasons for not doing this initially were complicated. One was due was the standard shape of the fridge not being able to fit 4 chambers as we wanted to do for the project. This would prove difficult with also having to figure out how to fit the other components such as the solenoid valves and placement of hardware to control the environment. As most modern freezers open from the top, this could lead to damaging the hardware of the project. The other reason was the difficulty of temperature regulation. With a freezer, we would have to think of other methods to keep warmer temperatures in the chambers.

5.5 Maintainability/maintenance:

During the build portion of the project, we had to replace some of the temperature sensors due to overexposure of water based on how the software was made. We then fixed that issue by fixing the code for the software.

5.6 Retirement, replacement, or disposal of the project:

Over time the temperature/humidity sensors' lifetime will expire. The instructions to fix this are in the maintenance manual. Over time the water jars that hold the humidifiers will need to be refilled again when at a low water level.

6. Administrative Section

6.1 Project progress

Our project progressed through two semesters, Fall 2021 and Spring 2022. These two semesters could also be split into three phases each; summing up to six phases total: initial design, obtaining hardware and funding, software design, hardware assembly, control testing, and final testing.

During the beginning of the first semester, we set out on drawing concepts, designs, and methods to approach the topic of our project. We also set to contact our faculty supervisor, Dr. Nathalia Peixoto, through email, requesting assurance for our initial approach, as well as the steps needed to prepare the documents and deliverables. For example, one of the objectives given by the faculty was to build a website for our project. Such objectives and suggestions were all discussed through email, reaching the date of the first meeting. In that initial meeting, we discussed the initial designs and methods as planned by the team and had valuable feedback from the faculty as well.

Feedback given in the first meeting also covered obtaining the parts required to complete our design. From buying the tools needed for testing, to making a parts list of the actual components to be used in our design approach, we had that covered in the second phase of the project. Moreover, our dabble with hardware research made us aware of the need for a sponsor. As such, our faculty suggested that we contact the Patriot Green Fund (PGF) for sponsorship, due to the sustainability nature of our project. Our contact with the PGF ran very smoothly through a series of emails between our Project Manager and the PGF Sustainability Program Manager, Sarah D'Alexander. Towards the end of the first semester, we had successfully obtained sponsorship from the PGF as well as a few of the baseline hardware needed for the initial setup, from the actual fridge to the small components needed to perform our initial testing; with the rest already ordered and expected to ship in the upcoming months.

Towards the beginning of the second semester, we had progress in the software portion. First, we focused on the menu code as a good start for implementing the main functionality of the Fridge. Our code progress was in parallel with the supply of the hardware as orders were arriving. The main file as well as the helpers were complete and integrated by the end of week 10 of the second semester. After the second progress report with the faculty, we received feedback on the user friendliness of the functions that needed improvement. After that, the team formulated multiple proposed redesigns of the functionality and User Interface. (The rest is upcoming and will be included in the final report).

Overlapping the phase of software design was the actual hardware assembly. As mentioned above, the progress of the hardware assembly depended on the arrival dates of the parts and components used. Hardships in the assembly phase as well as the extended ETA for the components forced us to rethink some of them as well as minor redesigns of the assembly itself.

In addition to the hardware and software phases, we performed a control test to measure the performance of regular refrigerators, before comparing to that of our Fridge. Such refrigerators were found in our home, where we performed the control test. More information about this test may be found in section 6.

Upon completion of the hardware and software phases, we were ready to perform the final test on our Fridge. The Fridge was assembled in a lab on campus, provided by our faculty, so the test was performed on-campus as well. More information about this test may be found in section 3 and 4.

6.2 Did you complete all tasks successfully?

The initial plan the team set out to do for the project is completed. Although we were off schedule due to unforeseen events from the project and redesigning certain parts of the project, we were still able to complete it with enough time for proper testing. All hardware and software activities we planned to do over the course of the semester were completed.

6.3 Did you have to change the design & task schedule? What were the changes and why?

During the Fall 2021 semester, we had to make changes in the project. This was because of the unexpected time that was spent on certain parts of the project. This took longer than expected due to the limited space we had and how we had to manage the space with the chambers with also the other components going in there as well. Another change involved the misters for each chamber. This took the longest in the project due to the misters having a high-power consumption, large mist rate and, the container needing to take a lot of volume. We went through several different versions before we were able to select an excellent choice for the mister. On the Software side, we had to extend some portions of the code to other parts due to wanting to meet the goals we set out to do, we had to take more time in troubleshooting than we initially thought.

6.4 Any extra (not-planned) activities you had to carry out?

At the beginning of the project back in Fall 2021, we initially thought we would be given a fridge to work on. Unfortunately, we had to go out and get a fridge based on our needs for the design of the project. We initially planned to obtain a fridge through Facebook-marketplace because of the cost-effective options of fridges available that were valued at a price in our price range. The issue was that the people offering the fridge online were giving out fridges that were not up to our standards. Fortunately, we were able to obtain a fridge that met our standards.

During the build phase of our project, we ran into issues of some specific parts not being able to arrive on time due to multiple reasons. One was of the manufacturer did not make those parts due to supply-chain issues. Another reason was due to current world issues that we did not factor in causing an issue on the supplies coming in on time. The way we fixed this was with the help of our sponsor, the Patriot Green Fund, and the help of a member of the organization, Sarah D'alexander, we were able to find alternative parts or an alternative supplier in a reasonable time.

Another activity we had to carry out was unexpected was finding the appropriate mister. The mister we initially had was not going to work with our design for multiple reasons. I was consuming a large amount of power, that being 24W of power. Its rates of mist would be so great it would lead to the user needing to refill the water more. This also led to a need for a water supply big enough to accommodate the rate of mist, leading to an issue of volume in the chambers. The other issue of the mister was it would make a lot of heat from the mist, melting through containers if close to the point where the mist comes out. After several redesigns of the mister, the team went with a mister that only needed 5W of power, had a low mist rate, and sprayed a cool mist.

6.5 Funds spent

Approximate Total Cost of Products

Item	Cost	Paid by
Fridge	$240	Team
Parts for Fridge	$900	Patriot Green Fund
Parts for Fridge	$150	Team
Total	$1,290	

Figure 16: Approximate Cost of Products

According to Table 8.2, the approximate total cost of the project was around $1290. This includes products we initially got for the project but were not able to be used for the design and could not be returned due to the seller's return policy. Thanks to the help of the Patriot Green Fund, we were able to get the supplies we needed for the project.

6.6 Man-hours devoted

Man-Hours (492 & 493)

Member	Hours
Cameron	260
Nestor	260
Matthew	260
Ahmed	260
Total	1040

Figure 17: Man-Hours of Total Project

7. Lessons learned

7.1 Additional Knowledge and Skills Learned

During the last year, we acquired skills that will be put to effective use in our professional lives. No other class that we have taken for our degree is tailored to what we will soon experience as we obtain employment.

Beginning with ECE 492 we learned how to carefully write a proposal. For the whole semester we analyzed the project inside and out, which is something that is suitable for large and expensive projects. This first step helps avoid costly mistakes and save labor, so it is easy to see why it is standard for engineering projects.

Even with careful planning there were unforeseen issues that sent us back to the hardware store. Because the air hoses kinked easily, we had to replace some of the straight connectors with 90-degree connectors. Also, to add stability we needed to buy Velcro strips to secure the coolers in place. This was a lesson in planning.

7.2 Teaming Experience

From the beginning we divided up our project into separate tasks. Everyone had their own work to do, and we all worked hard to complete the project on time. Due to the complex nature of our design, parts of tasks required team effort. In these cases, the entire team came together to find a working solution. The misters especially required such coordination.

Together we went through five designs for the misters. We all did research on several types of misters and each time we tried a fresh style of mister we had to produce a different container. The first mister that we worked with generated a lot of heat and needed to be at a certain depth underwater. We all debated on which container to use. At first, we used a small shallow piece of Tupperware with a Snap-On lid but that did not work because the top of the container was too close to the heat and melted. There were also several other attempts where we used taller containers and had trouble with water splashing out of the container. Because of this and other problems with excess heat and power consumption, we decided to move on to another style of mister.

While the misters required the most team effort, there were not any sections of this project that did not require a group effort. Everything we did was carefully planned and discussed.

8. References

[1] M. Hawkins, "LCD test script using 12c backpack (lcd_i2c [python])," GitHub, 2015. [Online]. Available: https://github.com/gpul- labs/GPUL_RFID/blob/master/LCDScreen/lcd_i2c.py

[2] "Raspberry Pi Tutorial: How to Use the DHT-22," instructables circuits, October 7, 2018 [Online]. Available: https://www.instructables.com/Raspberry-Pi-Tutorial-How-to-Use-the-DHT-22/

[3] "State Electricity Profiles." eia.gov. https://www.eia.gov/electricity/state/

[4] S. Price "Average Household Cost of View." ValuePenguin, https://www.valuepenguin.com/how-much-we-spend-food (accessed May 7, 2022)

Appendix A- Proposal (ECE 492)

(1) Cover Page

ECE-492 Team #11

Sustainable Fridge Proposal

Cameron Flores, Matthew Philips, Nestor Arellano, Michael Harm, Ahmed Jaber, & Zachary Solini

2) Executive Summary

The project goal is to create a new type of fridge that will allow fruits and vegetables to last far longer than what they currently last do to the current fridges on the market. The fridge will be able to do this with new internal Crisper drawers using technology to control the temperature and humidity in the drawers, thus putting the internal environment of the crispers at the correct level that the fruit or vegetable likes. These new crispers will have temperature and humidity sensors to keep track of the internal levels, small mist machine to create humidity, vents and fans leading to inside of fridge for cooling and removing humidity, and a Peltier module to warm the crispers insides up. Finally, throughout all this will be a processing module controlling these devices and a display showing data about the crispers along with a user input for selection of what type of fruit or vegetable is being put where.

The benefits and deliverables to this new fridge model will be fruits and vegetables will last longer, thus allowing the modern household to not have to go to the grocery store as much. This also saves on money in the long run and helps the environment. The project team will deliver to the project lead the new Fridge, project computer code that will be used in the processing module, and design diagrams showing how everything is hooked up. All these documents will in essence allow anyone to produce the team’s new fridge model.

3) Problem Statement

Motivation:

Most families and single person homes face a crisis in the form of not having enough nutritional fruits and vegetables in the fridge. A modern family normally has to go to the store around twice a week since the normal sized fridge crisper drawers are too small to have a large number of fruits and vegetables in them and more importantly the fruits and vegetables stored in them regrettably turn bad within days. Therefore, we have come up with a new type of crisper drawer that will keep fruits and vegetables fresh longer and more of them in the drawer. In this way families will have the nutritional fruits and vegetables they need for longer and they will not have to go to the store as often.

Identification of Need:

- To Design 2-4 large _ by _ Crisper Drawers, and 2-4 small _ by _ Crisper Drawers that will keep Fruits and Vegetables Fresh longer.
- Each drawer will be insulated using rugged cooler types of technology. Current drawers are not insulated.
- Each drawer will have the ability to affect the temperature and humidity inside. Current drawer types can allow minimal control of humidity and no control over temperature.
- An LCD display or another type of display will allow user to interact with the inside of each drawer controlling the temperature and humidity from preset measurements where the user only has to enter what type of fruit or vegetable inside the drawer. Current fridges have no interaction and control, but they do have an LCD display showing current temperature throughout the fridge.
- Drawers must keep track of temperature and humidity and compare those it to preset measurements that will determine if the temperature or humidity needs to be increased or decreased which the system will automatically do without user input.
- New fridges on the market with humidity control are expensive. Ours will be considerably less in cost.

Market/Application Review:

The new designed crisper drawers will be a major enhancement over current models. These current models only allow very small adjustments in humidity and even then, it is only in the expensive fridges that have this control; such fridges like the Samsung RF28R7351SR, or the Viking VCSB5483SS have a cost of $3000-$12,000. This project will at least match the performance of current products while keeping the design cost low. The chambers in our design will act as a replacement for standard crisper drawers. It will showcase how technology in a fridge can be utilized to extend food life and indirectly influence better eating habits.

4) Approach

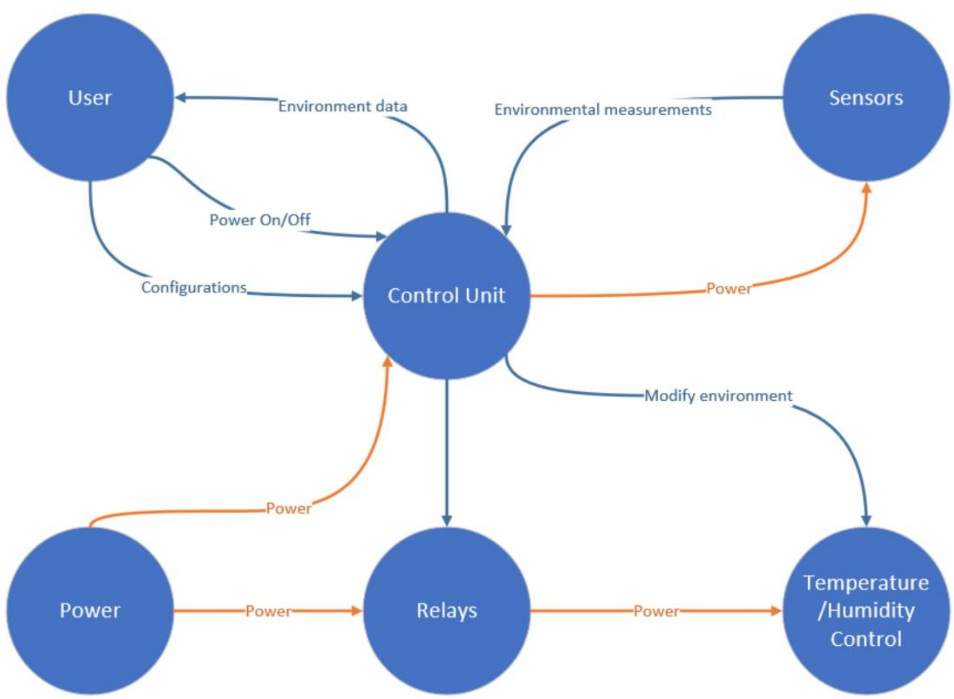
Problem Analysis

Introduction

The objective of this project is to reduce food waste, primarily produce discarded due to decay. Additionally, it will promote healthier diets by keeping vegetables fresh for longer periods of time making them more accessible to the user.

User interface design

The user interface will constantly display a splash screen with the current produce group, temperature, and humidity in each chamber. Additionally, the user will be given the option to reset individual chamber's presets for specific vegetable/fruit groups which involves adjusting humidity and temperature requirements.



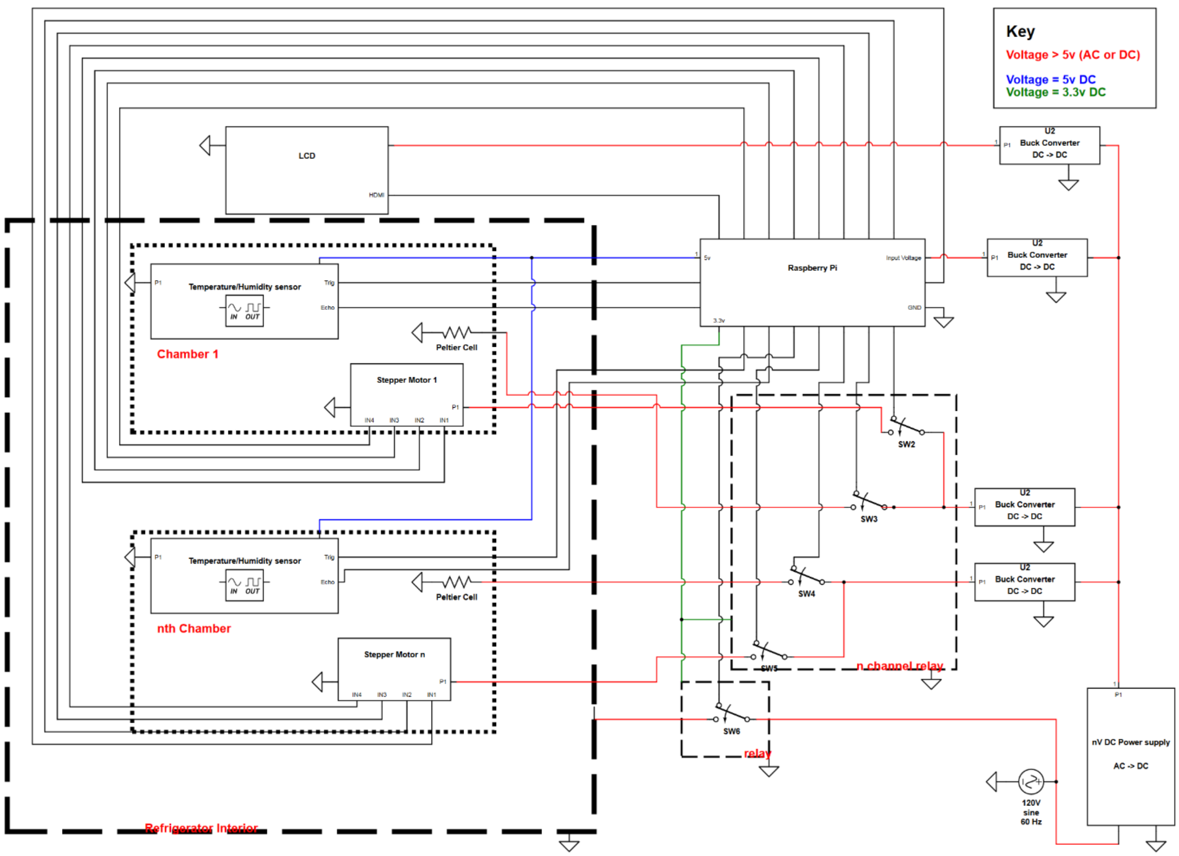
Usage Scenarios

The most common usage scenario will involve a user arriving with fresh produce. Firstly, the user will observe the splash screen to check for available chambers and what vegetable groups are currently within the refrigerator. Next the user will sperate and produce that can go into the existing chambers. After that the user will select from the display the options to configure the next available chamber, the system will display presets with descriptions of the produce that is acceptable to store in those conditions. Lastly the user will place all the produce in its corresponding chamber. The system will display updated chamber assignments and temperature/humidity readings.

Prototype

Initial prototype, with n-produce chambers available. The value 'n' will depend on the volumetric capacity of the refrigerator obtained. The system presets will be derived using the published data from the university of Maine [6]. There will be three main temperature settings as observed in the research: cold (0 - 2.2) °C, cool (4.5 - 12.8) °C, and warm (12.8 - 15.5) °C [6]. Lastly, the humidity data indicates that 95% humidity will suffice for most of the produce [6]. This value can be adjusted, but initial prototyping should leave it constant to reduce complexity.

**this design will be updated to use all available space for produce chambers as instructed by the customer.



<u>Design components and specifications</u>

Stepper motor: 28BYJ48-12-300-01

Operational voltage range: 5 – 12 V

Pull-in torque at 500pps: ≥ 0.0294 N-m

Number of phases: 2

Resistance per phase: 300 Ω

Motor driver Chip: ULN2003A

Max input Voltage: 30 V

Max output Voltage: 50 V

Max continuous collector current: 500 mA

Max continuous base current: 25 mA

Operating ambient temperature range: - 40 to 85 °C

Operating voltage: 5V

Temperature/Humidity sensor: DHT11

Operational voltage range I/O: 3 – 5 V

Max current: 2.5 mA

Operational temperature range: 0 - 50°C ± 2°C accuracy

Operational humidity range: 20-80% ± 5% accuracy

Sampling rate: 1Hz

Peltier cells: TES1-12703 (30 x 30mm)

Operational Temperature: -156°C to 80°C

Max current: 3.5 A

Max voltage: 15.4 V

** All data taken from each device's corresponding data sheet.

Decision matrices

Cooling methods rated (1 – 3)

	Cost	Member experience	Complexity estimate	Score
Refrigeration ducts	2	1	3	6
Peltier cells (for cooling)	2	1	1	4

Sensing methods rated (1 – 3)

	Member experience	accuracy	Humidity measurement capable	cost	Complexity estimate	Score
DHT11	3	1	3	3	3	13
LM35	1	3	N/A	3	2	9

Humidity control rated (1 – 3)

	Cost	Size of equipment	Maintenance	Score
Misting system	1	1	1	3
Peltier cells (heat to produce steam)	3	3	3	9

Processing module

	Cost	Member experience	Complexity estimate	Score
Raspberry Pi 3	3	3	2	8
BeagleBone Black	2	1	2	5

The decision matrix above was how we determined what type of equipment would be best used. These were rated on a scale from 1 to 3. The highest score of the method would be the method that we would use on the project going forth.

Work Products

The design will require extensive programming, since it will be primarily controlled by a Raspberry Pi 3 the main program will make use of extensive open-source code. The code referenced will give the Raspberry Pi control over multiple peripherals including stepper motors, temperature/humidity sensors and relays.

Integration with existing processes

The completed project will not need to integrate with other existing systems to do its job. Making the current system information accessible on mobile devices will be on the top of our wish list for this project.

Feedback from customer

The customer was satisfied with the initial prototype design, they did stress that they would rather have a refrigerator dedicated only to produce instead of a dual-purpose system.

Additionally, they proposed possibly testing vitamin A and other nutritional values in produce after prolonged storage.

Approach

To meet our project requirements the team has proposed the following approach.

- To keep similar produce groupings at optimum storage temperature we have sectioned off the main storage area of a refrigerator into individual temperature and humidity-controlled chambers.
- To measure temperature and humidity, sensors are housed within each chamber.
- To keep temperature and humidity levels at optimum refrigeration ducts and Peltier modules are installed in each chamber. If that design fails, we will switch to modifying a heat pump that will remove heat from the chamber in order to cool down the food.
- To power all added components to the refrigerator buck converters are added in parallel to the refrigerator’s main power input. Relays are added to enable the control of power delivery.
- To control power delivery, monitor temperature/humidity, and adjust temperature/humidity values a Raspberry Pi 3 is added to the refrigerator. If the Raspberry Pi 3 is unable to do these tasks, we will use another processing module to run our commands, such as BeagleBone, MSP430, etc.
- To give the user feedback and control over the system conditions, an interactive display is installed.

- To test the efficacy of the system a control refrigerator will be set up with identical produce stored inside.

Project requirements specifications

Mission requirements

The system shall decrease refrigerated produce lost to decay and encourage healthier diets.

Operational Requirements:

Input output requirements:

- The device shall take input from the temperature and humidity sensors
- The device will accept user input to adjust temperature within chambers
- The device shall use stepper/servo motors to open selected chamber for ventilation
- The device shall display measurement readouts to user
- The device should have an lcd screen to display measurement readouts
- To maintain a constant environment, the device should report measurements to the cloud every 5 to 10 minutes

Functional requirements:

- The device shall measure temperature and humidity
- The device shall select the appropriate chamber for ventilation
- The device should report measurements to the user through the cloud every 5 to 10 minutes
- The device should detect errors and report to the user

Technology and system-wide requirements:

- The device shall use a Raspberry Pi because it meets our project requirements and high number of GPIO ports allow us to control multiple components at the same time
- The device should be low-cost
- The device will display data on an LCD
- The device should have internet access and communicate data to the cloud in order to report sensor readings and system status to the user

Alternative approaches

Our proposed solution involves a dual approach of informing and educating the user and temperature and humidity monitoring and control. Having both of these allows for more customization to the user's needs. Alternative approaches could focus solely on one of these components and disregard the other. For example, focusing mainly on educating users on how to best store food, or leaving out the end-user aspect and taking a broader optimization approach. This might involve keeping the entire fridge at temperatures and humidity levels that are optimal only for common fruits and vegetables. Other approaches might focus on Ultraviolet light, which has been used successfully in many areas of the food industry, especially dairy. Having UV light as a potential deterrent for bacteria and mold growth could possibly extend the shelf life of some refrigerated items. Finally, another approach might go into depth on controlled-atmosphere storage, keeping oxygen and ethylene gases in low concentrations to prevent microbial growth.

