AROAA: Automated and Research Oriented Aeroponic Agriculture

fubini.swarthmore.edu/grow

Jess Karol and Connie Bowen

Swarthmore College Department of Engineering

May 6, 2016

E090 Senior Design Final Report

ABSTRACT:

Controlled environment agriculture (CEA) is any system that alters the environment in which crops are grown. This environmental intervention can range from low-tech plastic crop covers to high-tech computer controlled indoor greenhouses. For our project, we designed a fully automated miniature greenhouse that uses aeroponic watering, LED growth lighting, and soilless growing medium to produce crops in a highly controlled environment. The system is designed to be used for conducting experiments to optimize crop conditions and grow food efficiently. Associated with the physical instrumentation is a website that allows users to view all the data and control the conditions of the greenhouse through an easy to use interface. Currently, the system is set up to test the effects of varying the quantity of water that is applied to each of the three crop rows. However, the system can be further modified to investigate other environment conditions such as light or nutrient levels. Over the course of the project, built a prototype of the system with full functionality and ran four trials to test the system. These trials helped us improve the implementation of the system as well as allowed us to begin optimizing the environment for various species of plants.

TABLE OF CONTENTS

Abstract: 1

Chapter 1: Introduction 4

- Project Goals 4
- What is vertical farming and indoor agriculture 4
- Issues with traditional agriculture 6
- The implications of vertical farming as a potential solution 6
- Application of our project in the vertical farming space 7

Chapter 2: Experimental Design 7

- Conceptual basis for design 7
- Realistic Constraints 10
- Overview of setup 10
- Design Features 10
  - Fluid Components (Aeroponics) 10
  - Nutrients 12
  - Structural Components 13
  - Envelope Components 14
  - LED Lighting 15
  - Electrical Components 16
  - Software Components 17

Chapter 3: Trials 20

- Purpose: 20
- Trial 1 20
- Trial 2 22
- Trial 3 25
- Trial 4 31

Chapter 4: Conclusions and Discussion 35

- Potential Improvements 35
  - Water Catchment Tray 35
  - Configuration to Test More Parameters 35
  - Configuration to Test More Plants 35
- Acknowledgements 35

References 36
Appendices.. 37
A: Vertical Farming Image 37
B: Computer Code 38
Data Collection Script 38
Light Cycle Script 39
Fan Cycle Script 40
Water Cycle Script 41
Web Application Repository 42

Chapter 1: INTRODUCTION

PROJECT GOALS

Our project goal is to design and implement a system for automated and research oriented aeroponic agriculture. The design is to serve two main functions. First, the system should have the ability to grow crops sufficiently and monitor the conditions that the crops experience. Second, the system should be able to conduct experiments to gather data on how to optimize the crops environment conditions. For the scope of this project, we will be experimenting with water quantity as an independent variable; however, the system can be further modified to address other variables such as light amount or nutrient quantity.

We wanted to build a system that is not only functional, but modular, employs good design principles, and is aesthetically appealing. We wanted a prototype that could be used in a consumer product as well as rigorously designed for research. In our construction, we utilized recycled components that were repurposed from previous engineering projects in order to reduce the carbon footprint and stay within a reasonable budget. While this took us away from the consumer product goal because the materials are heavy, we see our project as a conceptual prototype for a more customized and light-weight future iteration.

For the research aspect of our product, we wanted to contribute insights to the emerging field of vertical farming, so our highly controlled and closely monitored system mimics that of a vertical farm setting. By continuously collecting and analyzing data about the conditions of our environment as well as storing this information on a remote server, our system is engineered to be a workstation for researching crops growing in a similar but larger scale setting.

The core motivation for our project is to prototype a potential solution to large scale sustainable food production. According to the United Nations, the world population is projected to reach 8.5 billion by 2030. Our current traditional agricultural institutions are already being pushed to the limits using practices that are contributing to climate change and harming ecosystems. Furthermore, an increase in 1.5 billion people over the next 15 years will undoubtedly exacerbate these problems as well as cause food scarcity and starvation around the world. Our project allows us to think about this enormous problem and work with one promising technology that we believe could be a positive force correcting a dire situation.

WHAT IS VERTICAL FARMING AND INDOOR AGRICULTURE

Vertical farming is a proposed solution to the growing food, water and energy crisis. The concept was introduced by Dr. Dickson Despommier in the early 2000's.[1] It includes farming in vertical layers in a controlled environment to increase yield and conserve valuable resources. See appendix A for an example of a high tech vertical farm in Singapore to provide some pictorial information on how these

---

[1] Despommier, Dickson D. *The Vertical Farm: Feeding the World in the 21st Century*. New York: Thomas Dunne Books/St. Martin's Press, 2010.

systems look. From a theoretical perspective, it is helpful to think about resource conservation in the context of farming from a systems perspective.

Inputs to agricultural systems can include: water, lights, nutrients, soil, land, carbon dioxide, oxygen, energy (to operate machinery), herbicides, and pesticides. With both non-organic and organic conventional practices, many of these resources exist in abundance naturally (light, O_2, CO_2, soil, and sometimes nutrients). However, the non-abundant resources (water, land, pesticides, and energy from fossil fuels) are what can cause the system to be inefficient and lead to harmful consequences for the environment. Vertical farming practices forgo some of the abundant resources outside, but may counter-intuitively be a more efficient system that conserves and protects resources that are required for traditional farming.

Due to the 2nd law of thermodynamics, there will be losses in a given system, but we can tune the inputs in order to attain a more optimally functioning system that minimize losses. The more we have control over a system, the easier the goal of resource reduction becomes, especially when we are focused on conserving specific resources like fossil fuels and water. By sacrificing some of the abundant resources through farming indoors, we can potentially create a more controlled system that finds a more optimal ratio between each of the inputs. For instance, vertical farms better utilize land by farming at multiple vertical levels, they minimize water input by recycling and reusing run-off, and an indoor environment eliminates the need for pesticides. However, we now must provide the plants artificial light, nutrients, and control the indoor humidity/temperature. The result is using a tiny fraction of the land, approximately 70% less water, but more energy to supply power to lights, pumps, HVAC, sensors and other electronic components. Theoretically, the electricity that supports these devices can be generated using solar or other sustainable sources of energy, while traditional farming relies heavily on fossil fuels to power tractors and irrigation systems. Lastly, vertical farming can utilize automation systems to perform tasks like harvesting, while traditional agriculture still relies heavily on human labor.

While food transportation is not directly tied to farming systems, it is important to consider when analyzing the entire food production process. Indoor farms should not depend on the climate outdoors, so they are located where food is needed to minimize transportation waste and reduce the loss of food as it travels over long distances. Furthermore, indoor vertical facilities can grow large amounts of food in urban environments, excessively cold or hot environments, or environments without sufficient rainfall.

As technology advances, it is likely that indoor systems will increase in efficiency at a rate much faster than conventional practices because Controlled Environment Agriculture (CEA) technology is at the beginning of the innovation S-curve. LED lighting has been and will continue to become more energy efficient, data driven optimization techniques will increase growth rates and will make more crop rotations possible, and more automated harvesting technologies will continue to developed as the viability of this type of farming increases and more money is invested in the industry. The Cornell CEA program notes that the non-solar energy required to grow and transport produce 1000 miles is equivalent to the energy required to grow food locally using CEA technology.[2] This break even statistic is bound to improve as the technology advances and more investment is made in the industry.

---

[2] "Controlled Environment Agriculture." *Cornell : Research*. Web. 06 May 2016.

Issues with Traditional Agriculture

Our current global food system is deeply flawed, and agricultural production is a large part of the problem. Soil fertility is declining due to over-fertilization and a reliance upon intensive monoculture, water-shortages are increasingly severe causing problems when crops are grown in hostile environments, fertilizer-runoff is causing algae blooms in our lakes and oceans, and a large portion of agricultural products are lost between production and consumption due to poor transportation practices.[3] Issues related to water usage and nutrient pollution are becoming increasingly problematic as water tables continue to drop and climate change increases drought severity and likelihood.