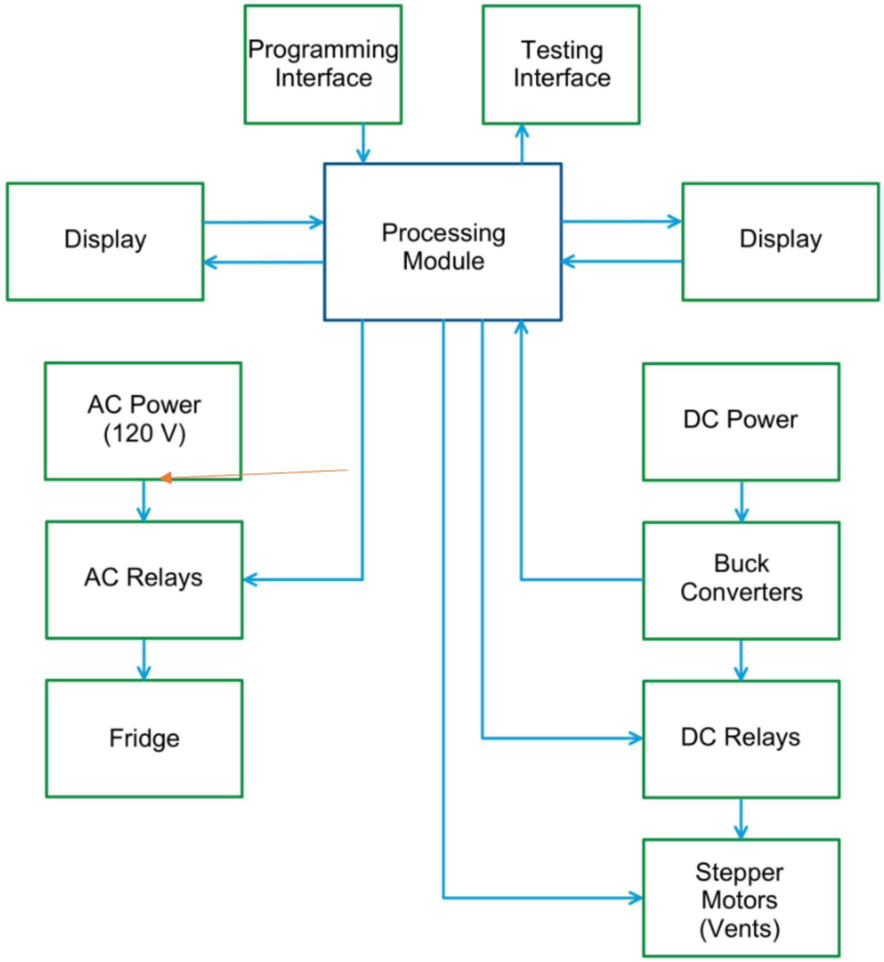
Introduction to background knowledge/phenomenology supporting the project

The more we learn about nutrition and health, the more we realize how important it is to eat a wide variety of fruits and vegetables. In a 2015 report, the CDC stated that fewer than 10% of Americans are eating enough fruits and vegetables [1]. However, in the United States nearly half of all produce is wasted, mostly at the retail and consumer level [2]. In addition, the United States Department of Agriculture (USDA), states that most adults need to eat between 1.5 - 2 cups of fruit and 2 - 4 cups of vegetables every day [3]. The need for our project becomes very clear when you combine all of these with recommendations to make fewer trips out of the house amid the Covid-19 pandemic and the fact that most refrigerators on the market have very limited vegetable storage space. Building a refrigerator designed to store more produce and optimized to keep it fresh addresses all of these problems. When consumers can store enough produce and trust that they have enough time before it goes bad, they can take fewer trips to grocery stores, have enough vegetables on hand to meet USDA recommendations, and waste less food.

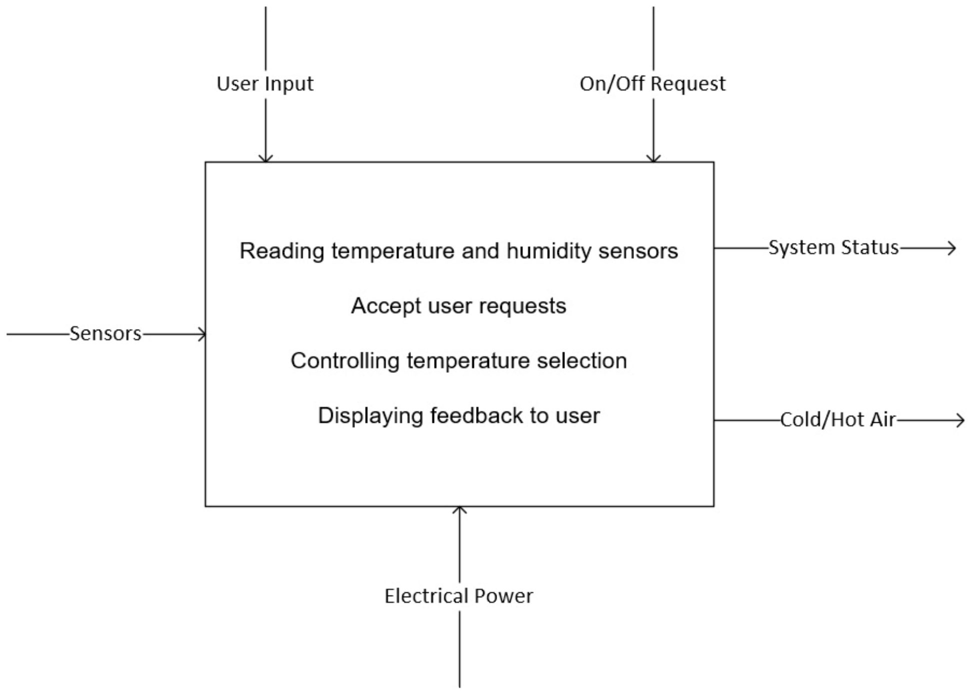
Spoilage in fruits and vegetables is most commonly caused by bacteria, yeast, or mold growth. Thus, delaying spoilage can be accomplished by delaying the growth of these three things. The controllable factors that have the greatest impact on microbial growth include "nutrient availability, moisture, pH, oxygen levels, and the presence or absence of inhibiting substances (e.g., antibiotics)" with temperature being the most impactful parameter in regards to food storage [4]. While having a low temperature is important for preserving food, stability of temperature also plays a significant role. Limiting temperature and humidity fluctuation caused by air movement that occurs when refrigerator doors are opened will help to slow microbial growth and enzyme activity that contribute to spoilage [4].

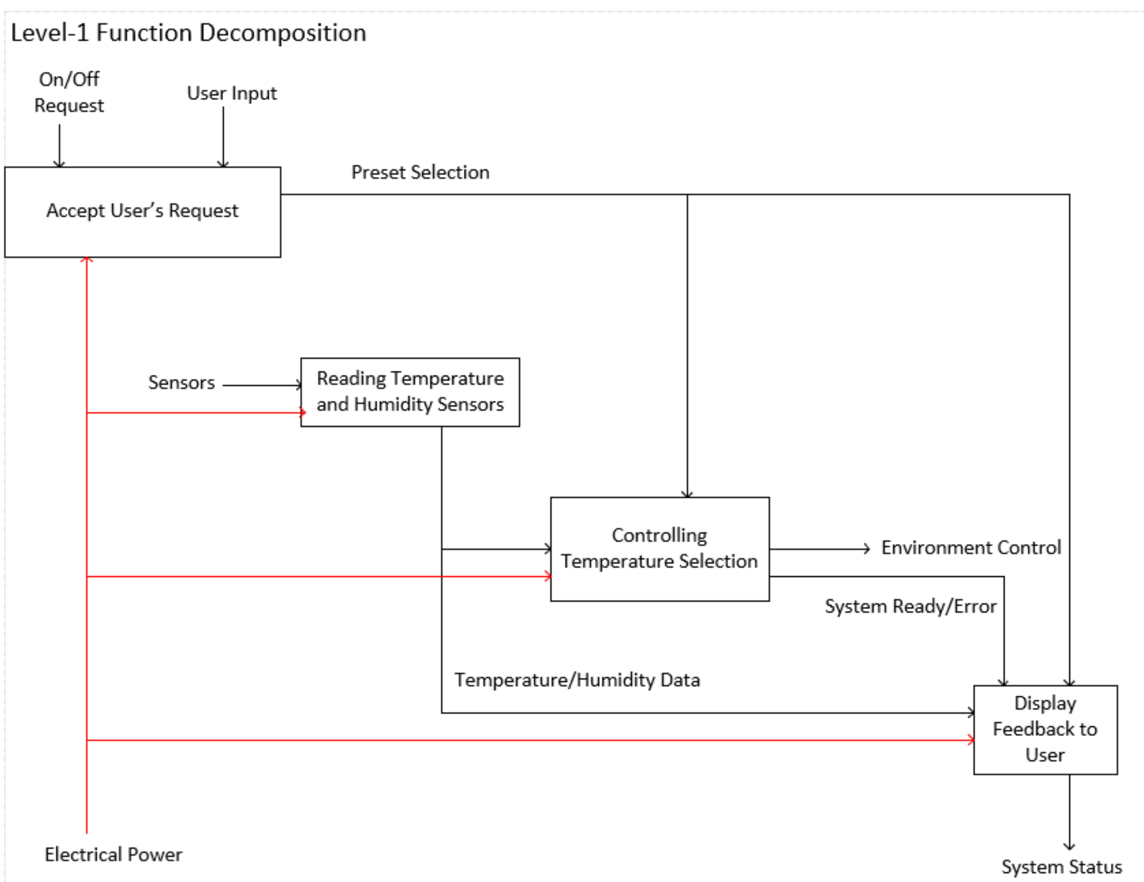
There is some fluctuation in the environment best suited for different fruits and vegetables. In general, vegetables need higher humidity than fruits, which usually equates to less air circulation. Further, ethylene producing plants need more air circulation to displace ethylene gases that speed up ripening [5]. Having separate compartments that can have varying temperatures, air circulation levels, and humidity levels is critical to delaying spoilage of produce. A thorough solution to extending the shelf life of fruits and vegetables needs to consider the interactions between optimizing air circulation and humidity while maintaining a stable low temperature in a way that is convenient and adaptable.

5) System Design



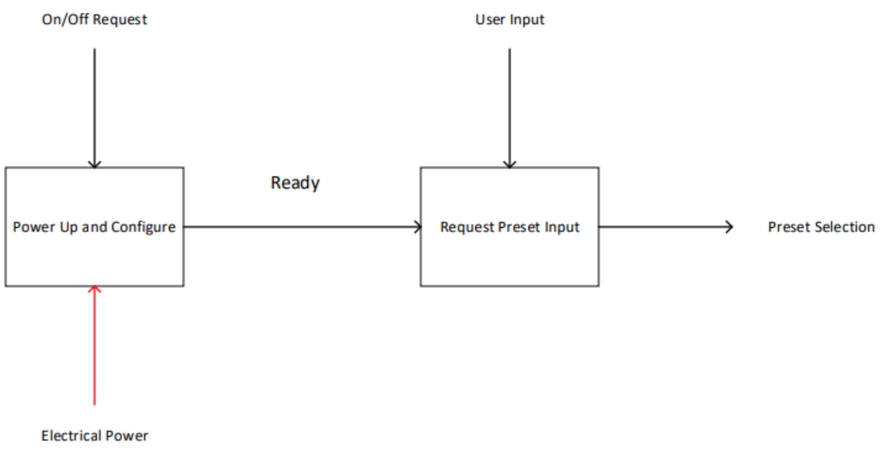
Level-0 Function Decomposition


User Input
On/Off Request
Reading temperature and humidity sensors
Accept user requests
Controlling temperature selection
Displaying feedback to user
System Status
Sensors
Cold/Hot Air
Electrical Power

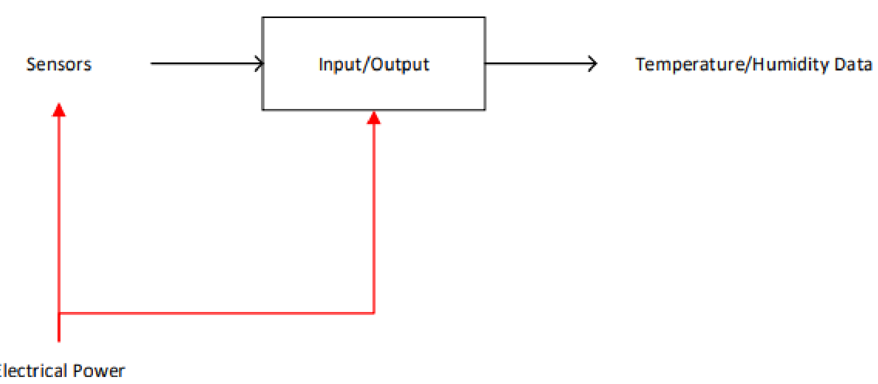

Level-1 Function Decomposition
On/Off Request
User Input
Accept User's Request
Preset Selection
Sensors
Reading Temperature and Humidity Sensors
Controlling Temperature Selection
Environment Control
System Ready/Error
Temperature/Humidity Data
Display Feedback to User
Electrical Power
System Status

Level-2 Function Decompositions (4)

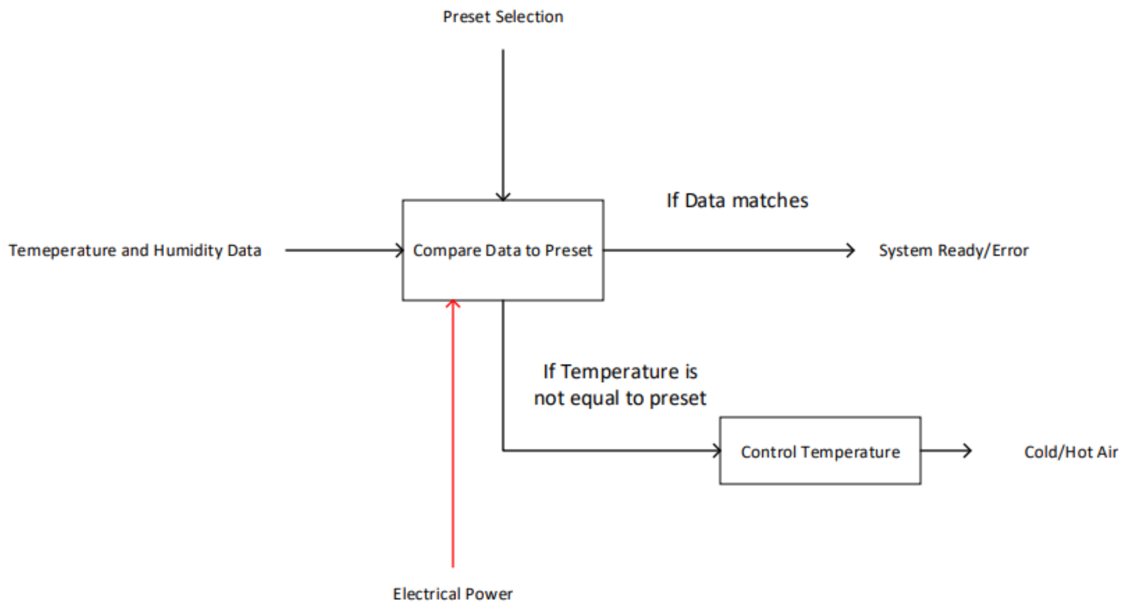
Function: Accept User Request



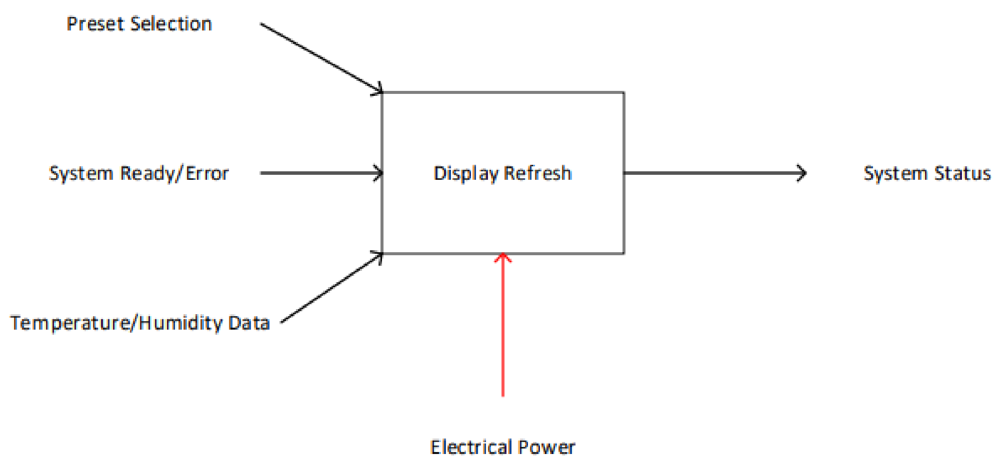
Function: Reading Temperature and Humidity Sensors



Function: Controlling Temperature Selection



Function: Reading temperature and humidity



6) Preliminary Experimental Plan

Requirements for experiential validation.

- Existing data on preserving different types of food shall be verified.
- The fridge shall have constant temperature and humidity.
- The fridge touchscreen will have data and quick sets for various fruits and vegetables.

Experiments to be conducted next semester.

By controlling the temperature and humidity we should obtain similar results to the university of Maine. We can verify this by setting up a normal fridge with the same food in it as the modified fridge. By comparing the two groups of food from the same source we will be able to record how much longer one sample lasts then the other. A test should be performed for vitamin A in the control and experimental groups. The experiment involving vitamin A samples is much more scientific then simply looking at or smelling the food to see if it is rotten. Data points from multiple vitamin A tests will be used to make comparison graphs where we will be able to prove that our fridge is having the effect that we are looking for.

Besides the main tests there are a few small tests that should be done. It will be easy to verify through the sensors that our temperature and humidity are constant and at the desired levels. If our output data shows, there is too much variance in our conditions then our smart fridge shall be adjusted.

Power usage is another thing that we can check. A simple current measuring device

7) Tasks to complete

- Have the Programming module to communicate with the Display, Temperature and Humidity Sensors, and Stepper motors
- Have the Temperature and Humidity sensors and Display give information back to the processing module

- Have the programming module control the relays to power the fridge and stepper motors when needed to
- Have a DC power supply go to a buck converter to distribute power to the processing module and relays
- Have the Temperature and Humidity Sensors give back data to the processing module to determine other actions

Responsibilities

Everyone on the team will need to take VSL training before the building part of the project. We will need this to use the lab that will be our place of storage for our fridge. This will make sure if someone cannot help due to reasons, other people on the team can support and help with the work.

Nestor will be in charge of finding a fridge to modify. This will be done using websites like Craigslist or Facebook Store to find the fridge we would need. Our primary location to find a fridge will be in the Northern VA area. Once we find a fridge that looks similar to our conceptual sketch, we will decide as a team if this will be the fridge we wish to use. If we have a majority vote (4 out of 6 votes) say yes, Nestor and another member of the team will be in charge of bringing the fridge to our allowed location (Peterson 4000) to build and modify the fridge.

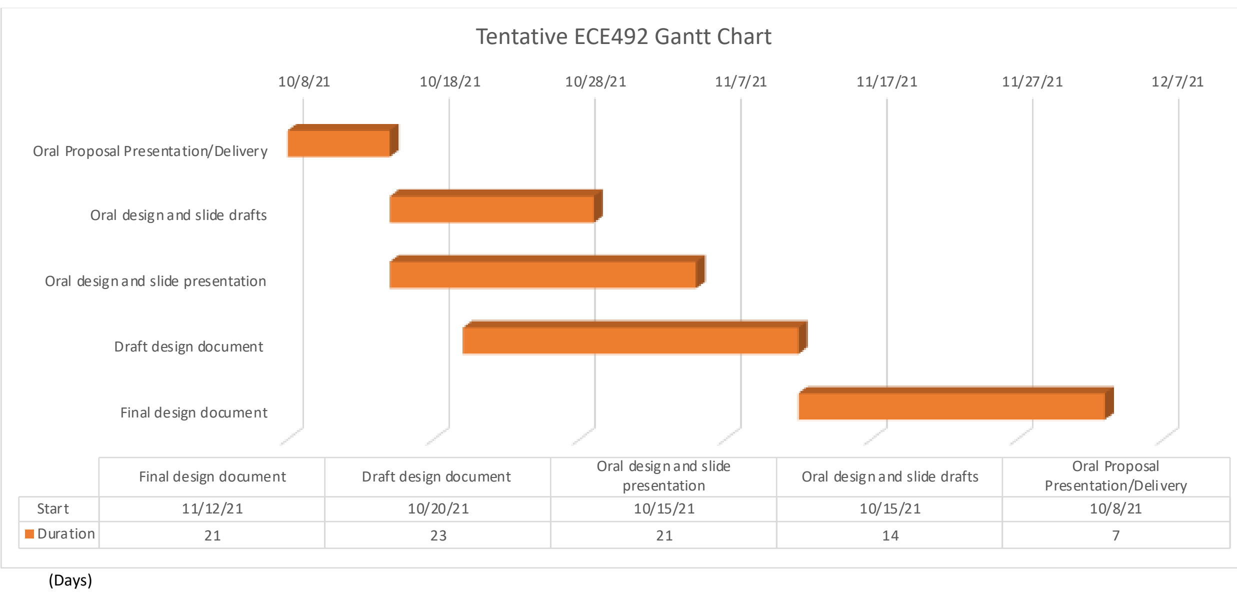
Ahmed, Zach, and Cameron will be the primary members in making the code for the processing module. Nestor, Michael, Matthew will be supportive members on this task and will assist the primary team if requested. The code will be either Python3 or C programming language. We will use code that is either made by the team or use open-source software that can be implemented with the components. Once that is done, we will then combine the different pieces of code together for the fridge.

Ahmed, Zach, and Cameron will lead the team in finding ways to have the program have better overall performance with the rest of the team supporting. Ahmed will be in charge of making sure the code's performance overall is good because of his background knowledge in coding.

Nestor, Michael, Matthew will be the primary members for designing the fridge and the electrical components. Ahmed, Zach, and Cameron will be supportive members on this task and will assist the primary team if requested. These tasks include wiring, checking for electrical safety, soldering, and any other task that uses power.

Schedule (ECE 492)

Assignment	Due date	Deliver to	Status	FS Feedback by	Part of the syllabus schedule		Key
Project Title Form Delivery	10-Sep	FS,CC	DONE	N/A	Yes		FS= Faculty Supervisor
Draft Proposal	1-Oct	FS	DONE	Oct 5th	Yes		CC= Course Coordinator
Proposal Presentation Slides Draft	6-Oct	FS	DONE	Oct 8th	Our own schedule		
Proposal Delivery	8-Oct	FS,CC		Oct 15th	Yes		
Proposal Presentation	15-Oct	FS		Oct 15th	Yes, Pushed back by a week		Final Assignment Due
Oral Design Draft	27-Oct	FS		Oct 29th	Our own schedule		December 3rd
Presentation Slides Delivery Draft	27-Oct	FS,CC		Oct 29th	Our own schedule		
Oral Design Review	5-Nov	FS		Nov 5th	Yes		Prefer To Finish By
Presentation Slides Delivery	5-Nov	FS,CC		Nov 5th	Yes		November 23rd
Draft Design Document Delivery	12-Nov	FS		Nov 17th	Yes		
Design Document Delivery	3-Dec	FS,CC		Dec 14th	Yes		



8) Potential Problems

Knowledge and skills to be learned:

[1] Each team member shall take the VSL training before using the bioengineering lab on GMU campus

[2] Team members shall become familiar with cooling/heating components that will be potentially used, i.e. Peltier Cells, etc.

[3] Team members should research specific nutritional information for potential produce to be stored properly in the end product, i.e. What ideal temperature to preserve specific vitamins, etc.

The team will repurpose used fridge(s) in this project, therefore some risks shall be considered:

- Team members shall address the potential sanitary issues.
- Team members shall be careful when disassembling/reassembling the fridge knowing the hazards of electric shock and potential decompression of the pressurized system.
- Team members shall research the components to be used and assess compatibility with the existing components within the fridge.

9) References

[1] Centers for Disease Control and Prevention, “Only 1 in 10 adults get enough fruits or vegetables,” *Centers for Disease Control and Prevention*, 16-Nov-2017. [Online]. Available: https://www.cdc.gov/media/releases/2017/p1116-fruit-vegetable-consumption.html. [Accessed: 13-Oct-2021].

[2] UN Environment Programme, “Worldwide Food Waste - ThinkEatSave,” *THINKEATSAVE REDUCE YOUR FOODPRINT*, 2013. [Online].

Available: https://www.unep.org/thinkeatsave/get-informed/worldwide-food-waste. [Accessed: 13-Oct-2021].
[3] U.S. Department of Agriculture, “Vegetables,” *MyPlate*. [Online]. Available: https://www.myplate.gov/eat-healthy/vegetables. [Accessed: 13-Oct-2021].
[4] R. P. Singh and N. W. Desrosier, “Packaging,” *Encyclopædia Britannica*, 23-Aug-1998. [Online]. Available: https://www.britannica.com/topic/food-preservation/Packaging#ref50584. [Accessed: 13-Oct-2021].
[5] J. Nichols, “Refrigerator humidity effects on produce quality,” *MSU Extension*, 25-Sep-2018. [Online]. Available: https://www.canr.msu.edu/news/refrigerator_humidity_effects_on_produce_quality. [Accessed: 13-Oct-2021].
[6] K. L. B. Gast, “Bulletin #4135, “Storage conditions: Fruits and vegetables,” *Cooperative Extension Publications*, 2001. [Online]. Available: https://extension.umaine.edu/publications/4135e/. [Accessed: 13-Oct-2021].