The agricultural industry is by far the largest consumer of water in the US and around the world.[4] More than 70% of the water flowing from the Colorado River to its basin states is used for agriculture.[5] Clearly, reducing the quantity of water consumed by agriculture will have a significant effect on reducing global water consumption. However, water scarcity is not the only problem with agriculture's intensive use of water, runoff causes water pollution that is harmful to human health and detrimental to the environment.

Water from agriculture collects nutrients as it is passed through nitrogen rich soil, these nutrients flow into streams, rivers, and ultimately our oceans. In the ocean, nutrients sit at the surface and stimulate algae to grow in over-abundance. Once the algae die, it sinks to the bottom and decomposes, taking oxygen out of the water. This process causes large hypoxic zones, which have levels of oxygen so low that no aquatic life can survive. These zones no longer serve as a carbon sink, which increases atmospheric carbon levels and leads to even more global climate change. By reducing the amount of nutrient pollution from water runoff in conventional agriculture, we can fix this catastrophic problem.

The Implications of Vertical Farming as a Potential Solution

Vertical farming system can help solve many of the issues that large scale traditional agriculture causes, and like any new technology, introduce some new problems. If fresh produce is efficiently grown in cities, the food is more directly sold to consumers in the area, which would minimize transportation and food processing waste.

Water conservation is quite possibly the largest advantage of vertical farming. Hydroponic systems containing fertilizers and pesticides are self-contained and recycle nutrients. Plants can absorb the nutrients as they are recirculated throughout the system. According to Cornell's Controlled Environment Agriculture Lab, recirculation can save roughly 70% of the water that traditional agricultural uses, and mitigate the harmful environmental impacts that water runoff has.

---

[3] Gunders, Dana. *Wasted: How America Is Losing Up to 40 Percent of Food from Farm to Fork to Landfill.* Publication no. IP:12-06-B. 2012 NRDC Issue Paper Wasted How America is Losing Up to 40 Percent of Its Food from Farm to Fork to Landfill (3).pdf.

[4] "If You Think the Water Crisis Can't Get Worse, Wait Until the Aquifers Are Drained." National Geographic. National Geographic Society. Web. 06 May 2016.

[5] Zemora et al.

Of course, in an indoor setting, there is a relatively high demand for energy because grow lights and HVAC is energy intensive and water movement requires work from pumps. One of the goals of indoor agriculture is to capture maximal sunlight ("free" light) and use the most efficient HVAC and plumbing designs possible. Furthermore, sustainable energy sources are often taken into consideration when buildings are created/adapted to house vertical farms.

We have addressed the energy problem in several ways. We have designed the lighting system to operate only as needed - on a sunny day, the box can be oriented near a window so that the grow lights don't need to be turned on. Additionally, we are minimizing water utilization through recirculation. Finally, we considered the life cycle of the materials used to build this prototype - many components were old, refurbished materials.

Application of our project in the vertical farming space

Our project is intended to be useful in several potential applications. First and foremost, it is useful as a controlled environment to conduct research on optimize growing conditions for specific crops. The information obtained from controlled trials can be shared with the OpenAg API to help farmers, conventional or urban, make decisions based on empirical data to maximize crop yield. These conditions can be replicated on a large scale to efficiently produce food because our systems environment is designed to simulate these conditions.

Further, our box can be used for educational purposes - envision a classroom with a class garden. Students can learn about fluid dynamics from our misting system, coding to adjust system parameters, biology from observing seed to plant growth, and nutrition and cooking using fresh produce that they have produced. Additionally, our box is useful for the individual consumer seeking year-long access fresh greens - it consumes relatively little power, takes up minimal space, and requires no day to day maintenance. Our website can easily be accessed anywhere an internet connection is available, enabling the user to manage plant conditions from remote locations.

Chapter 2: Experimental Design

Conceptual basis for design

Our system was inspired by MIT's City Farm Food Computer. Similar to MIT's project, we designed a small-scale environment for application in research, education, and food production.[6] We deviate from MIT's project in several key ways - our box is mobile and was designed to be adjustable and useable for a variety of crops. The plant tray can be raised and lowered and can be replaced to accommodate larger pots. Additionally, there are 3 separate rows (9 individual crops per row) with variable settings. The goal of this design was to maximize the number of individual plants (to enable statistically significant findings from our experiments) while keeping the size of the box small for maximum control and uniformity of controlled conditions within the box.