Resources to consider

Patents

[7] LG Electronics Inc, "Method for controlling operation of refrigerator with two evaporators" U.S. Patent US6883603B2 Jan 22, 2004. [Online]. Available: https://patents.google.com/patent/US6883603

[8] Del Monte Foods Inc, "Fruit and vegetable preservation process," U.S. Patent CA2791028A1 Apr 23, 2013. [Online]. Available: https://patents.google.com/patent/CA2791028A1/en

Research

[9] J. A. D. Nohay *et al.* "Design and fabrication of a portable solar powered thermoelectric refrigerator for insulin storage," Control and System Graduate Research Colloquium (ICSGRC), 08-Aug-2020. [Online]. Available: https://ieeexplore.ieee.org/document/9232573. [Accessed: 14-Oct-2021].

[10] P. R. V. Meris *et al.* "Design, fabrication and testing of portable thermoelectric refrigerator with Arduino-based temperature controller for Prevnar 13 vaccine," Control and System Graduate Research Colloquium (ICSGRC), 08-Aug-2020. [Online]. Available: https://ieeexplore.ieee.org/document/9232625. [Accessed: 14-Oct-2021].

Appendix B- Design Document (ECE-492)

Sustainable Fridge

ECE Team #11

Cameron Flores, Nestor Arellano, Michael Harm, Matthew Phillips, Zachary Solini, Ahmed Jaber

Problem Statement (Short Introduction)

The motivation behind the sustainable fridge project is for users to have an increase of fruits and vegetables in their diet. Current crisper drawer technology is unable to keep produce fresh for extended periods of time. To rectify this the team has devised a new crisper drawer design utilizing technology that is not currently implemented in crisper drawers. These new crispers will have an insulated environment such that the temperature and humidity of the drawer will be tailored towards the contents that are stored in the drawer. This environment will be monitored by a processing module with a temperature and humidity sensor. The environment will be adjusted by distributing the refrigerator unit’s cooling power to an indicated crisper, this will be achieved by implementing a combination relays and solenoid valves. Feedback will be returned through a display, and the user will be able to control the temperature and humidity setting of each crisper. This will set the control for the processing module to compare with the current environment. If the sensor data does not match the preset data, an adjustment will be made, thus optimizing the environment for the contents inside. The user will also be able to turn on or off the drawers individually.

Requirements Specification

The user shall be able to store larger quantities of natural produce in this fridge and expect them to last longer as healthy and appealing as they were at the time of storage. The user shall also not worry about conflicting storage requirements for each type of produce stored as there shall be independent chambers with the required conditions preset for each type of produce.

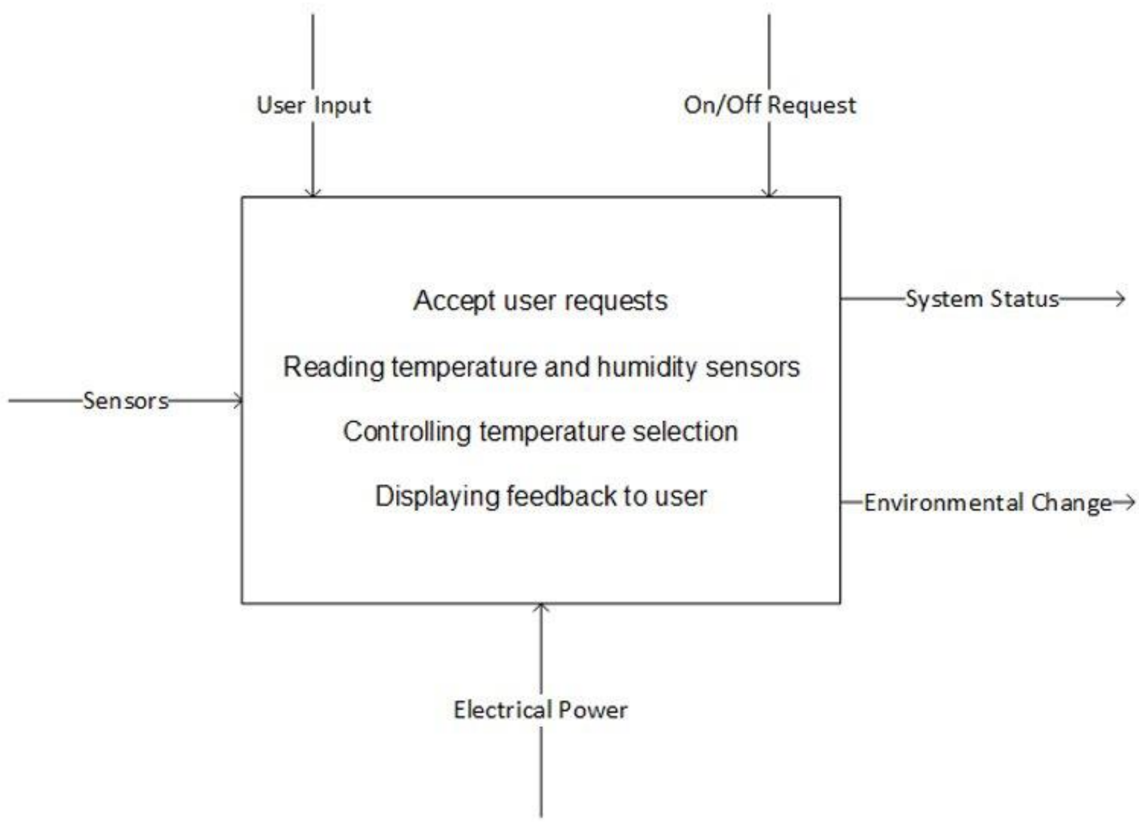
Therefore, the fridge shall have large chambers with isolated environments so that one would not interfere with another's. Moreover, there shall be presets of temperature and humidity loaded for each category of produce according to their specific requirements of ideal storage settings.

The Fridge shall also contain an LCD which will be the main output media to the user. There will be a main screen, displaying a representation of each chamber in list view, as well as a settings bar at the bottom of the list. The user will use buttons to navigate through this main screen, then for each chamber selected, the user will be presented with a list of presets to select according to the type of produce needed to be stored in the chamber in question. As an alternative to the presets, the user will also be presented with an option to disable a certain crisper and re-enable when they see fit to do so. The settings bar will provide the user options including changing Display Brightness and changing units between metric and imperial.

Buttons shall be the main input media from the user. There shall be four main buttons: up, down, back, and select. These buttons should be debounced through software handled by the slave raspberry pi, where an edge will be registered after a certain amount of time from the button press.

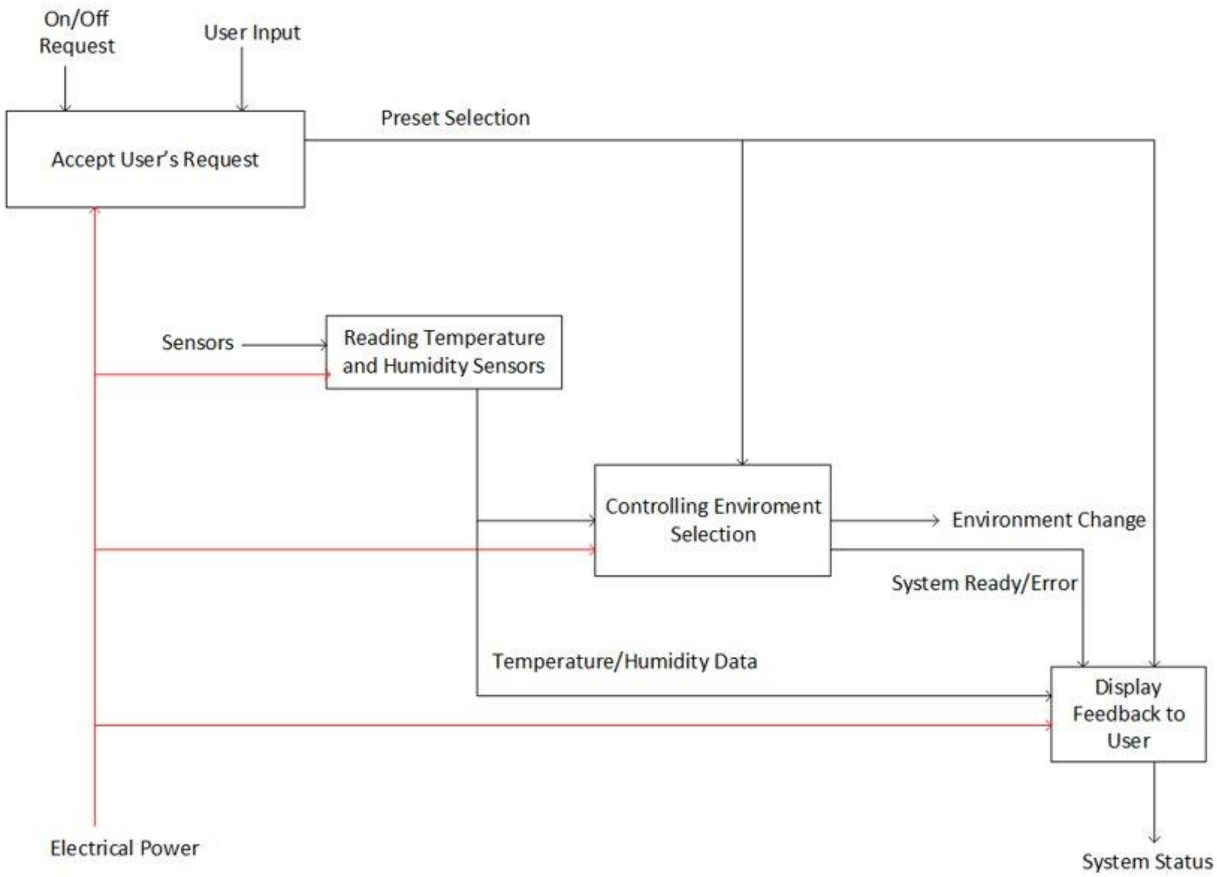
System design/architecture

Level-0 Functional Decomposition: The white box contains the top-level functions of the system, inputs, and outputs.



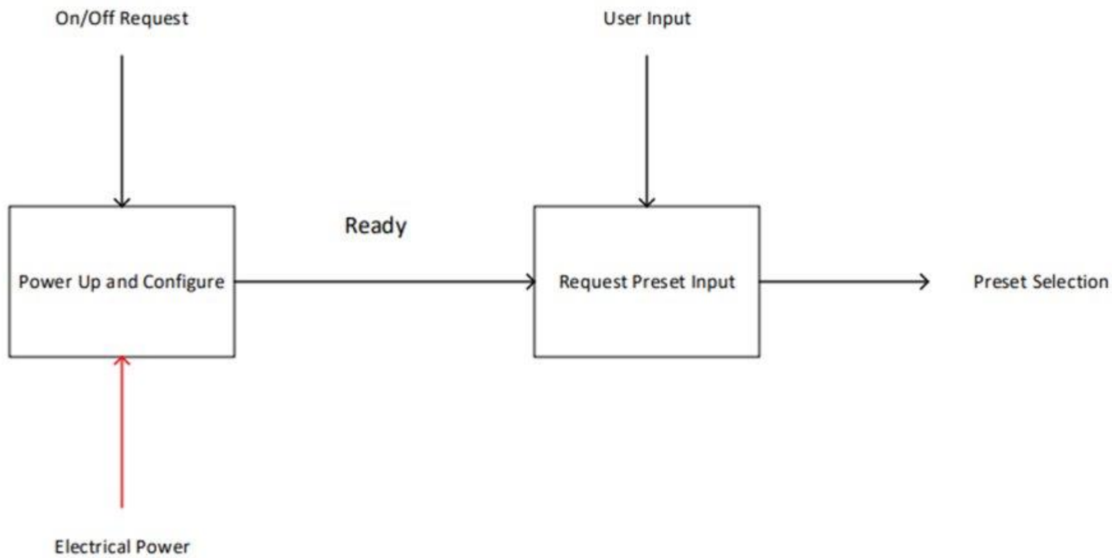
Level-1 Functional Decomposition: Demonstration of the relationship between the top-level functions specified in our Level 0 decomposition.

Level-1 Function Decomposition

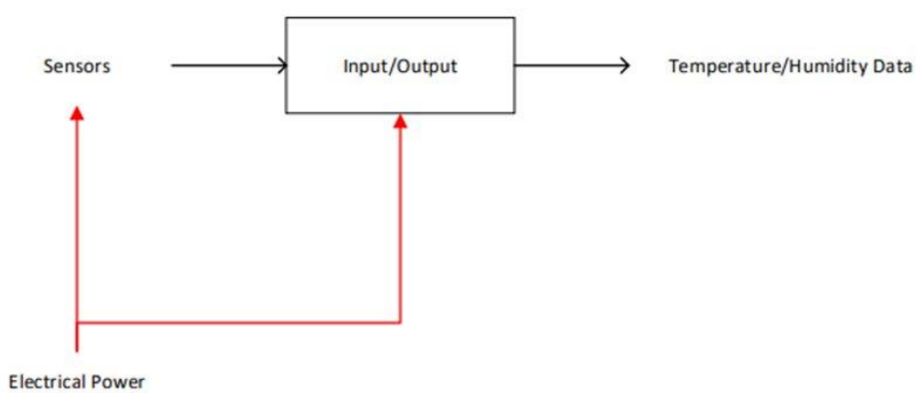


Level 2 Functional Decomposition: A closer look at each Level-1 function as it relates to inputs, outputs, and other functions. We have a Level-2 flow diagram for each of our four top-level functions specified above.

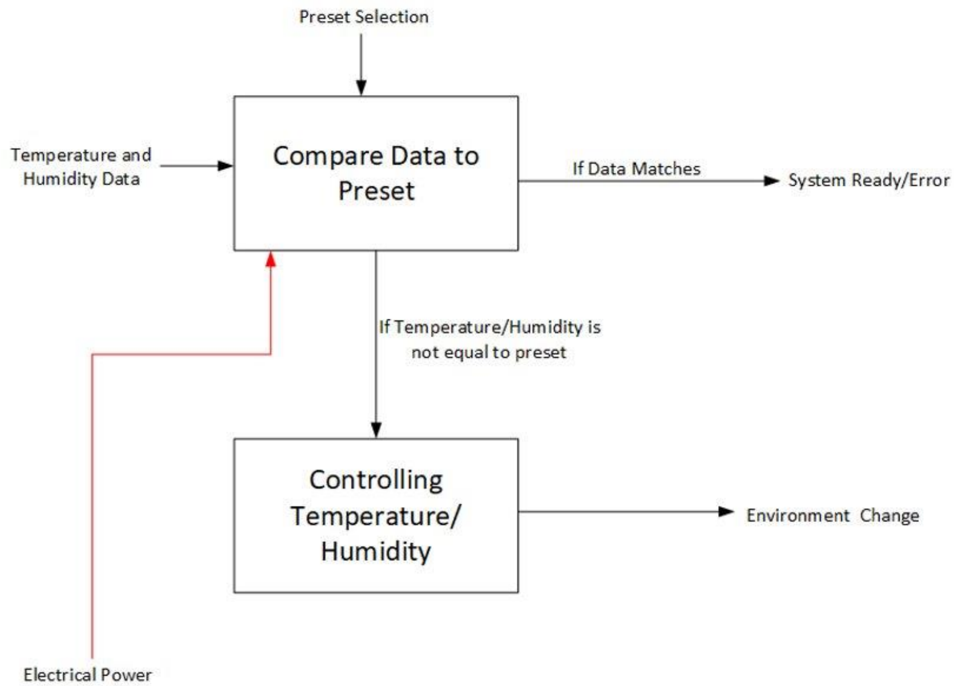
Function: Accept User Request



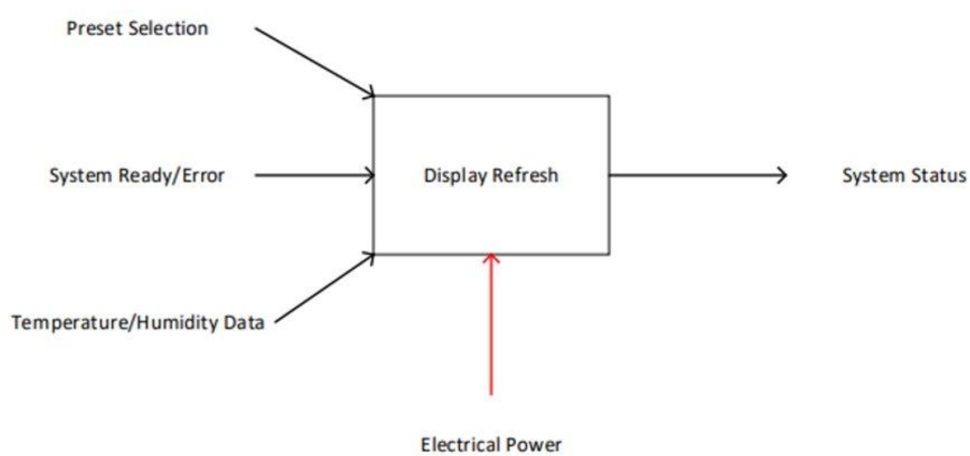
Function: Reading Temperature and Humidity Sensors



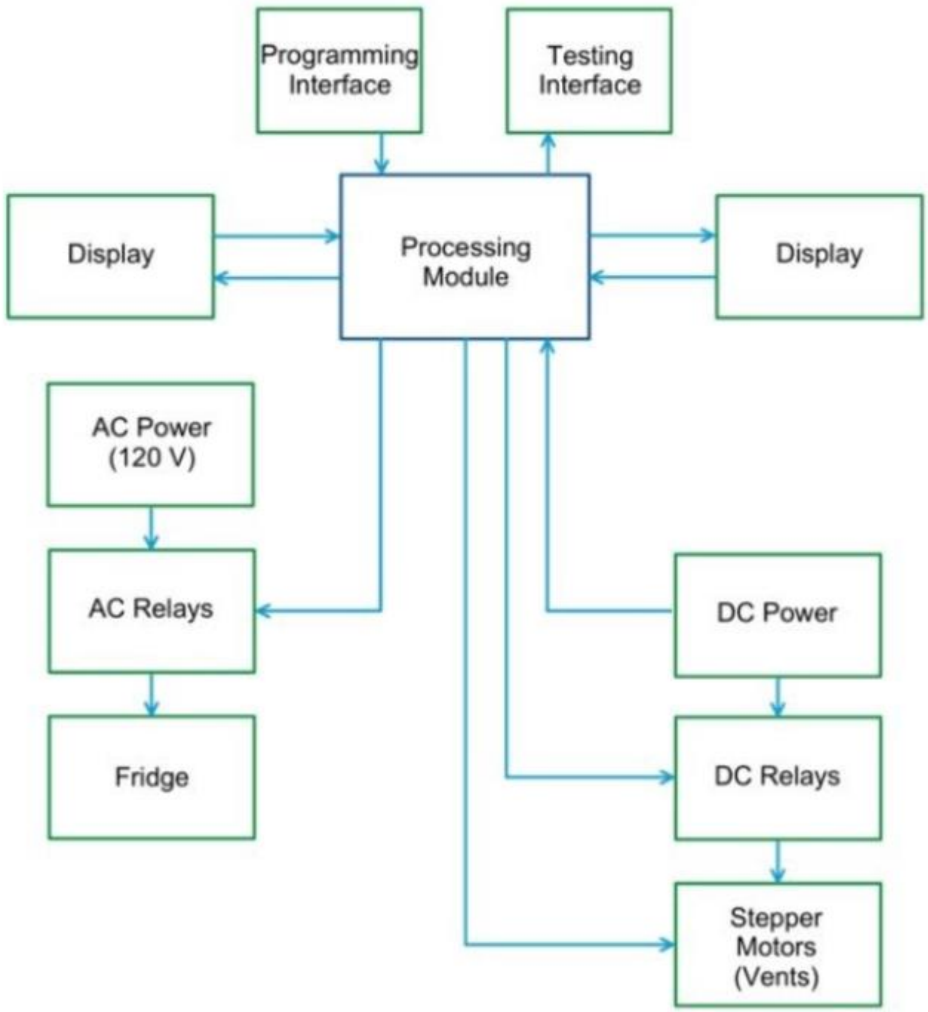
Function: Controlling Environment Selection



Function: Reading temperature and humidity



Design Architecture



Background knowledge

Key factors in produce degradation include Temperature, Humidity, and Ethylene gas sensitivity/production. Home refrigerators tend to be set at a temperature of approximately 4-5 °C. This temperature is a compromise to maintain food fresh for longer periods of time without the risk of freezing other items located in the shared refrigerated space [4]. As seen in TABLE 1 a more appropriate temperature for produce should be set to either a temperature below 4 degrees, as is the case for most leafy greens or above as recommended for okra, papaya, and passion fruit [3][4]. Certain plants known as “Climacteric” produce a colorless gas known as Ethylene during various stages of their lifecycle. For climacteric plants, the gas can cause them to ripen faster, making them sweeter and softer. In non-climacteric plants such as leafy greens this gas can cause produce to decay prematurely. This relationship must be considered when storing varied species of produce together [3][4][5][6]. Another factor to be taken into consideration is dehydration, since the produce is no longer attached to their corresponding plant it is left without a supply of water and minerals to continue its biological processes, meanwhile it will continue to consume what little resources it has until it has exhausted them. For this reason, to reduce water loss through transpiration, the produce will be stored in at a high humidity of 90% as indicated by the published data [3][4]. The reduction is water lost through the skin of the produce should allow the stored produce to continue its biological functions for a longer period before commencing the decay process [5][7]. The tables 1 & 2 provide some example data that will be used to demonstrate the teams process for deciding on the presets that will be implemented in the design. The data set utilized in this section has been supported by two scholarly sources [3][4]. The newly designed fridge will be able to support the final presets but will not be limited to the produce listed. The user will have to decide which category their new produce item falls into. The tables below indicate how the preset groupings are formed.

TABLE 1
Ideal Handling Conditions for Various Produce

Produce item	Storage Temperature (C)	Storage Temperature (F)	% Relative Humidity	Ethylene Production	Ethylene Sensitivity	Storage Life (days)
Acerola cherry	0	32	85 - 90	VL	L	42 - 56
Apple (malus pumila)	-1	30	90 - 95	VH	H	90 - 180
Apple (Yellow Newton)	4	40	90 - 95	VH	H	30 - 60
Apricot	-0.5	31	90 - 95	M	M	7 - 21
Artichoke	0	32	90 - 100	VL	L	7 - 21
Arugula	0	32	95 - 100	VL	H	7 - 10
Asparagus	2.5	36	95 - 100	VL	M	14 - 21
Avocado	3	37	85 - 90	H	H	14 - 28
Banana	13	56	90 - 95	M	H	7 - 28
Celery	0	32	98 - 100	VL	M	30 - 60
Chard	0	32	95 - 100	VL	H	10 - 14
Cilantro	0	32	95 - 100	VL	H	7 - 14
Collards	0	32	95 - 100	VL	H	10 - 14
Corn	0	32	95 - 98	VL	L	5 - 8
Cantaloupes	2	36	95 - 98	H	M	14 - 21
Mushrooms	0	32	90 - 95	VL	M	7 - 14
Okra	7	45	90 - 95	L	M	7 - 10
Onions	0	32	65 - 70	VL	L	30 - 240
Papaya	7	45	85 - 90	M	M	7 - 28
Parsinp	0	32	95 - 100	VL	H	120 - 180
Passion fruit	10	50	85 - 90	VH	M	21 - 28
Peach	0	32	90 - 95	M	M	14 - 28
Pear	0	32	90 - 95	H	H	60 - 210
Peas	0	32	90 - 98	VL	M	7 - 14
Pepper (Bell)	7	45	95 - 98	L	L	14 - 21
Pineapple	7	45	85 - 90	L	L	14 - 28
Pomegranate	7	45	90 - 95	VL	L	60 - 90
Rutabaga	0	32	98 - 100	VL	L	120 - 180
Spinach	0	32	95 - 100	VL	H	10 - 14
Summer squash	7	45	95 - 100	L	M	7 - 14
Sweet potato	13	55	85 - 95	VL	L	120 - 210
Tamarind	2	36	90 - 95	VL	L	21 - 28
Tomatillo	7	45	85 - 90	VL	M	21 - 28
Tomato	10	50	85 - 90	H	L	7 - 21
Watercress	0	32	95 - 100	VL	H	14 - 21
Watermelon	10	50	90 - 95	VL	H	14 - 21

** This table consists of a few examples taken from the appendix of the book *Postharvest Technology of Horticultural Crops* [3].
These examples will be used to demostrate the process of creating the system presets.
Groups may include all of the present produce, but are not be limited to the items in this table.

TABLE 2
Key

Ethylene Gas production rate		Ethylene Gas sensitivity
Very Low (VL)	<0.1 ul/kg-hr	Low (L)
Low (L)	0.1 ul/kg-hr	Moderate (M)
Moderate (M)	0.1 -> 10 ul/kg-hr	High (H)
High (H)	10 -> 100 ul/kg-hr	
Very High (VH)	<100 ul/kg-hr	

** This table is also refrenced from *Postharvest Technology of Horticultural Crops* [3].
Shown here are the five levels of Ethylene gas production along with their corresponding rates.
The three levels of Ethylene sensitivity are also included.

TABLE 3

Produce Sorted by Ethylene Susceptibility

Low susceptibility	Moderate susceptibility	High susceptibility
Acerola cherry	Apricot	Apple
Artichoke	Asparagus	Arugula
Corn	Celery	Avocado
Onions	Cantaloupes	Banana
Bell Pepper	Mushrooms	Chard
Pineapple	Okra	Cilantro
Rutabaga	Papaya	Collards
Sweet potato	Passion Fruit	Parsnip
Tamarind	Peach	Pear
Tomato	Peas	Spinach
Pomegranate	Summer squash	Watercress
	Tomatillo	Watermelon

**Produce from TABLE 1 is sorted based on Ethylene gas sensitivity.
The three groups map to the three sensitivity levels mentioned in TABLE 2.

TABLE 4

Low Susceptibility Group's Ethylene Production

VL	L	M	H	VH
Acerola cherry	Corn		Tomato	
Artichoke	Onions			
Rutabaga	Bell Pepper			
Sweet potato	Pine apple			
Tamarind				
Pomegranate				

**Low susceptibility group from TABLE 3 sorted by ethylene gas production rate

TABLE 4.1

Teperature Preferences of Produce with Low Susceptibility & (VL) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
Acerola cherry	Tamarind	Pomegranate	Sweet potato
Artichoke			
Rutabaga			

**TABLE 4 Very Low ethylene producers sorted by temperature preference

TABLE 4.2

Teperature Prefecences of Produce with Low Susceptibility & (L) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
Corn		Bell pepper	
Onions		Pineapple	

**TABLE 4 Low ethylene producers sorted by temperature preference

TABLE 4.3

Teperature Prefecences of Produce with Low Susceptibility & (H) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
			Tomato

**TABLE 4 High ethylene producers sorted by temperature preference

TABLE 5

Moderate Susceptibility Group's Ethylene Production

VL	L	M	H	VH
Asparagus	Okra	Apricot	Cantaloupes	PassionFruit
Celery	Summer squash	Papaya		
Mushrooms		Peach		
Peas				
Tomatillo				

**Moderate susceptibility group from TABLE 3 sorted by ethylene gas production rate

TABLE 5.1

Teperature Prefecences of Produce with Moderate Susceptibility & (VL) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
Celery	Asparagus	Tomatillo	
Mushrooms			
Peas			

**TABLE 5 Very Low ethylene producers sorted by temperature preference

TABLE 5.2

Teperature Prefecences of Produce with Moderate Susceptibility & (L) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
		Okra	
		Summer squash	

**TABLE 5 Low ethylene producers sorted by temperature preference

TABLE 5.3

Teperature Prefecences of Produce with Moderate Susceptibility & (M) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
Apricot		Papaya	
Peach			

**TABLE 5 Moderate ethylene producers sorted by temperature preference

TABLE 5.4

Teperature Prefecences of Produce with Moderate Susceptibility & (H) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
	Cantaloupes		

**TABLE 5 High ethylene producers sorted by temperature preference

TABLE 5.5

Teperature Prefecences of Produce with Moderate Susceptibility & (VH) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
			Passion fruit

**TABLE 5 Very High ethylene producers sorted by temperature preference

TABLE 6
High Susceptibility Group's Ethylene Production

VL	L	M	H	VH
Arugula		Banana	Avocado	Apple
Chard			Pear	
Cilantro				
Collards				
Parsnip				
Spinach				
Watercress				
Watermelon				

**High susceptibility group from TABLE 3 sorted by ethylene gas production rate

TABLE 6.1
Teperature Prefecences of Produce with High Susceptibility & (VL) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
Arugula			Watermelon
Chard			
Cilantro			
Collards			
Parsnip			
Spinach			
Watercress			

**TABLE 6 Very Low ethylene producers sorted by temperature preference

TABLE 6.2
Teperature Prefecences of Produce with High Susceptibility & (M) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
			Banana

**TABLE 6 Moderate ethylene producers sorted by temperature preference

TABLE 6.3

Teperature Prefecences of Produce with High Susceptibility & (H) Ethylene Production

30 - 35 (F)	35 - 40 (F)	40 - 45 (F)	> 45 (F)
Pear	Avocado		

**TABLE 6 High ethylene producers sorted by temperature preference

TABLE 7

Potential Groupings

Group 1 : 34 (F) : 90%	Group 2 : 32 (F) : 90%	Group 3 : 52 (F) : 90%	Group 4 : 50 (F) : 90%	Group 5 : 33 (F) : 90%	Group 6 : 44 (F) : 90%
Acerola cherry	Celery	Pomegranate	Tomato	Corn	Bell pepper
Artichoke	Mushrooms	Sweet potato	Passion fruit	Onions	Pineapple
Rutabaga	Peas	Watermelon		Apricot	Asparagus
Tamarind	Arugula	Banana		Peach	Tomatillo
Pear	Chard			Cantaloupes	Okra
Avocado	Cilantro				Summer squash
Apple	Collards				Papaya
	Parsnip				
	Spinach				
	Watercress				

**Compatible groupings formed by taking into account temperature preferences, ethylene sensitivity, and ethylene production rates.
The temperature for each group was determined by taking the average of all individual recommended temperatures.
Humidity was set at 90 % due to the similarity in humudity preferences across all groups. An exception being Onions which prefer low humidity.

Groups were formed by dividing the data up by low (L), moderate (M), and high (H) sensitivity to ethylene gas. Each of the three groups can then be divided by their own Ethylene gas production levels, very low (VL), low (L), moderate (M), high (H), and very high (VH) [3]. The subgroups can be further divided into their preferred storage temperature. Groups for presets can be formed by comparing temperature preference overlap. Additionally, it is important to note that ethylene gas compatibility must also be considered, high sensitivity items can be stored along with other sensitive items and low/very low ethylene gas producers.

TABLE 8
Recommended Groups

Group 1: 34 (F) : 90%	Group 2: 32 (F) : 90%	Group 3: 52 (F) : 90%	Group 4: 50 (F) : 90%
Acerola cherry	Celery	Bell pepper	Tomato
Artichoke	Mushrooms	Pineapple	Passion fruit
Rutabaga	Peas	Asparagus	
Tamarind	Arugula	Tomatillo	
Pear	Chard	Okra	
Avocado	Cilantro	Summer squash	
Apple	Collards	Papaya	
	Parsnip		
	Spinach		
	Watercress		

** Main presets to be implemented in sustainable fridge design.

The final 4 groups are characterized as follows:

Group 1:

This group pairs produce those outputs very low quantities of ethylene gas and is not sensitive to ethylene with produce that is sensitive to ethylene and produces substantial ethylene. The initial set of produce should not be affected by the high gas output of the second set, and the 2nd set will mainly interact with their own ethylene gas output. Lastly an additional constraint of temperature is to further extend the storage life of the produce.

Group 2:

This group combines produce that it both moderately and highly susceptible to ethylene, both groups produce very little ethylene making them a suitable match for joint storage.

Group 3:

Group 3 consists of produce with low ethylene sensitivity and low ethylene production combined with moderately ethylene sensitive produce which has very low, low, and moderate ethylene outputs. An additional trait this group shares is a preference for temperatures warmer than 0 °C.

Group 4:

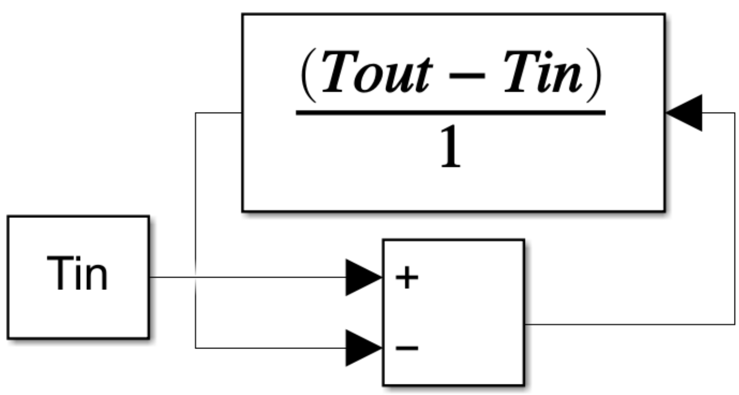
Group 4 will have low and moderate susceptibility produce with high and very high rates of ethylene gas production. Additionally, this group will also prefer warm temperature settings and will stay at a temperature of at least 7 °C.

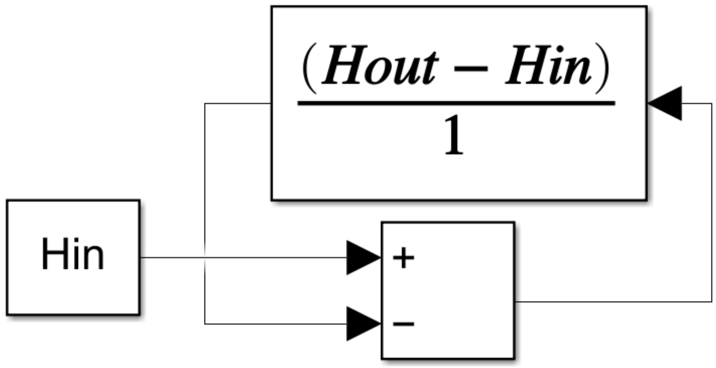
Lastly, if needed low ethylene susceptibility produce with a preference of warmer temperature may be stored in group 4 this would include produce such as: Pomegranate, Sweet potato, Bell peppers, and pineapple.

**Produce not listed in TABLE 1 will have to be put into one of these 4 groups depending on best fit between required produce parameters and device presets, this is recommended to achieve the longest storage life possible for the produce item.

Calculations to see if Temperature and Humidity in containers matches up with user selection

For the changing of the environment in the containers, the initial temperature and humidity will act as feedback to combine with the initial value





This leads with the transfer functions

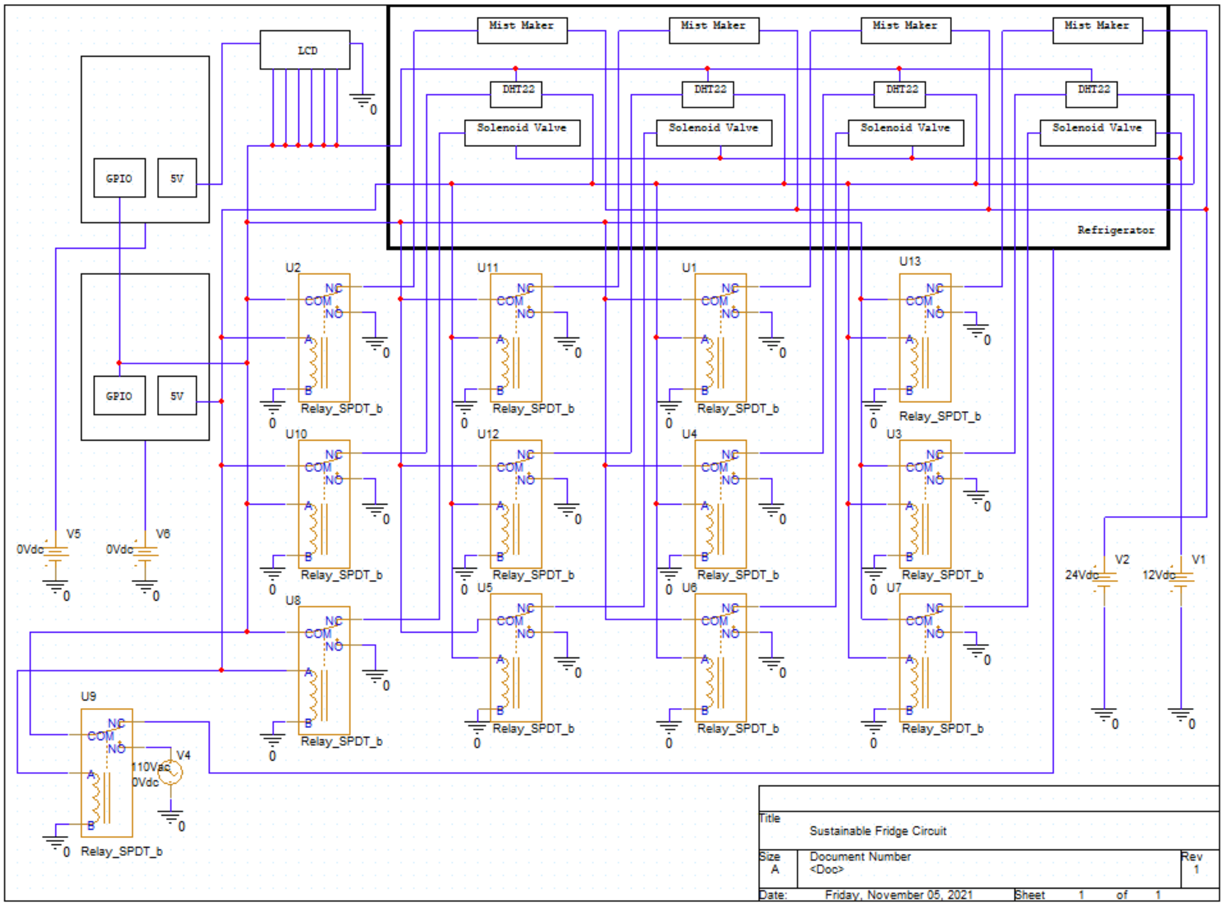
$$\frac{Tout}{Tin} = \frac{1}{1 + (Tout - Tin)} \qquad \frac{Hout}{Hin} = \frac{1}{1 + (Hout - Hin)}$$

Where: Tout – Final Temperature; Tin – Input Temperature; Hout – Humidity Output; Him – Humidity Input

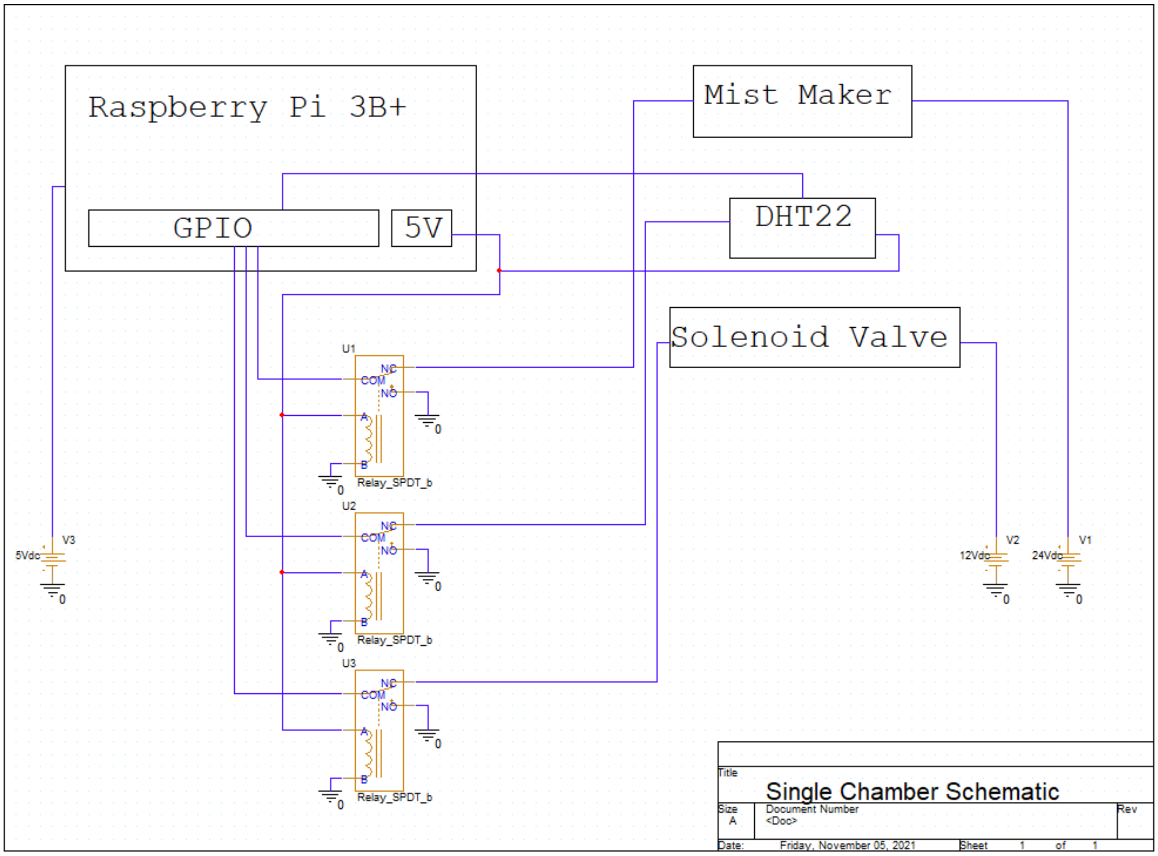
If Tout=Tin or Hout=Hin the function will equal to 1 signifying that the user’s input and the final value match up

Detailed Design

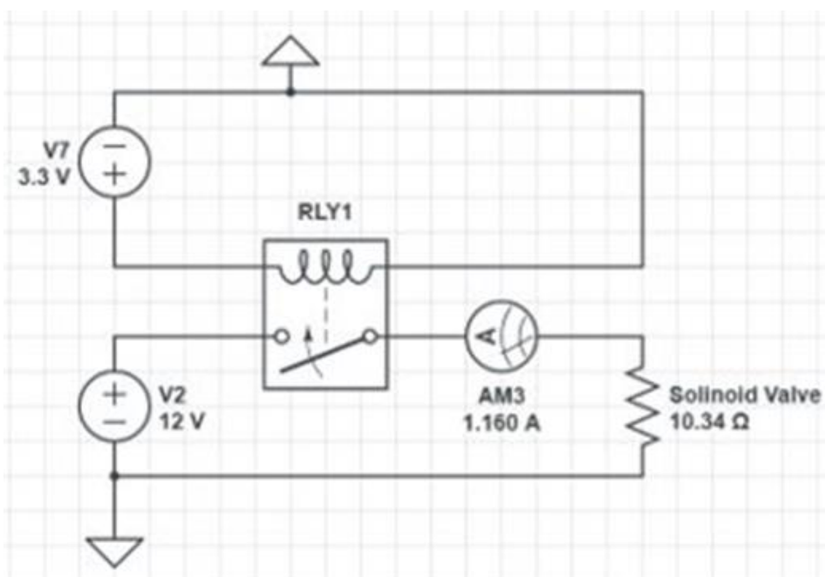
1. **Circuit Schematics, State diagrams, algorithm flowcharts**



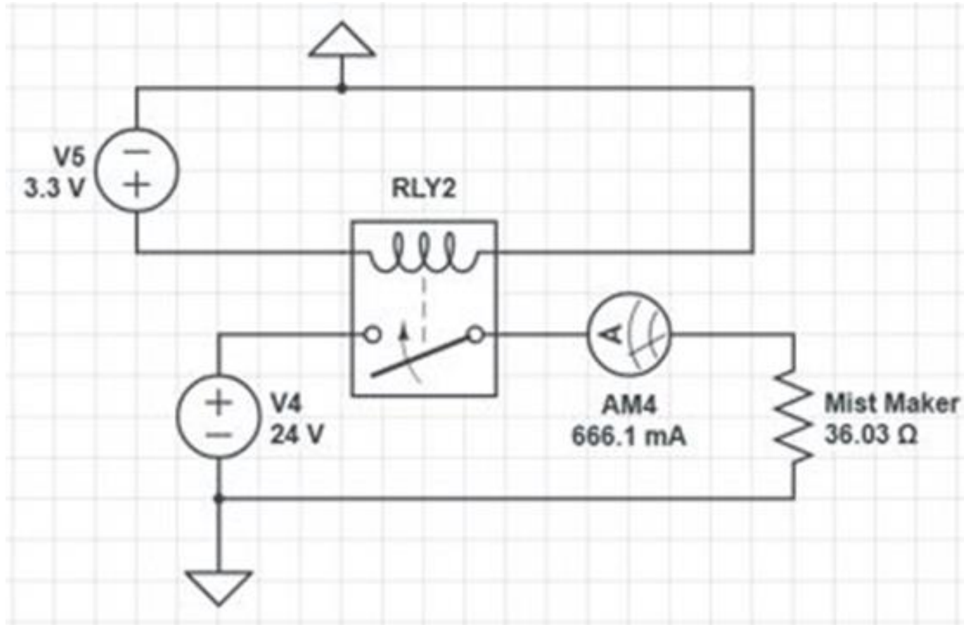
This initial circuit design will utilize at least 12 relays to control power delivery to all the individual system components. The refrigerator unit will be controlled by the Raspberry Pi through a smart power strip, represented here by the lower relay connected to 110 Vac. Two Raspberry Pi units will share the GPIO load through an I2C connection. Individual DC power supplies will also be controlled through the smart power strip.



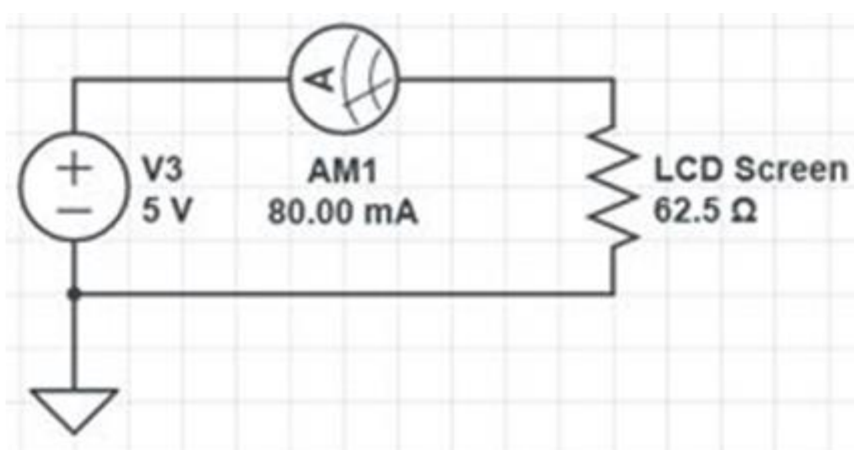
Each individual chamber will use 3 relays, a mist maker, a DHT22 sensor, and a solenoid valve.



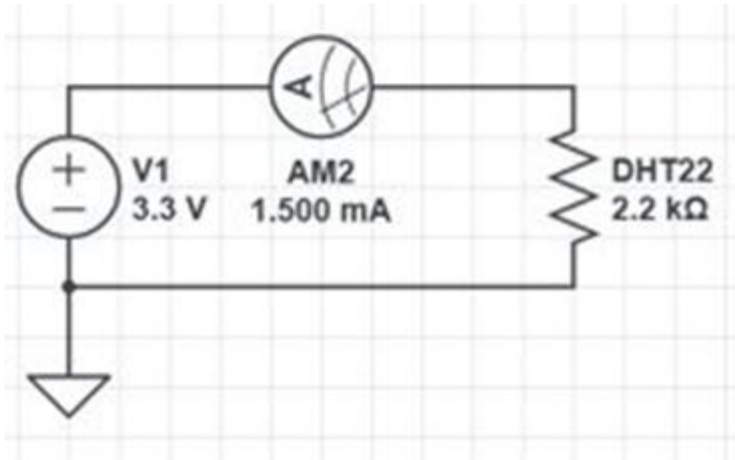
Schematic of the solenoid valve



Schematic of the mist maker



Schematic of the LCD Screen



Schematic of the DHT22

2. **Component and Interface description**

DHT22

The DHT22 is the component that is responsible for sensing and reporting back temperature and humidity data in the containers. How it works is that when given voltage, the sensor sends back a 40-bit array of data to give back to the processing module. There are 5 sets of 8 bits representing different values. From the most significant bit to least significant bit is the sets are humidity whole number, humidity decimal number, temperature whole number, temperature decimal number, and check number set [1]. The check number set is the sum of the values of the four previous sets [1]. If the check number set does not match the sum of the four sets, it will read as an error. If it matches, then the value is true.

Solenoid Valve

The Solenoid Valve is a component that is responsible for opening and closing the path for air to enter the container. How this is done is through a current running through the valve. When current runs through the valve, this will cause the pathway within the valve to open allowing air to pass through [2]. For it to close no current would have to run through.

16x4 LCD Screen

The LCD screen will be designed to display the current temperature/humidity in the fridge. How this works is when voltage is running through the device, it will be able to display text. The processing module will send out information to the LCD screen for the user to see the current values. It will show what options the user can do and display current values as well.

Buttons

The buttons will be used to select an action on the LCD screen. How this works is when the user pushes a button, the processing module will recognize this as an input and then run through the code to show a change in the display.

Drain System:

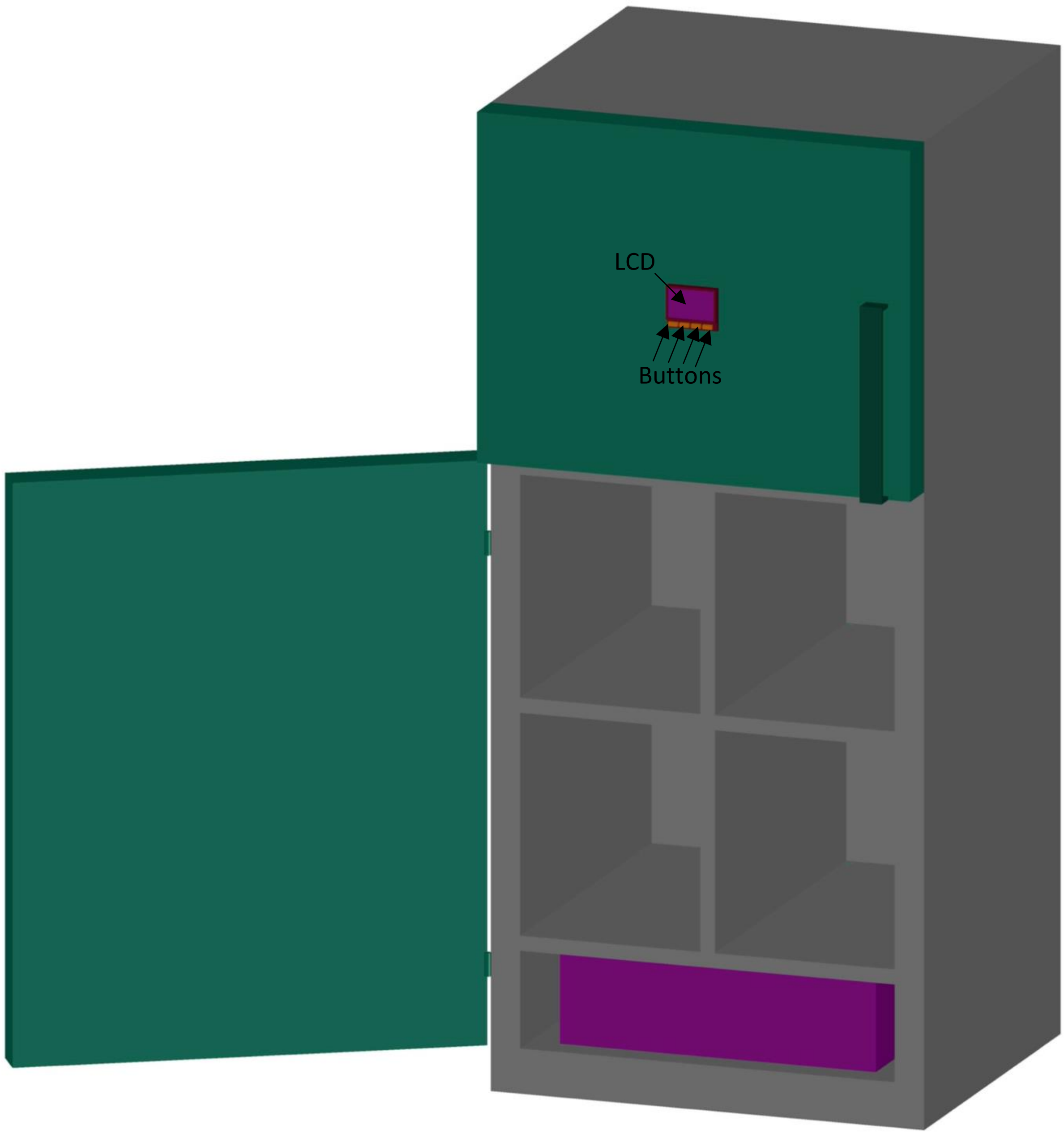
Each compartment will have a drain that will be connected to a drain tube which will deposit the extra water from the misters into the drain reservoir. To prevent unwanted temperature change in the chambers, we shall put a dip in the drain tubing. This idea is like the plumbing “traps” under sinks where the trapped water in the drain will prevent airflow. To prevent leakage, an overflow sensor will alert the user when the drain reservoir is full. The water level alarm will be a sensor that will notify the user that the drain reservoir needs to be emptied. This sensor will be attached near the top of the reservoir and will simply emit a repeated noise whenever it is submerged.

Interface

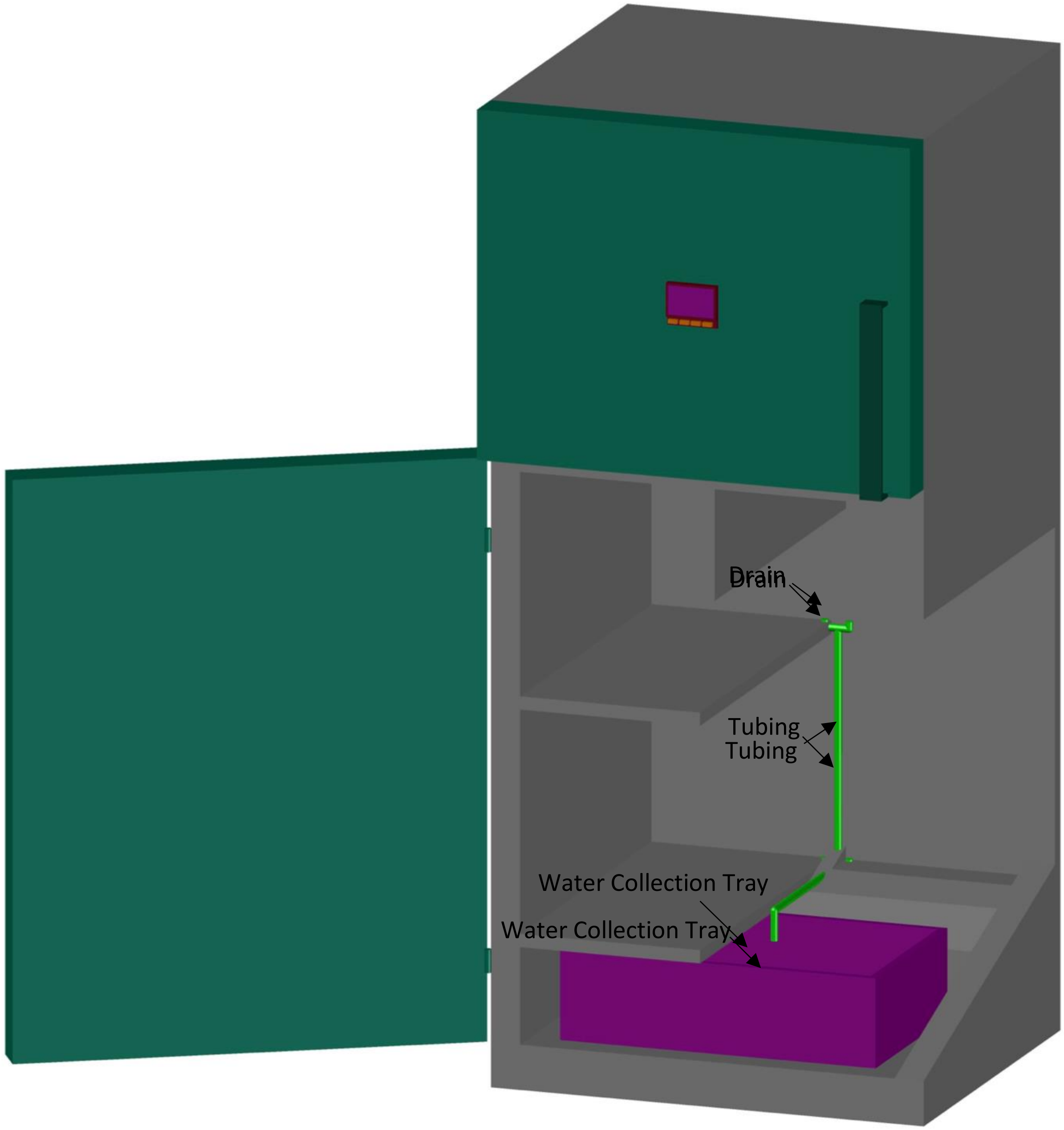
The interface will be the Raspberry Pi where all the components will meet and connect and communicate with one another. How this will work is that the results of one of the components will result in a change in another part of another component(s).

Mechanical Design

The following diagrams show the refrigerator body reflecting modifications to be done:

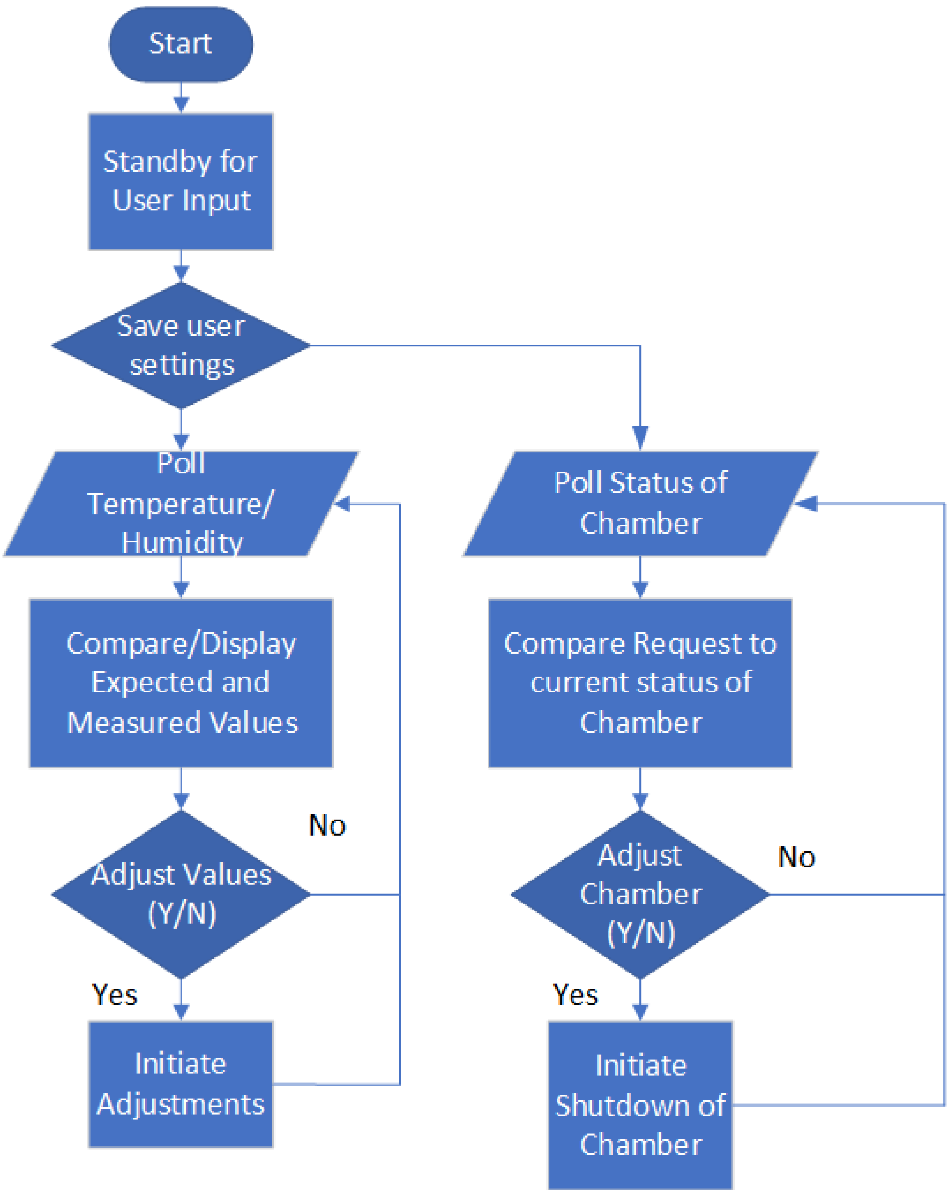


External View



Internal View

3. Flow diagrams + main parameters for simulation projects



How the Flow diagram works is that it will wait for user input. Once Given user input, it will save the user setting adjustments for later. Then it will go down two paths. Path one will get current readings of the temperature/humidity in the container. Next it will decide between two routes based on if the values match or not. If they do not match then it will initiate adjustments. Once it is done initiating adjustments it will go back in a loop to show the temperature/ humidity values. If it does match though it go to the temperature/ humidity value section.
Path two will check the current status of the container. Next it will compare that status with the requested command. Next it will decide between two routes based on that comparison. If they do not match then it will initiate a shutdown of the chamber and loop back to checking the status. If they do match then it just go back to the status section.

4. Detailed description of software structure

The software implemented will span through at least two raspberry pi's. First, the "slave" raspberry pi will handle the I/O interfacing with the user, which mainly consists of choosing a preset for each chamber as input which propagates into the "master", and sensor monitoring as output to the LCD. The "master" raspberry pi will poll through the chambers, comparing the temperature and humidity recorded by the sensors with the preset of the chamber in question. The readings will be sent to the "slave" which will display to the LCD, and the "master" will actuate the needed components that will moderate the temperature and/or humidity as needed.

The LCD will be the main output media to the user. There will be a main screen, displaying a representation of each chamber in list view, as well as a settings bar at the bottom of the list. The user will use buttons to navigate through this main screen, then for each chamber selected, the user will be presented with a list of presets to select according to the type of produce needed to be stored in the chamber in question. As an alternative to the presets, the user will also be presented with an option to disable a certain crisper and re-enable when they see fit to do so. The settings bar will provide the user options including changing Display Brightness and changing units between metric and imperial.

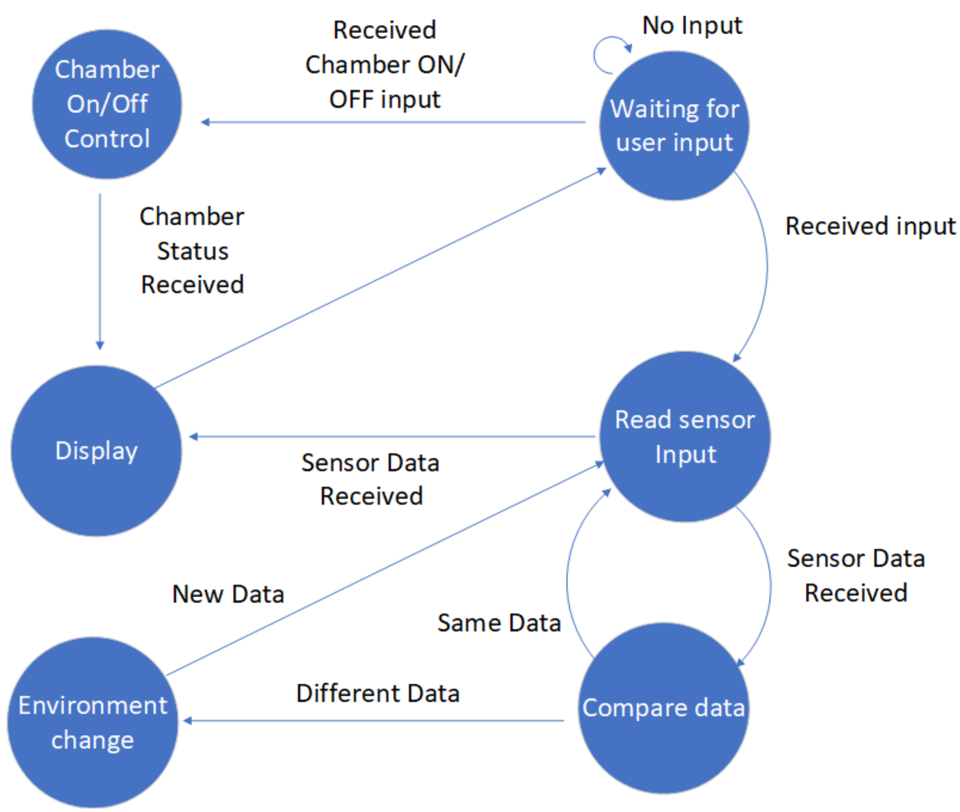
Buttons shall be the main input media from the user. There shall be four main buttons: up, down, back, and select. These buttons should be debounced through software handled by the slave raspberry pi, where an edge will be registered after a certain amount of time from the button press.

Temperature/Humidity sensor: In five-minute intervals, the master raspberry pi will accept data input from the sensors through GPIO ports, after calling the relay function that will activate the sensors. The data will only be recorded and handled in one chamber at a time. The raw data will be converted to Metric/Imperial units (according to a setting set by the user) and will be compared with the presets as given from the slave raspberry pi. From our initial testing, we noticed that DHT22 sensors can and will record “noise values”, which are spikes of high magnitude above or below the previous values. Therefore, the final code shall detect such values and retry the recording.

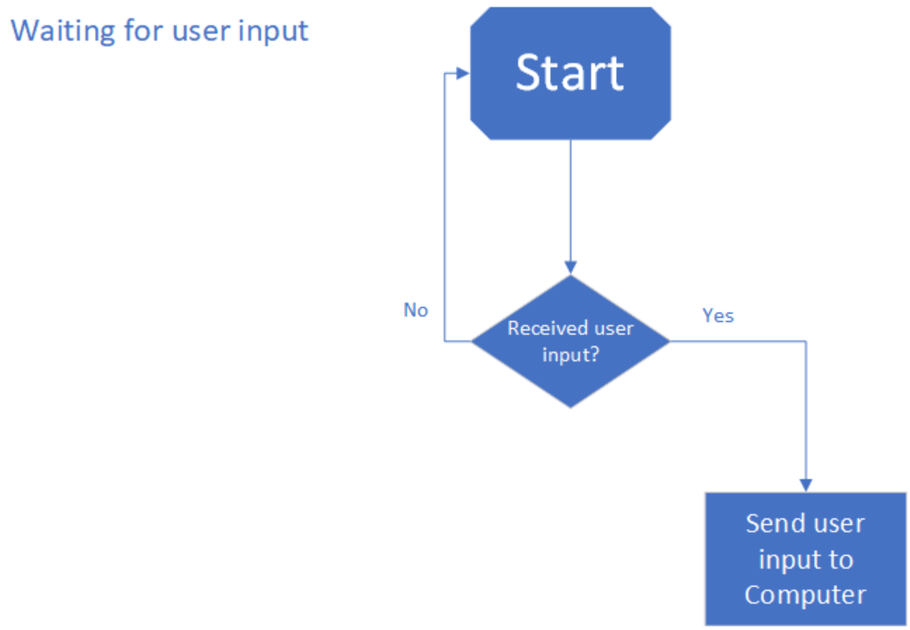
Solenoid Valves: After comparing the measured temperature with the preset of the chamber in question, certain logic will be operated: If the recorded temperature is less than the preset, the master will contact the slave to display an error message (this is a point where the system would require human interference). If the recorded temperature is within the preset boundaries, the temperature data will be sent to the slave which will display it. If the recorded temperature is higher than the preset, the master will actuate the relays needed to allow cold air to enter the chamber and send the temperature data to the slave.

Misters: After comparing the measured humidity with the preset of the chamber in question, certain logic will be operated: If the recorded humidity is greater than the preset, the master will contact the slave to display an error message (this is a point where the system would require human interference). If the recorded humidity is within the preset boundaries, the humidity data will be sent to the slave which will display it. If the recorded humidity is lower the preset, the master will actuate the relays needed to activate the misters and send the humidity data to the slave.

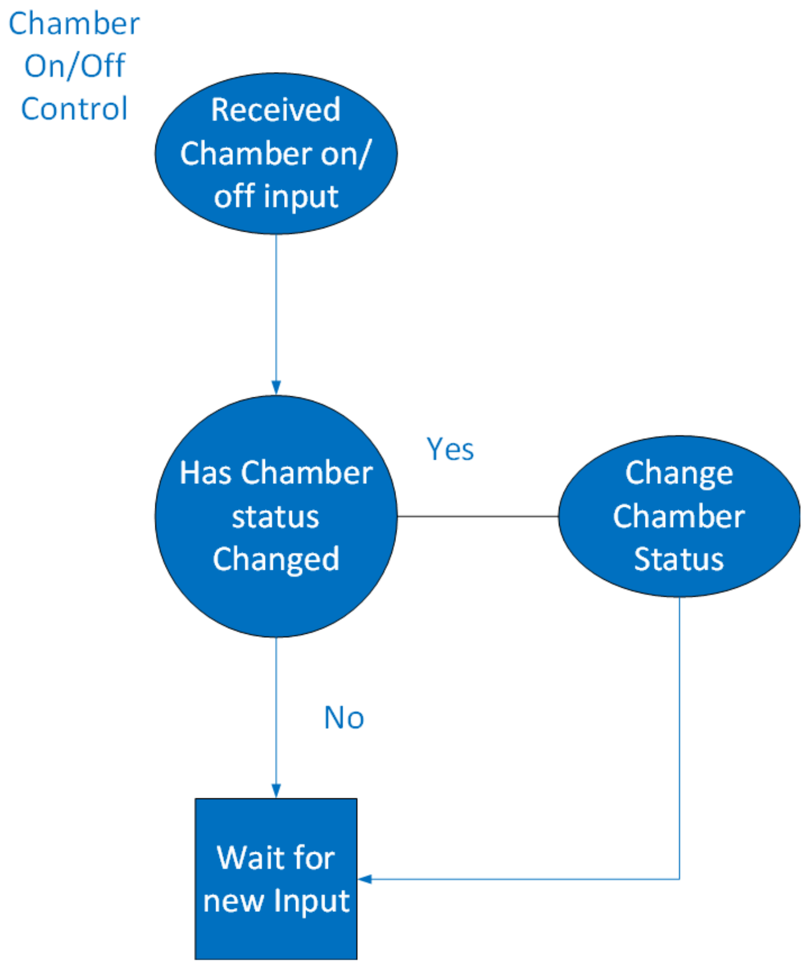
C) Include flow diagrams with identifications of subroutines and main parameters for simulation projects.



This state diagram shows the 6 subroutines which are broken down into the following flowcharts.

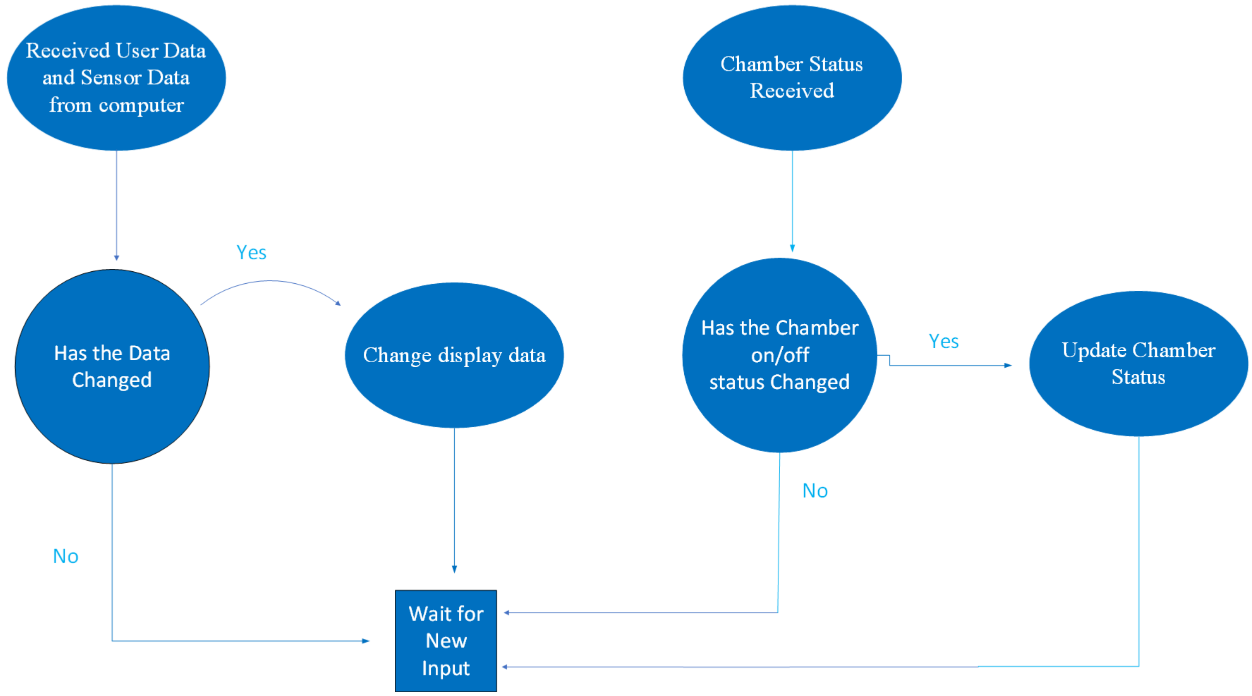


Once it starts, the waiting for user input subroutine toggles the received user input parameter. If no input is detected then the subroutine just cycles back to start which then goes back to received user input in an eternal loop until an input is detected. After a user input is received, that signal is sent to the processing module computer.

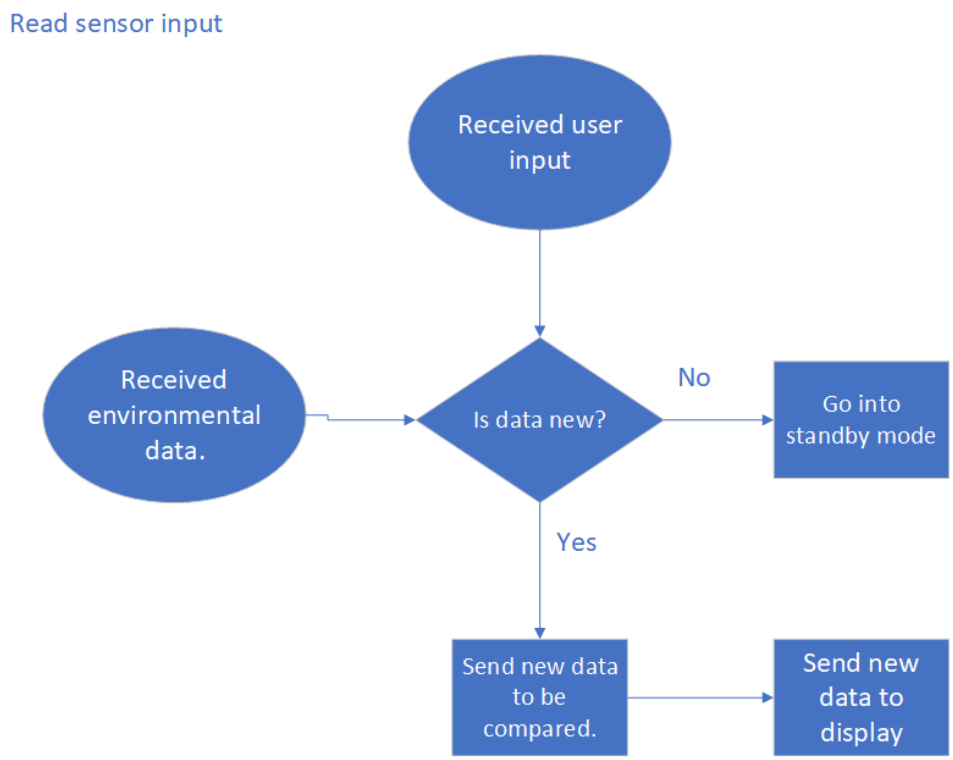


After receiving a chamber on/off input request, the Chamber On/Off Control subroutine will determine if the chamber status needs to change. If a change is required, the chamber control will turn off or on the chamber that was requested to change before it waits for new input mode. On the other hand, if the chamber status does not need to change it just waits for new input mode.

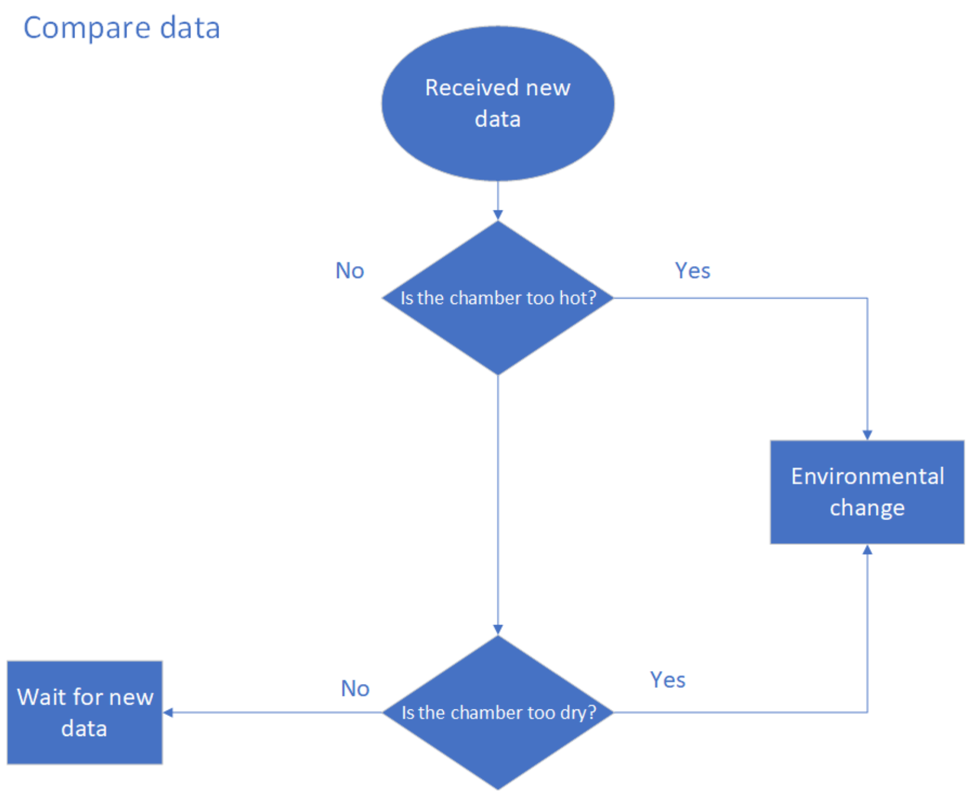
Display



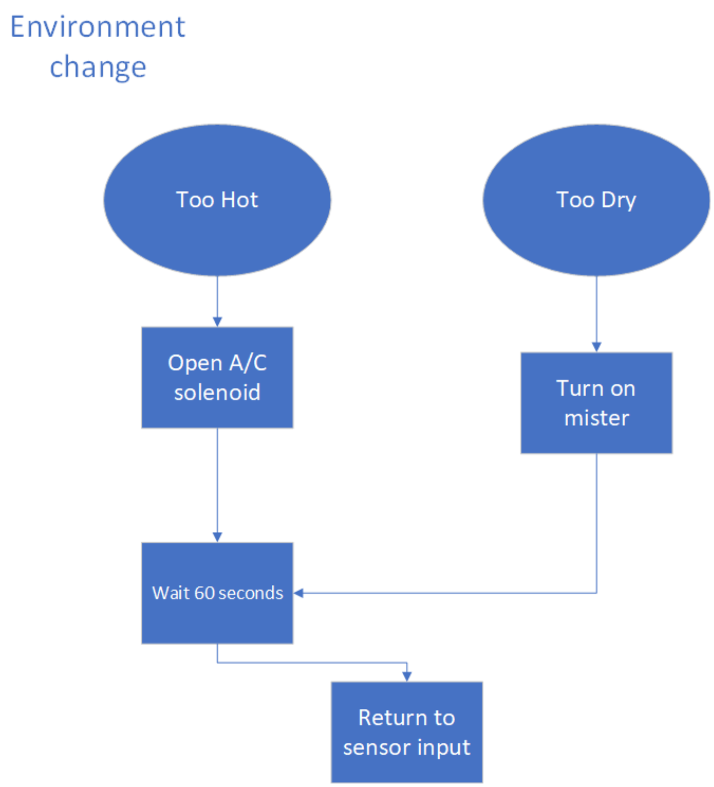
The Display subroutine after receiving the sensor data from the processing module computer determines if the data has changed. If it has then it changes the display data and then goes to waiting mode till data is received again, but if the data has not changed then it just goes to waiting mode till data is received again. The subroutine also receives a chamber status signal from the chamber of/off control. If the status has changed then the display will update the chamber status and then go to wait for new input mode. If the status has not changed then it just goes to wait for new input mode.



The Read sensor input subroutine receives user input and then checks to see if the data is new. If the data is new then the environmental data is then sent to the compare data subroutine and to the display subroutine. If data has not changed, then subroutine goes into standby mode.



The compare data subroutine receives the new data from the read sensor input subroutine and determines if the chamber is to hot or dry. If it is, the data is sent to the environmental change subroutine. If the data is not too hot or dry, then the subroutine goes into waiting mode till new data is sent.



In the Environment change subroutine, a too hot or too dry signal will appropriately open the Solenoid and/or turn on the mister. It will then wait for 60 seconds then close the solenoid and turn off the mister. After the subroutine returns to the read sensor input subroutine.

Prototype Progress report

Acquired components:

Refrigerator

Raspberry Pi 3B+

DHT22 and DHT11 temp sensors

Character LCD screen

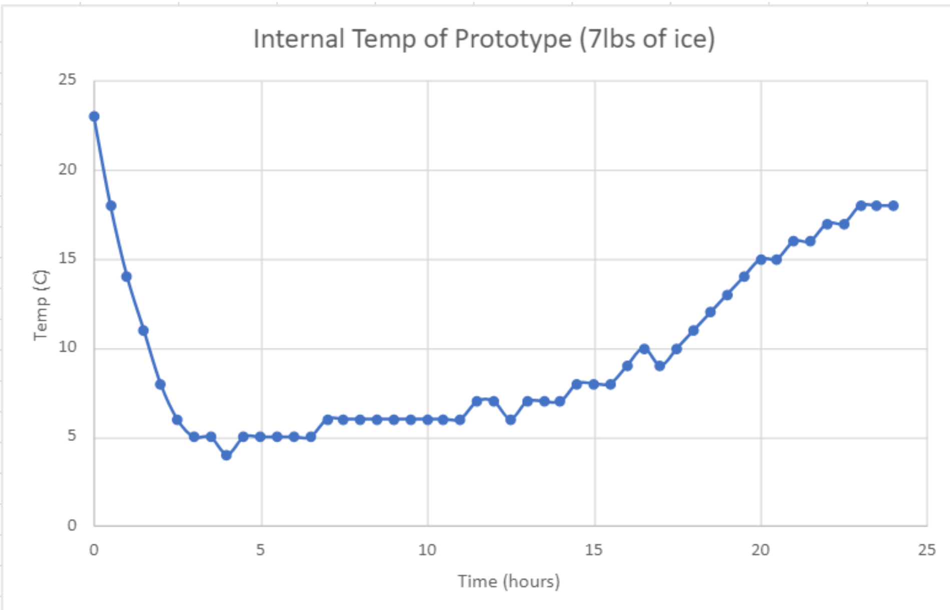
Note: We are currently working with the Patriot Green Fund to apply for funding for our project. When the application process is complete, we will be able to purchase more components.

Experimental progress:

Experiment 1:

We carried out our second test by using a polystyrene cooler as a model of a chamber and filled it with a 7lb bag of ice in a room temperature environment (23 C). Our goal was to test the viability of similar materials in our chambers and to get an idea of how rapidly temperatures in the chamber would rise.

Results of Experiment 1:



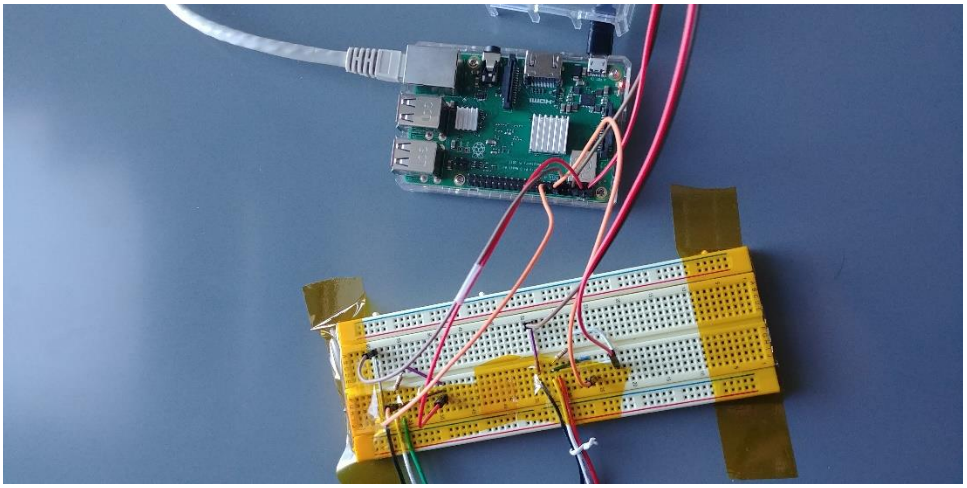
The initial internal temperature of the cooler was 23 degrees Celsius. In the first few hours, it cooled to 5 degrees Celsius. After 24 hours, the internal temperature of the cooler was still below the starting temperature. Overall, this experiment was a success. However, the DHT11 sensor data needs to be validated with a thermometer.

Experiment 2:

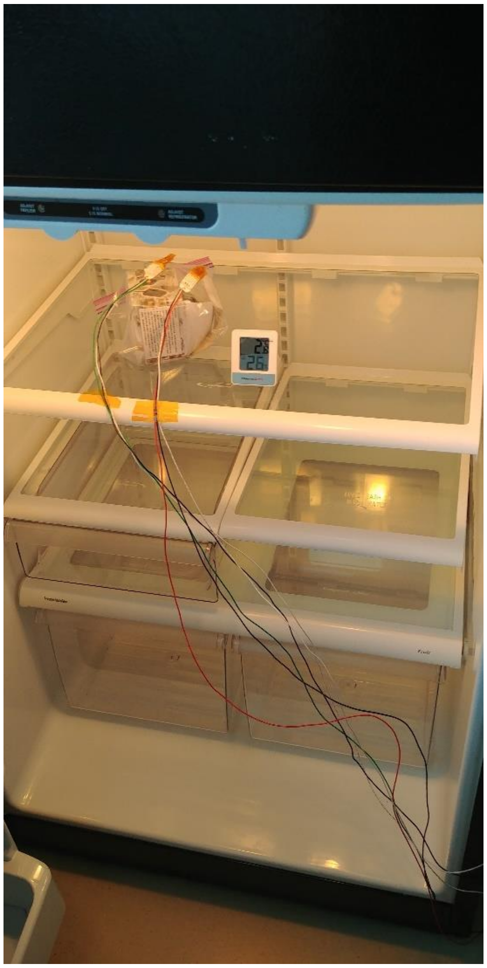
Results of Experiment 2.a:

The purpose of experiment 2.a is to test the functionality of new DHT22 code while also testing the performance of the DHT22 sensor at different VDD voltages. The test was run as follows.

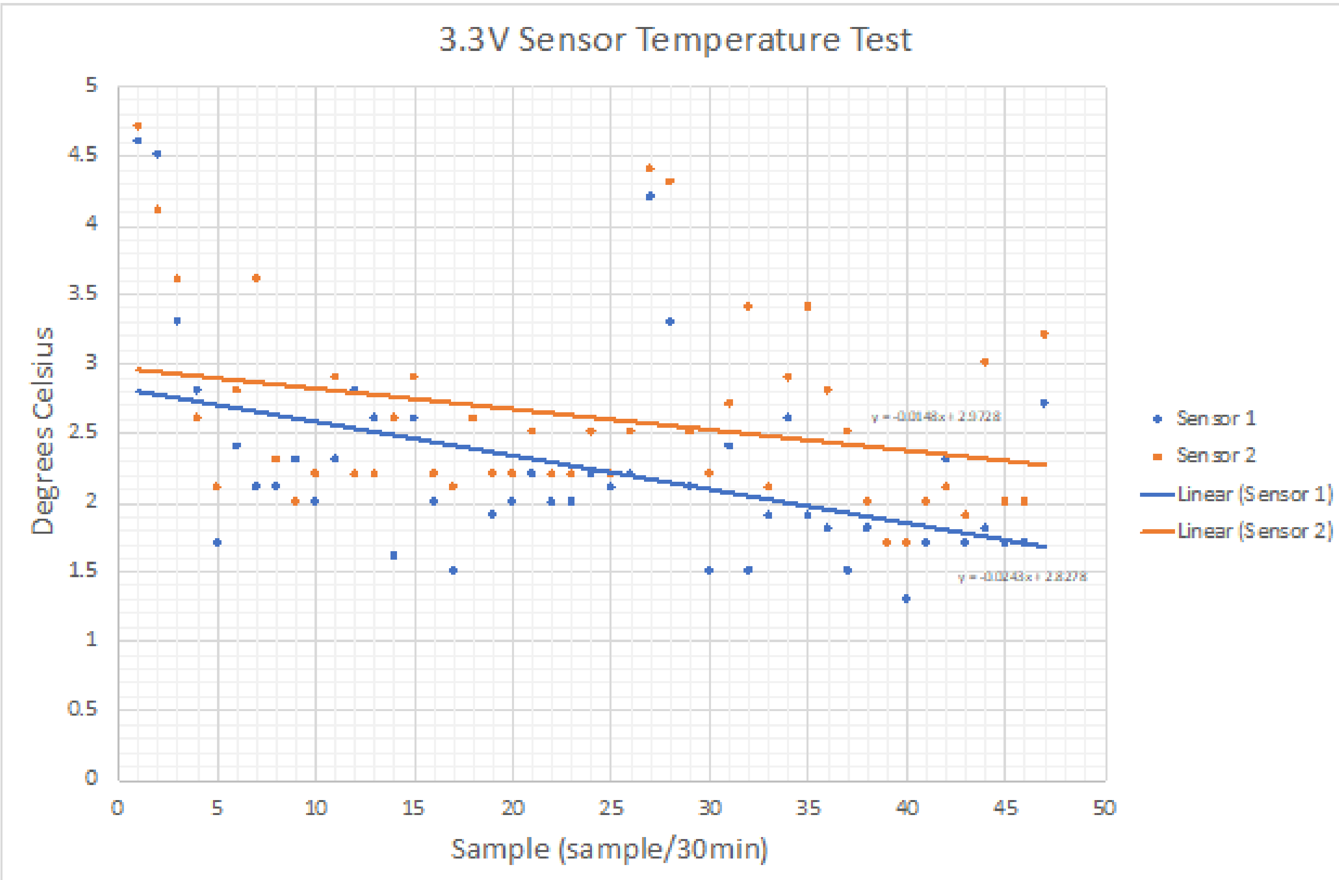
- A breadboard with 2 DHT22 circuits was set up, and VDD was set to 3.3V.



- Two DHT22 sensors were placed below the refrigerator cold air blower in such a way that they were suspended and not physically in contact with any surface. A Thermapro temperature and humidity sensor was also placed nearby to provide data for future percent error calculations (see Tables 9.1 & 9.2).



- The new DHT22 python code was initiated and left to collect data for 24 hours. Data was sampled at a rate of one sample every 30min resulting in 48 temperature data points and 48 humidity data points. The results of the test are shown below.



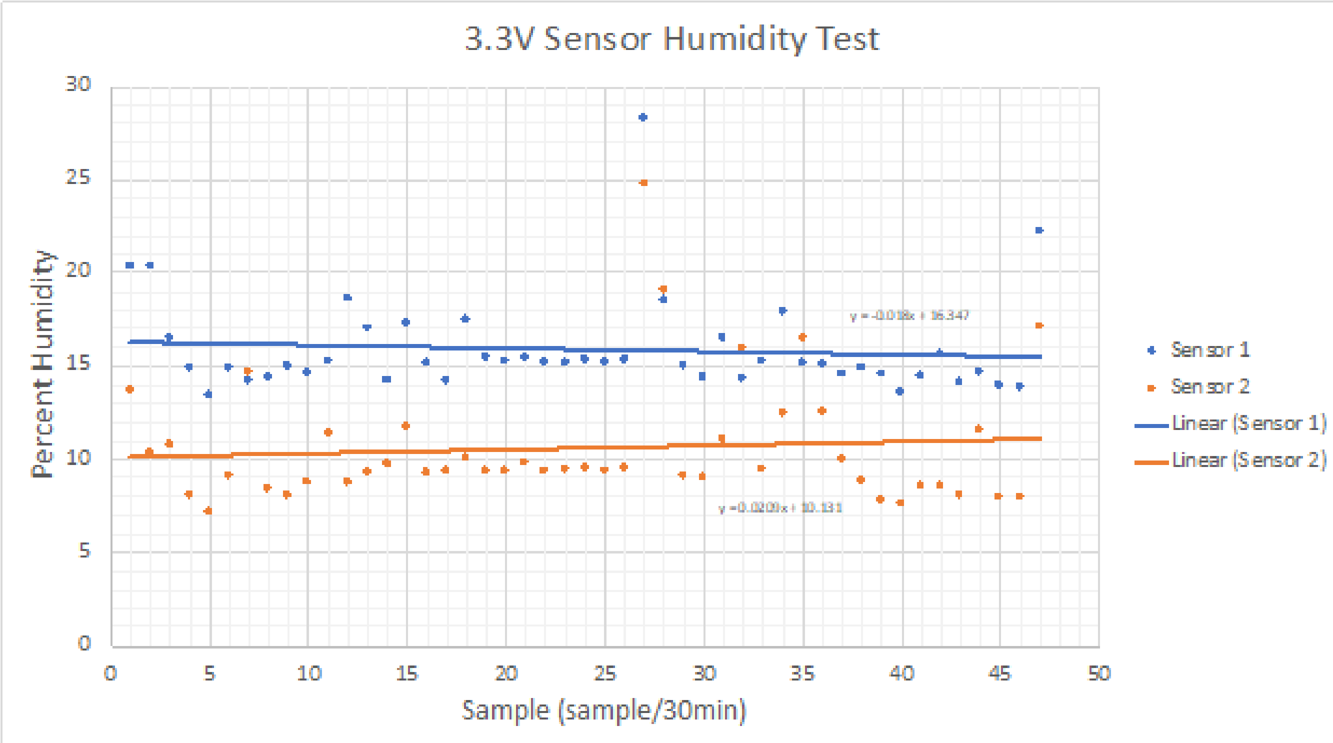
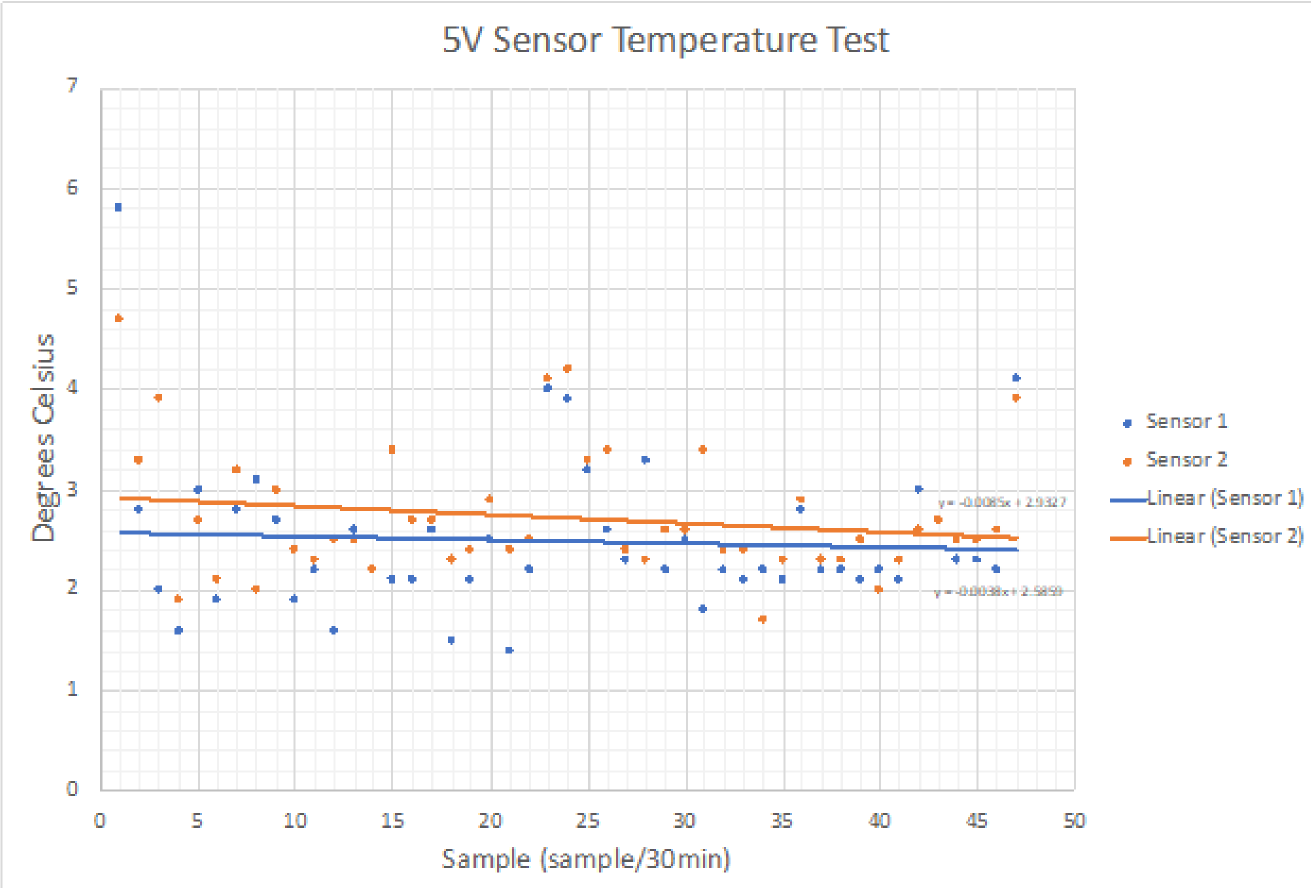


Table 9.1
3.3V Test Percent Error Calculations

Sample	Temp % Error 1	Humidity % Error 1	Temp % Error 2	Humidity % Error 2
1	28.125	36.25	26.5625	57.1875
46	30.76923077	44.4	53.84615385	68
47	58.82352941	28.38709677	88.23529412	44.83870968

** Samples for percent error calculations taken at the beginning and end of the test

- The results 2a yielded unsatisfactory percent error values. It was hypothesized that the error was due to the volatile environment present within the refrigerator as its environment is modified every time the door is opened. The VDD was changed to 5V and the script was run again.


5V Sensor Temperature Test
Degrees Celsius
7
6
5
4
3
2
1
0
0
5
10
15
20
25
30
35
40
45
50
Sample (sample/30min)
y = -0.0085x + 2.9327
y = -0.0038x + 2.5859
Sensor 1
Sensor 2
Linear (Sensor 1)
Linear (Sensor 2)

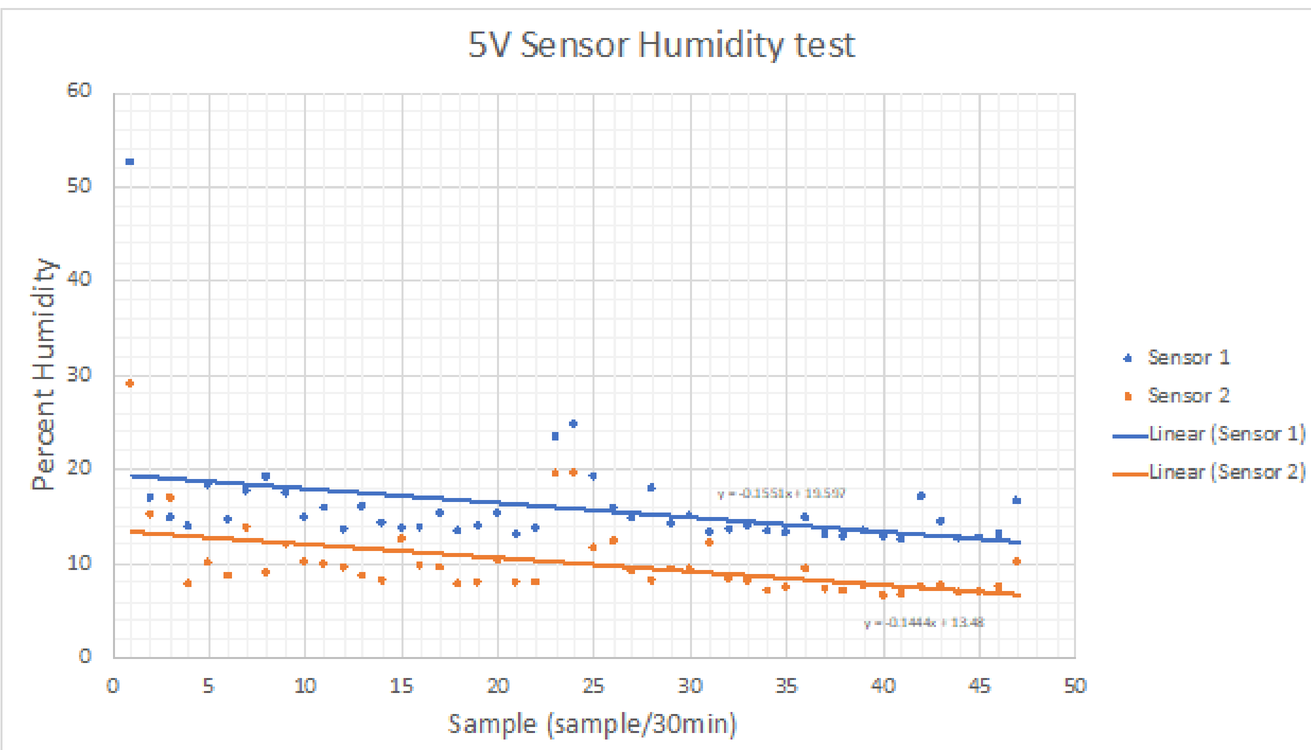

5V Sensor Humidity test
Percent Humidity
60
50
40
30
20
10
0
0
5
10
15
20
25
30
35
40
45
50
Sample (sample/30min)
y = -0.1551x + 19.597
y = -0.1444x + 13.48
Sensor 1
Sensor 2
Linear (Sensor 1)
Linear (Sensor 2)

Table 9.2
5V Test Percent Error Calculations

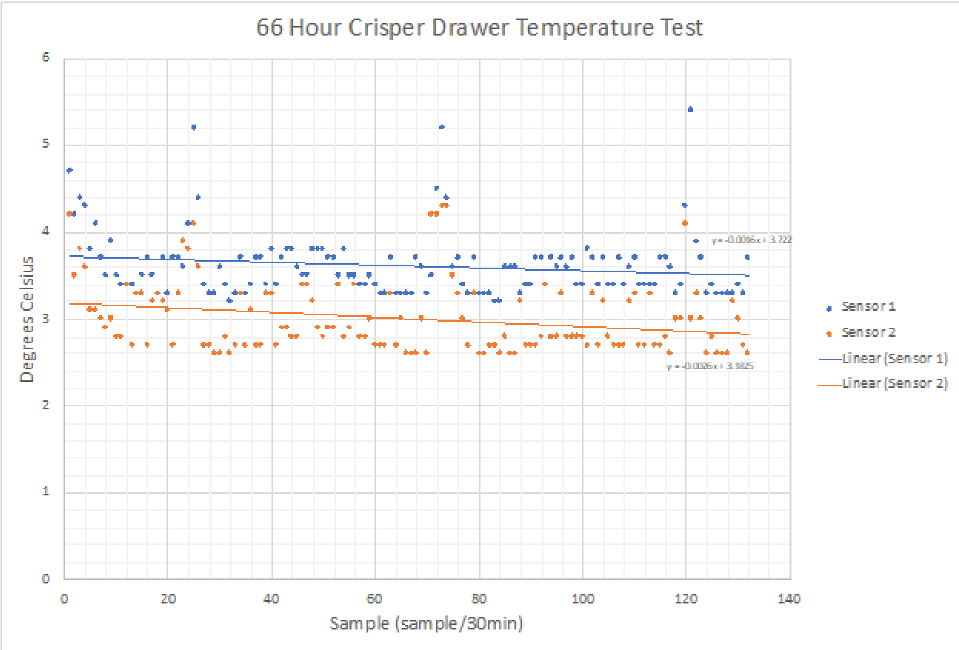
Sample	Temp % Error 1	Humidity % Error 1	Temp % Error 2	Humidity % Error 2
1	87.09677419	13.91304348	51.61290323	37.17391304
42	42.85714286	32	23.80952381	70.4
46	57.14285714	48	85.71428571	70.8
47	28.125	25.45454545	21.875	54.54545455

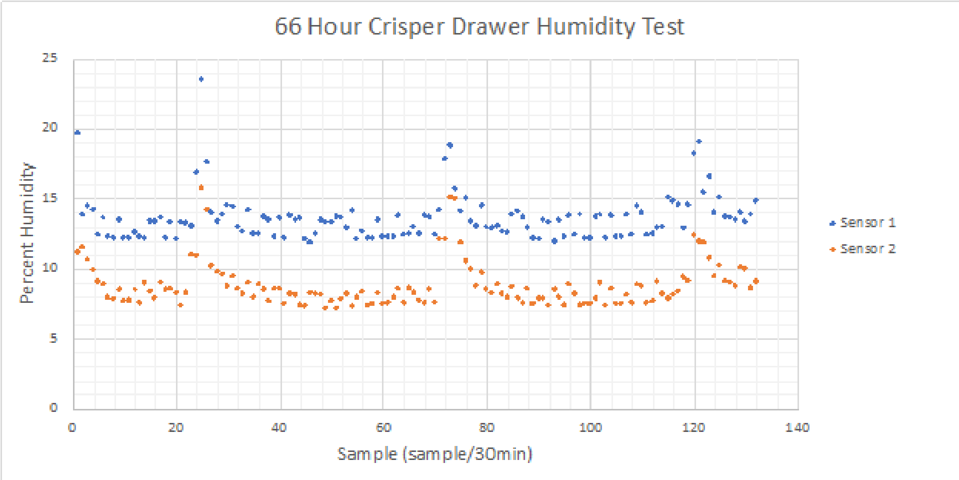
** Samples for percent error calculations taken at the beginning and end of the test

- The results of the second run were equally unsatisfactory, and there was no noticeable improvement from the increase in VDD.

Results of Experiment 2.b:

The purpose of experiment 2.b was to record the environment within the stock refrigerator crisper drawers. The temperature and humidity data were logged and displayed below. As expected, the temperature and humidity data demonstrated parabolic behavior as the refrigerator allows the temperature to increase to a maximum threshold temperature before resuming the cooling cycle. The increase in temperature allows for the humidity to increase within the drawers thus exhibiting the parabolic trend seen below.

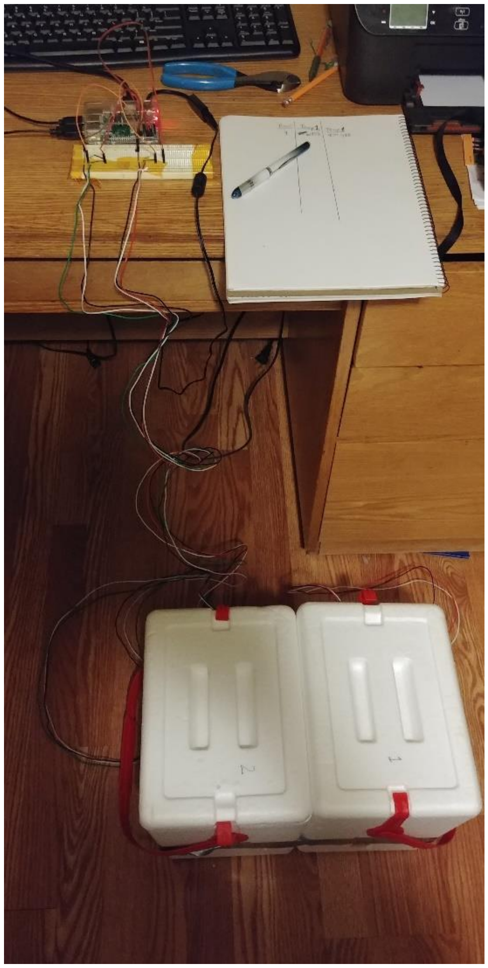




Results of Experiment 2.c:

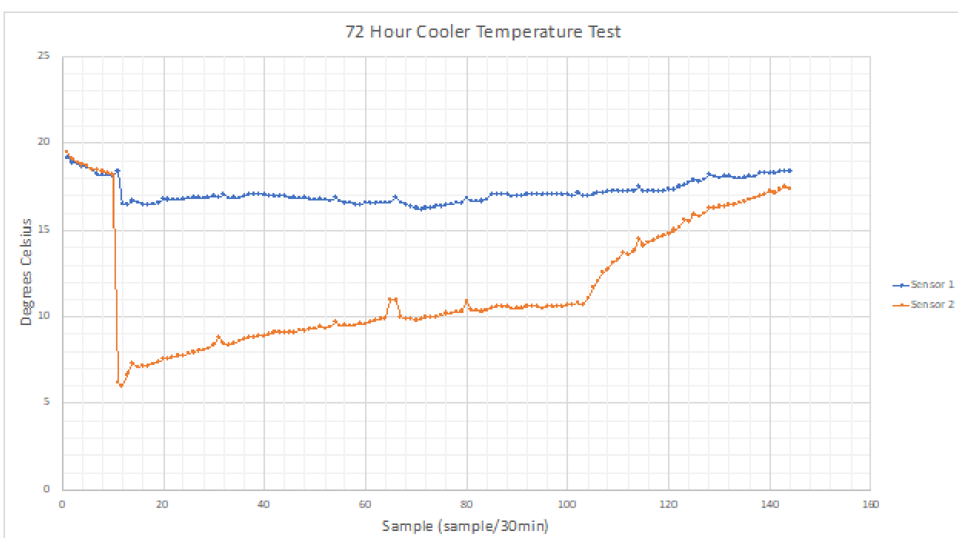
The purpose of experiment 2.c was to replicate the results of experiment 1 with the new Python3 code and DHT22 sensor. Additionally, the experiment was intended to demonstrate the robustness of the new program and performance of the DHT22 in a stable environment for a prolonged period. Multiple hand measurements were taken to test the sensors' accuracy as well. The experiment was run as follows.

- The original circuit and DHT22 setup from 2.a were implemented once again at 5v.
- Two 1", 11 Liter coolers were setup side-by-side to have a 2" insulating barrier between the two chambers. The experiment will also illustrate if more insulation will be required in future prototypes.



- The program was initiated, and cooler 2 was filled with 5 liters of ice after the 10th data point was collected. The setup was left running in a stable environment with an exterior temperature of 18.3 degrees Celsius.

- During the 72hr run manual data recordings were taken at random intervals. The results and percent error calculations can be seen below.



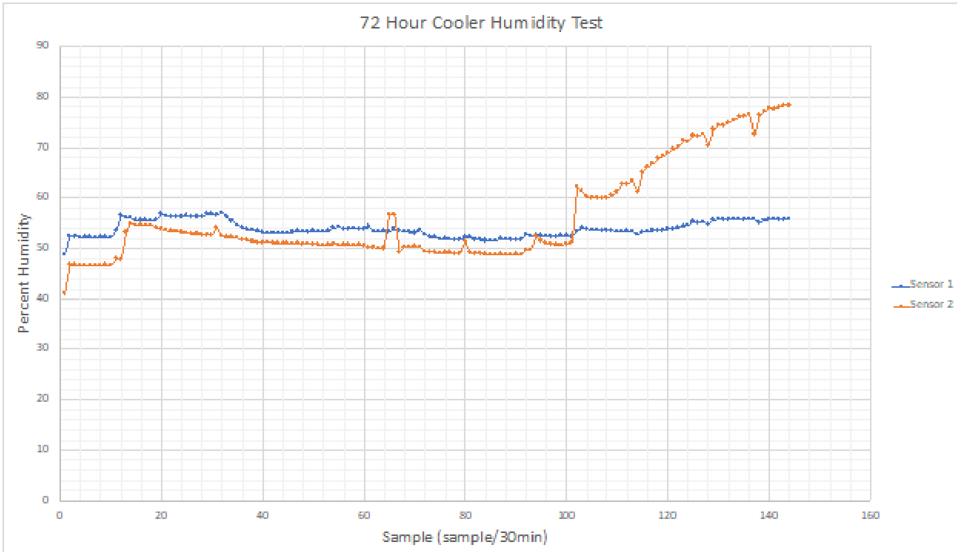


Table 9.3

Cooler Test Percent Error Calculations

Sample	Temp % Error 1	Humidity % Error 1	Temp % Error 2	Humidity % Error 2
9	0.55	11.06	1.08	4.00
18	3.51	6.73	9.88	4.21
30	2.30	5.37	4.55	6.07
52	1.75	4.90	1.06	7.82
64	1.78	4.71	0.00	9.09
78	1.78	3.60	0.00	9.07
90	1.73	3.60	0.00	9.44
100	1.16	3.14	0.94	8.93
112	0.57	4.90	2.16	7.79
126	1.66	8.04	2.47	7.31
136	0.55	7.31	1.18	6.46
144	0.00	7.31	2.25	8.95

** Manual samples for percent error calculations taken at random intervals during the testing period

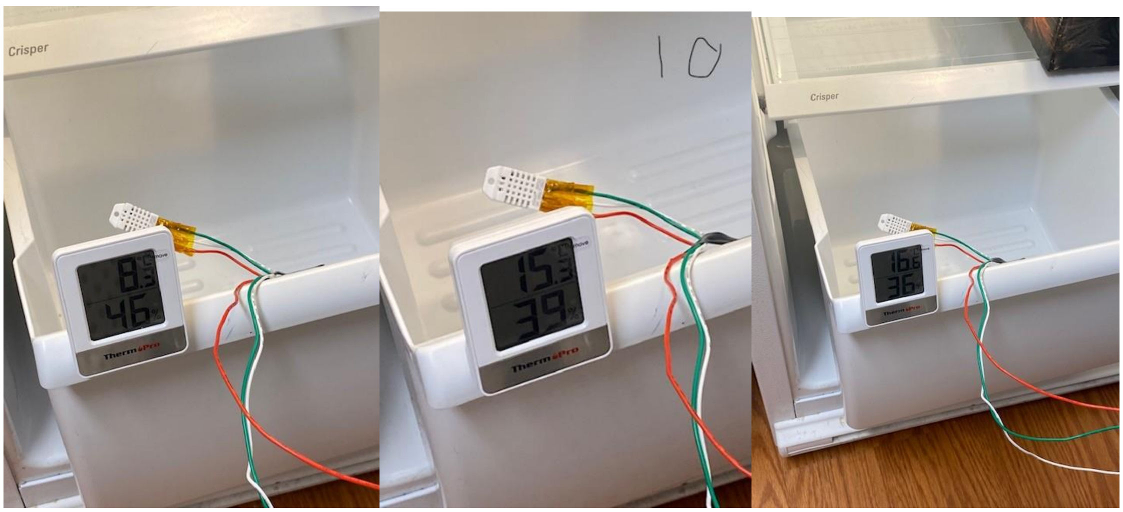
The 2" insulation thickness proved to be sufficient with only minimal change in temperature and humidity seen within the empty chamber 1, additional air gaps between the chambers can further improve performance.

In a stable environment the DHT22 demonstrated an increase in accuracy and yielded much improved percent error numbers. This can be improved upon through increasing sampling rate, detecting and replacing noise values with more accurate values.

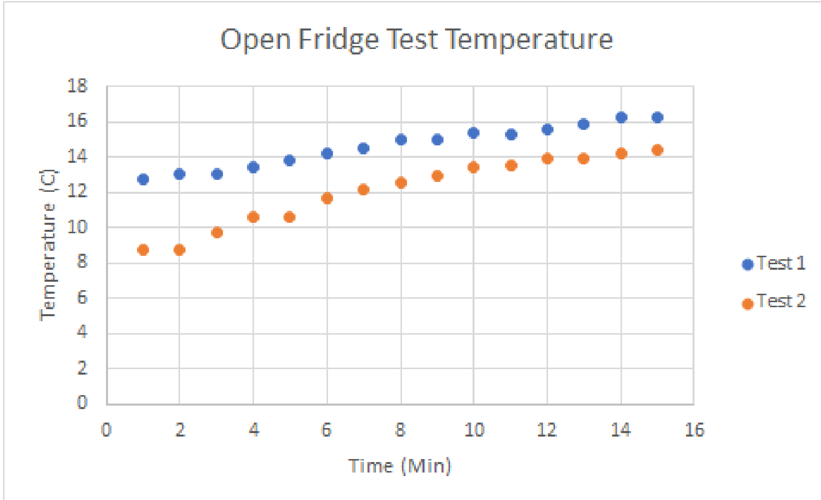
Experiment 3: Rate of change in environment with fridge door open with crisper drawer open

With this experiment, we were to find the difference in temperature over time with both the fridge open and drawer open to see the effect it had on both the DHT22 and the digital reader thermometer/humidity reader. The placement of the sensor the digital reader would be where the environment of the fridge and the room would then affect the readings. This test would last about 15 minutes.

The pictures below are how the experiment was done for test 1. These pictures (left to right) were taken at 1 minute, 10 minutes, and 15 minutes. The digital reader at 1 minute was off due to being in the fridge prior to the start and then over time, converted to the values in the DHT22 had.



Results of Experiment 3:



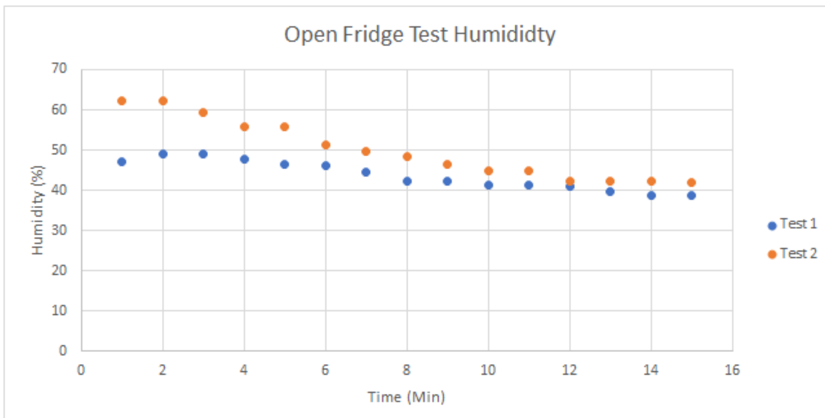


Table 10.1

Experimental vs Theoretical Temperature Test 1

Minute	DHT22 Temp (C)	DR Temp (C)	Aproximation Error (%)
1	12.7	8.3	53.01
10	15.4	15.3	0.65
15	16.2	16.6	2.41

Table 10.2

Experimental vs Theoretical Humidity Test 1

Minute	DHT22 Humidity (%)	DR Humidity (%)	Aproximation Error (%)
1	47.2	46	2.61
10	41.4	39	6.15
15	38.6	36	7.22

Table 10.3
Experimental vs Theoretical Temperature Test 2

Minute	DHT22 Temp (C)	DR Temp (C)	Aproximation Error (%)
1	8.7	7.4	17.57
7	12.2	12.8	4.69
15	14.4	15.5	7.10

Table 10.4
Experimental vs Theoretical Humidity Test 2

Minute	DHT22 Humidity (%)	DR Humidity (%)	Aproximation Error (%)
1	62.1	40	55.25
7	49.6	41	20.98
15	42	36	16.67

Results show that the sensor being nearby two separate environments caused a change in the reading compared to the previous tests where the environment was more closed in. The DHT22 sensor was able to get good temperature measurements when compared with the digital reader. The humidity results of this experiment tell us for this project, placement of the sensors is important to avoid any errors in the readings. With the points for Test 2 at the 5th data plot, that ended up being an error in the results with its value being in the negative temperature and giving a humidity over 100%. For the plot, it is the same as the 4th data plot. To avoid this issue next time, will use the method explained in the component section. This experiment showed the best option would be to place it near the solenoid valve and humidifier to have better readings at the of our containers for the fruits and vegetables.

Experiment 4: Two separate container environments

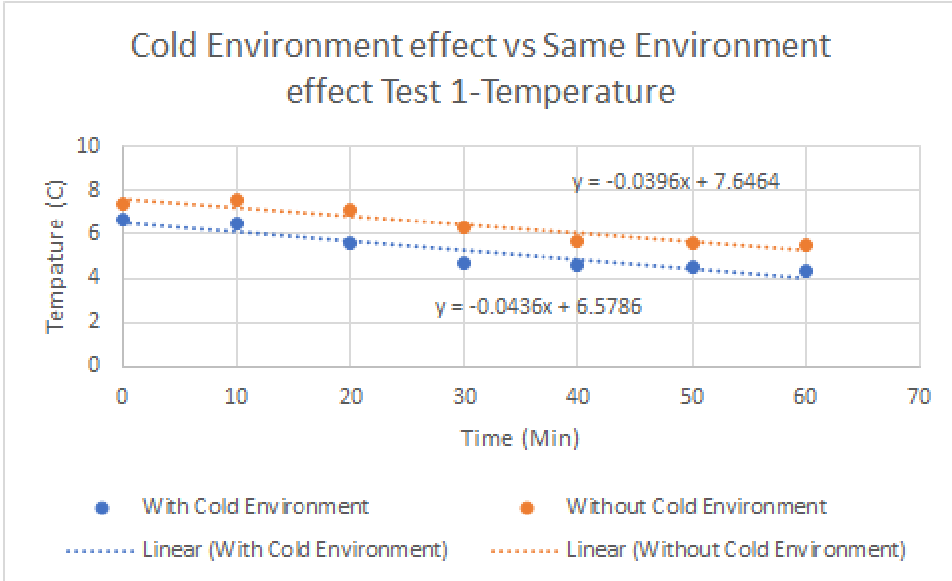
For this experiment, it was to see the effects of a crisper drawer containing a tray filled with ice on an empty, adjacent crisper drawer containing the sensor and a digital reader. To see if a change in temperature made an effect on the sensor drawer, we will compare the container with no ice and the container with ice to see an actual effect. The digital readers and sensors were exposed to the room environment long enough to notice a change in temperature and humidity when put into the fridge. Then for an hour, it measures the change in temperature and humidity. The point of this experiment is to see if an environment influences another environment.

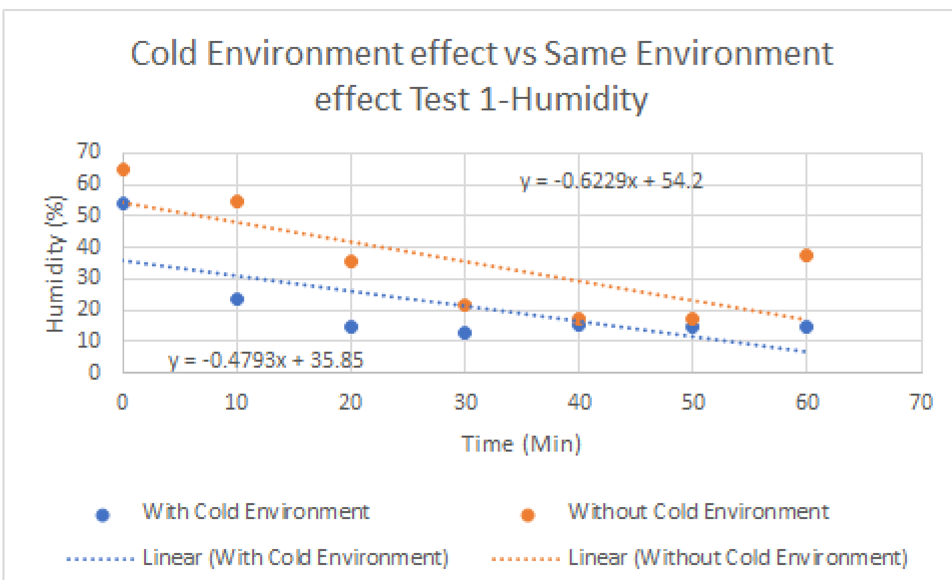
The pictures shown below is how the experiment was done for experiment 2 with ice. On the left shows the temperature when it started at 0 minutes and the right pictures shows after an hour. The sensor is reading temperatures in the left crisper drawer.

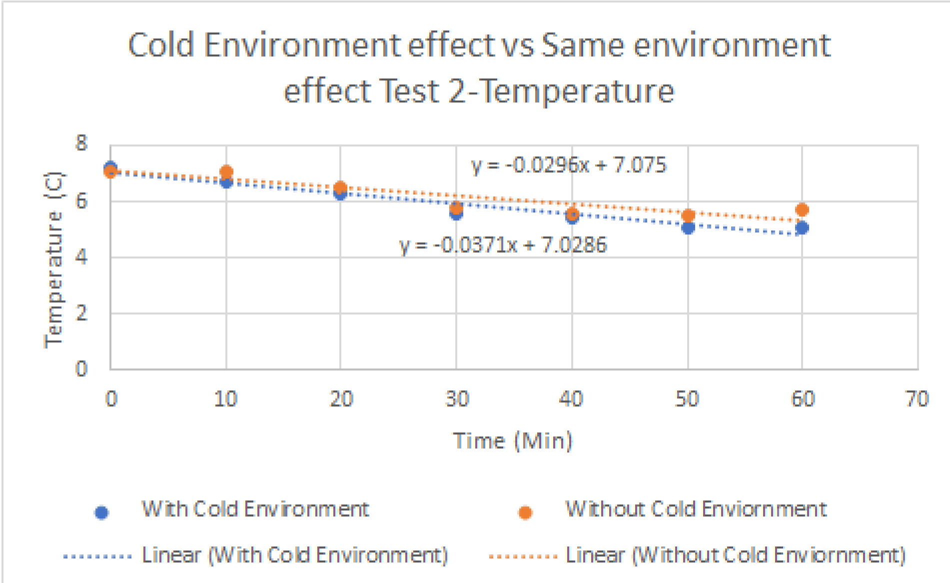

The pictures shown below are for experiment 2 with no ice. The left picture shows at 0 minutes and the right picture shows at 40 minutes. The left side is with the sensors the right side is with no ice


Results of Experiment 4:

The figures below show the results of the tests of the experiment. Note for the intercept, the equation is for the “Without Cold Environment” and Bottom is for “With Cold Environment”.


Cold Environment effect vs Same Environment effect Test 1-Temperature
Tempature (C)
10
8
6
4
2
0
y = -0.0396x + 7.6464
y = -0.0436x + 6.5786
0
10
20
30
40
50
60
70
Time (Min)
With Cold Environment
Without Cold Environment
Linear (With Cold Environment)
Linear (Without Cold Environment)


Cold Environment effect vs Same Environment effect Test 1-Humidity
Humidity (%)
70
60
50
40
30
20
10
0
y = -0.6229x + 54.2
y = -0.4793x + 35.85
0
10
20
30
40
50
60
70
Time (Min)
With Cold Environment
Without Cold Environment
Linear (With Cold Environment)
Linear (Without Cold Environment)



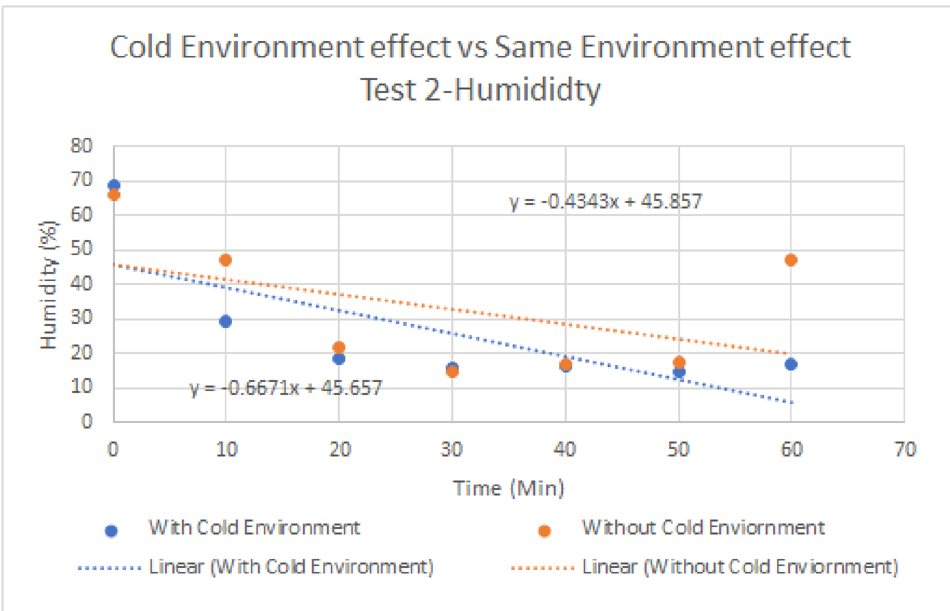


Table 11.1
Experimental vs Theoretical Temperature, With Ice Test 2

Minute	DHT22 Temp (C)	DR Temp (C)	Aproximation Error (%)
0	7.2	8.8	18.18
60	5.1	4.5	13.33

Table 11.2
Experimental vs Theoretical Humidity, With Ice Test 2

Minute	DHT22 Humidity (%)	DR Humidity (%)	Aproximation Error (%)
0	68.6	47	45.96
60	16.1	22	26.82

Table 11.3

Experimental vs Theoretical Temperature, Without Ice Test 2

Minute	DHT22 Temp (C)	DR Temp (C)	Aproximation Error (%)
0	7.1	7.5	5.33
40	5.6	4.8	16.67

Table 11.4

Experimental vs Theoretical Humidity, With Ice Test 2

Minute	DHT22 Humidity (%)	DR Humidity (%)	Aproximation Error (%)
0	65.80	45.00	46.22
40	16.70	22.00	24.09

The results of the experiment show that temperature and humidity sensor is not affected by the change in environment from an adjacent compartment. The rate of change for temperature is close and shows that there is little to no change with the different environment and with a same environment. At points, the humidity with no ice and with ice almost match to be the same except for a few times. For unknown reasons it makes spikes in its values and has greater approximation errors as a result. This will be addressed by making code to only allow act on values that are not obviously inaccurate.

Testing Plan

Testing will be done in Four main experiments, but equipment and software testing will happen throughout the building phase to make sure both software and equipment works as intended without being connected to other devices first. Testing individual functions is necessary to ensure components turn on, off, open, or close depending on the software and device being tested.

Experiment #1 (Mission Requirement Evaluation)

Goal: Measurement of Decay in contents of Crisper.

System Components: Completed Fridge, Measuring Devices, DC

Testing Process:

- After turning on and selecting crisper number and contents from list, measuring devices will measure how much vitamin A and other decaying elements decrease over the testing period.
- Data collection from devices will be done every 24 hours for each food item in crisper being tested for 1-2 weeks.
- Data collection will be done twice for 2-4 fruit types and 2-3 vegetable types.

Data Processing and Visualization: The data will be plotted onto a graph with datapoints for each food.

Evaluation: Data will be compared to average decaying times of food items. Will see if our fridge increases amount of days food stays fresh and how much longer they stay fresh.

Note: A test could be done with an Assortment of Fruits and Vegetables in each crisper. We can experiment with a crisper having a mix of fruits or vegetables if there is time.

Experiment #2 (Operational Requirement Evaluation)

Goal: To test that all devices work as supposed to once fridge is put together fully.

System Components: Completed Fridge, Measuring Devices

Testing Process:

- After turning on, we will measure current, voltage, and power into all devices.
- Will test the display and interface by going back and forth through compartment numbers and food lists using interface buttons and seeing feedback on display.
- Will test each compartment's ability to maintain a steady environment by choosing a food setting then analyzing the data from that compartment.
- Data collection will be from the sensor.
- Data collection will be done for 1-2 hours for each compartment taking temperate and humidity data every 10 mins.
- Data collection will be done 5 times for each compartment. Changing the user input in such a way that only one variable changes for 4 of the trials. For example, for trial 1 only temperature increases, but for trial 2 only humidity increases, and so on. The fifth trial though will be changing selected food items every 20 mins, thus changing the environment each time.

Data Collection: Measurment's will be recorded

Evaluation:

1. Verification that measurements match the desired values.
2. Verification that displays and interface works, by showing feedback of interface chooses and sensor data.
3. Verification that environment becomes steady after obtaining the desired environment.

Experiment #3 (Functional Requirement Evaluation)

Goal: To make sure that DHT sensor data is correct.

System Components: Fridge, Battery powered temperature and humidity sensor

Testing Process:

1. A comparison will be made between the DHT sensor data and a battery temperature and humidity sensor that will be placed in each compartment.
2. Data collection will be for 8 hours, with new user input every 2 hours. We will change either the temperature or humidity user inputs, but not both at the same time.
3. Sensor data will be sampled every 10 mins.
4. Data collection will be done twice for each crisper.

Data Processing and Visualization: Data collected from both sensors will be put in a graph to compare.

Evaluation: By focusing on data comparing DHT sensors to the battery powered sensor's temperature and humidity values, we can determine that the DHT sensors are working correctly.

Note: Testing time could be extended to be for 12 hours with changing times being every 3 hours if not enough data is being generated. May also need to change sensor data sampling times every 5mins instead of 10mins.

Experiment #4 (Function Error Requirement Evaluation)

Goal: To have processing module detect errors and either fix them by restarting device or shutting them down and telling user about error.

System Components: Fridge, Battery powered temperature and humidity sensor, PC, Timer

Testing Process:

- After turning on will induce error in the form of removing a DHT sensor to see how processing module handles not having a sensor.
- Data collection will be in the form of timing how long it takes an error message to show up on the Display.
- Will be done 3 times.
- Another error test will be keeping crisper door open for 2 hours.
- Data collection here will be by both sensors with sensor data being taken every 10 mins.
- At the same time will time how long error message takes to show on display.
- Data collection will be done 3 times.

Data Processing and Visualization: Raw data in the form of the display times will be record for removing sensor error. Raw data in the form of sensor data will be recorded alongside display times then plotted using a PC to see if error message sending times need to be adjusted in code for crisper door error.

Evaluation:

1) Will focus on Detecting times for errors.
2) Verify display error messages happen.

Tasks

Hardware Implementation:

The hardware team consists of Nestor, Matthew, and Michael, who will oversee the following tasks:

- Mounting the Components:
  - Fixing the components to their final position

    The components and their positions will first be measured to make sure they can be mounted safely. After they will be placed in their positions to see if any adjustments need to be made before mounting and installing begins.
    - Responsible: Nestor
  - Soldering components to the data and/or power cables

    All soldering of components will happen before installation of those components into their compartments.
    - Responsible: Nestor

- Mechanical modifications:
  - ~~Drilling into the fridge~~

    ~~Drilling holes into a fridge requires careful consideration. To avoid cutting into components, drilling a hole into the side of the fridge is the best option. The drill will be set to the slowest setting and after cutting each layer we will do a visual check for damageable components.~~
    - ~~Responsible: Matthew~~

  Update: We will not perform drilling onto the fridge due to there being too much of a risk in drilling through the fridge, possibly damaging a water/coolant line in the fridge. The only change in the design would be wire management and not anything performance related. [Edited by Cameron Flores, Sustainable Fridge Team, January 2022]
  - Making A/C tube connections

    The tubing shall be run out of the box that is collecting the cool air and into each compartment. To make the build process easier, the tubing will be attached to the compartments before placing the compartments in the fridge.
    - Responsible: Matthew

  3. Installing drain system.

    Installing the drain system will be done in a comparable way as the A/C tube connections but with smaller tubes. The drain tubes will be prefabricated to the compartments before placement.
    - Responsible: Matthew
- Wiring:
  - Cable management

    The cables shall be enclosed within a liquid tight conduit tubing to keep it both protected and organized. They will have no sharp bends in the tubing for both safety and so that the wires will not be at risk of being pinched.
    - Responsible: Michael

~~Early in the semester, the hardware team will attempt to locate a refrigerator from the transfer station to practice drilling, wiring, and mounting. This is to help ensure that no mistakes will occur when working on our final system.~~

Update: See update on page 51. [Edited by Cameron Flores, Sustainable Fridge Team, January 2022]

Software Implementation:

- Code at the external I/O level:
    - LCD will be the main output media to the user. There will be a main screen, displaying a representation of each chamber in list view, as well as a settings bar at the bottom of the list. The user will use buttons to navigate through this main screen, then for each chamber selected, the user will be presented with a list of presets to select according to the type of produce needed to be stored in the chamber in question. The user will also be presented with an option to disable a certain crisper or re-enable it when they see fit to do so. Otherwise, if no produce is in a crisper or on/off option selected than the crisper will automatically be set to a predetermine basic environment. The settings bar will provide the user options including changing Display Brightness and changing units between metric and imperial.
        - Responsible: Cameron, Zach, Ahmed
    - Buttons shall be the main input media from the user. There shall be four main buttons: up, down, back, and select. These buttons should be debounced through software handled by the slave raspberry pi, where an edge will be registered after a certain amount of time from the button press.
        - Responsible: Ahmed
- Code at the relays level:
    - Temperature/Humidity sensor: In five-minute intervals, the master raspberry pi will accept data input from the sensors through GPIO ports, after calling the relay function that will activate the sensors. The data will only be recorded and handled in one chamber at a time. From our initial testing, we noticed that DHT22 sensors can and will record "noise values", which are spikes of high magnitude above or below the previous values. Therefore, the final code shall detect such values and retry the recording. The raw data will be converted to Metric/Imperial units (according to a setting set by the user) and will be compared with the presets as given from the slave raspberry pi.
        - Responsible: Ahmed
    - Solenoid Valves: After comparing the measured temperature with the preset of the chamber in question, certain logic will be operated: If the recorded temperature is less than the preset, the master will contact the slave to display an error message (this is a point where the system would require human interference). If the recorded temperature is within the preset boundaries, the temperature data will be sent to the slave which will display it. If the recorded

temperature is higher the preset, the master will actuate the relays needed to allow cold air to enter the chamber and send the temperature data to the slave.

- Responsible: Cameron

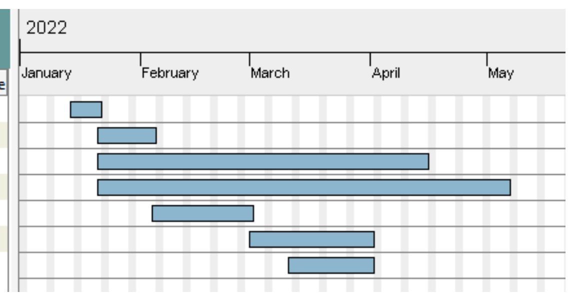
- Misters: After comparing the measured humidity with the preset of the chamber in question, certain logic will be operated: If the recorded humidity is greater than the preset, the master will contact the slave to display an error message (this is a point where the system would require human interference). If the recorded humidity is within the preset boundaries, the humidity data will be sent to the slave which will display it. If the recorded humidity is lower the preset, the master will actuate the relays needed to activate the misters and send the humidity data to the slave.
  - Responsible: Zach

Milestones

- Obtained Fridge
- Created 2 models of prototypes
- Did multiple experiments

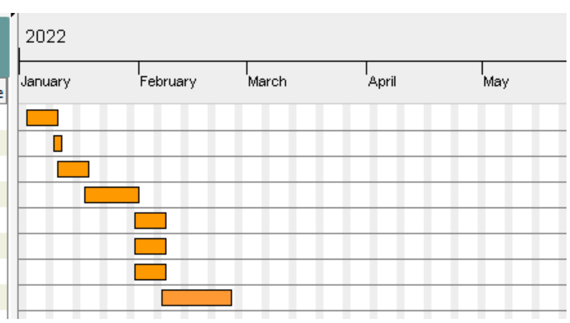
General Schedule

Name	Begin date	End date
Week 0, Review Information and Tasks	1/14/22	1/21/22
Project Title Form, Project Description/Abstract, Test Plan and WBS delivery	1/21/22	2/4/22
Rough Draft Final Report	1/21/22	4/15/22
Oral Presentation, Final Report Delivery, Project Poster Delivery	1/21/22	5/6/22
Mount components onto + Build Compartments	2/4/22	3/1/22
Testing, Troubleshooting	3/1/22	4/1/22
In Progress Presentation and Review, Progress Report #2	3/11/22	4/1/22

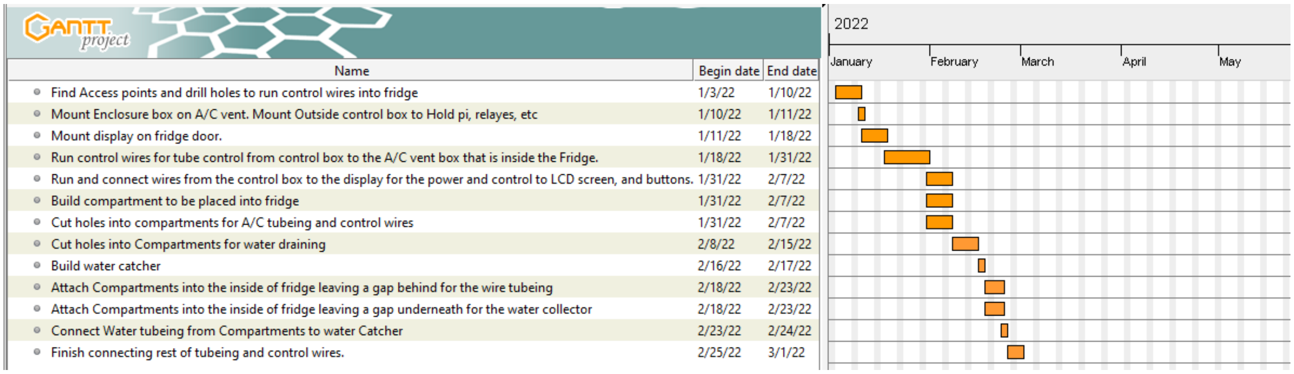


Schedule for Code Team

Name	Begin date	End date
Coding for Pi to Pi Communication	1/3/22	1/10/22
Code for Buttons	1/10/22	1/11/22
Code for the Display Initialzation	1/11/22	1/18/22
Code for the Menu Logic	1/18/22	1/31/22
Code for Temp/Humidity Sensors	1/31/22	2/7/22
Code for Solinoid Valves	1/31/22	2/7/22
Code for Misters	1/31/22	2/7/22
Combining Code and Debuging	2/7/22	2/24/22



Schedule for Build Team



Name	Begin date	End date
Find Access points and drill holes to run control wires into fridge	1/3/22	1/10/22
Mount Enclosure box on A/C vent. Mount Outside control box to Hold pi, relayes, etc	1/10/22	1/11/22
Mount display on fridge door.	1/11/22	1/18/22
Run control wires for tube control from control box to the A/C vent box that is inside the Fridge.	1/18/22	1/31/22
Run and connect wires from the control box to the display for the power and control to LCD screen, and buttons.	1/31/22	2/7/22
Build compartment to be placed into fridge	1/31/22	2/7/22
Cut holes into compartments for A/C tubeing and control wires	1/31/22	2/7/22
Cut holes into Compartments for water draining	2/8/22	2/15/22
Build water catcher	2/16/22	2/17/22
Attach Compartments into the inside of fridge leaving a gap behind for the wire tubeing	2/18/22	2/23/22
Attach Compartments into the inside of fridge leaving a gap underneath for the water collector	2/18/22	2/23/22
Connect Water tubeing from Compartments to water Catcher	2/23/22	2/24/22
Finish connecting rest of tubeing and control wires.	2/25/22	3/1/22

References

[1] T. Liu. "Digital-output relative humidity & temperature sensor/module." Sparkfun.com. https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf

[2] The Engineering Mindset, *How Solenoid Valves Work – Basics actuator control valve working principle.* (Mar. 3, 2019). Accessed: Nov. 5, 2021. [Online Video]. Available: https://www.youtube.com/watch?v=-MLGr1_Fw0c&t=109s

[3] A. A. Kader and M. Cantwell, "Appendix: Summary Table of Optimal Handling Conditions for Fresh Produce," in *Postharvest technology of Horticultural Crops*, Third., vol. 3311, Oakland, CA: University of California, Agriculture and Natural Resources, 2002, pp. 511–518.

[4] W. J. Florkowski, R. L. Shewfelt, S. E. Prussia, N. Banks, M. C. Dodd, and J. J. Bouwer, "IV Picking and packing," in *Postharvest handling a systems approach*, 3rd ed., San Diego, CA: Elsevier Science & Technology, 2014, pp. 457–465.

[5] W. J. Florkowski, R. L. Shewfelt, S. E. Prussia, N. Banks, P. Tonutti, and C. Bonghi, "'Omics' technologies and postharvest stress physiology," in *Postharvest handling a systems approach*, 3rd ed., San Diego, CA: Elsevier Science & Technology, 2014, pp. 523–533.

[6] E. Silva, "Respiration and ethylene and their relationship to postharvest handling," *eOrganic*, 24-Dec-2019. [Online]. Available: https://eorganic.org/node/2671. [Accessed: 21-Nov-2021].

[7] D. Holcroft, "Water Relations in Harvested Fresh Produce." The Postharvest Education Foundation, La Pine, May-2015.