[6] "Food Computers." OpenAG Initiative RSS. Web. 06 May 2016.

Aeroponic comes from the greek words aero (meaning air) and ponos (meaning labor).[7] While hydroponics involves growing plants with roots directly submerged in a water and nutrient solution, aeroponics involves exposing plant roots to a nutrient-laden mist.[8] In these types of systems, the roots are not surrounded by any medium, but protected in the light-proof barrier to the outside and sprayed on a time interval with a hydro-optimized nutrient solution. We decided to use aeroponics because it is relatively simple to implement, allows for a high level of controllability for experimentation, and minimizes algal buildup over time.

Conceptually, we would like the misting chambers for each row to be filled with a uniform cloud of water when activated to ensure each plant is receiving enough and the same amount of water. To achieve this goal, we purchased misting nozzles with a small orifice diameter to eject fine mist, and we deployed four equally spaced nozzles in each row to attain uniformity. Our design was also motivated by minimizing inputs, so instead of purchasing high intensity grow lights, we bought low power red and blue LEDs to be supplementary to external sources. This motivated the decision to make the outside of the grow box transparent to let sunlight enter the system. A side effect of this material created a highly transparent and visible system for viewers, which draws their attention and creates an opportunity to educate people about the project and vertical farming agricultural technology in general.

After trouble-shooting and running trials 1 and 2, our design ended up like that displayed in Figure 1. Future section explain individual components (the frame, the envelope in greater detail.

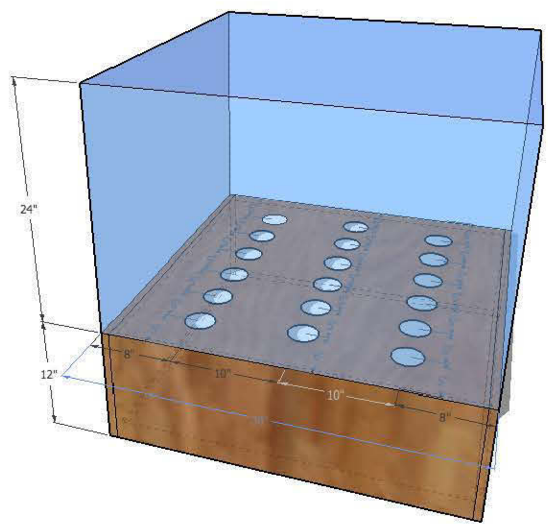


Figure 1: Preliminary sketch-up design of the grow box (excluding frame, aeroponic and electronic components). All marked measurements in inches.

[7] "Aeroponic Technology: Farming in air." Technology Times. (January 19, 2014 Sunday): 859 words. LexisNexis Academic. Web. Date Accessed: 2016/04/24.

[8] Despommier, Dickson. “A Farm on Every Floor.” NYTimes Op-Ed. Aug. 23, 2009.

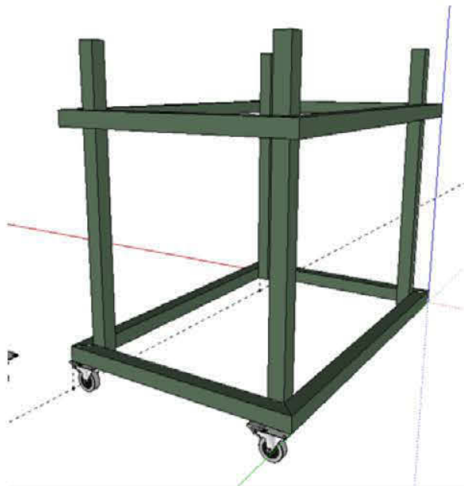


Figure 2 Preliminary sketch-up design of the Unistrut-frame component of the grow box. Unistrut is 1.5"x1.5" and cut to length, its grooves are inward facing, and the upper level of the frame is adjustable.

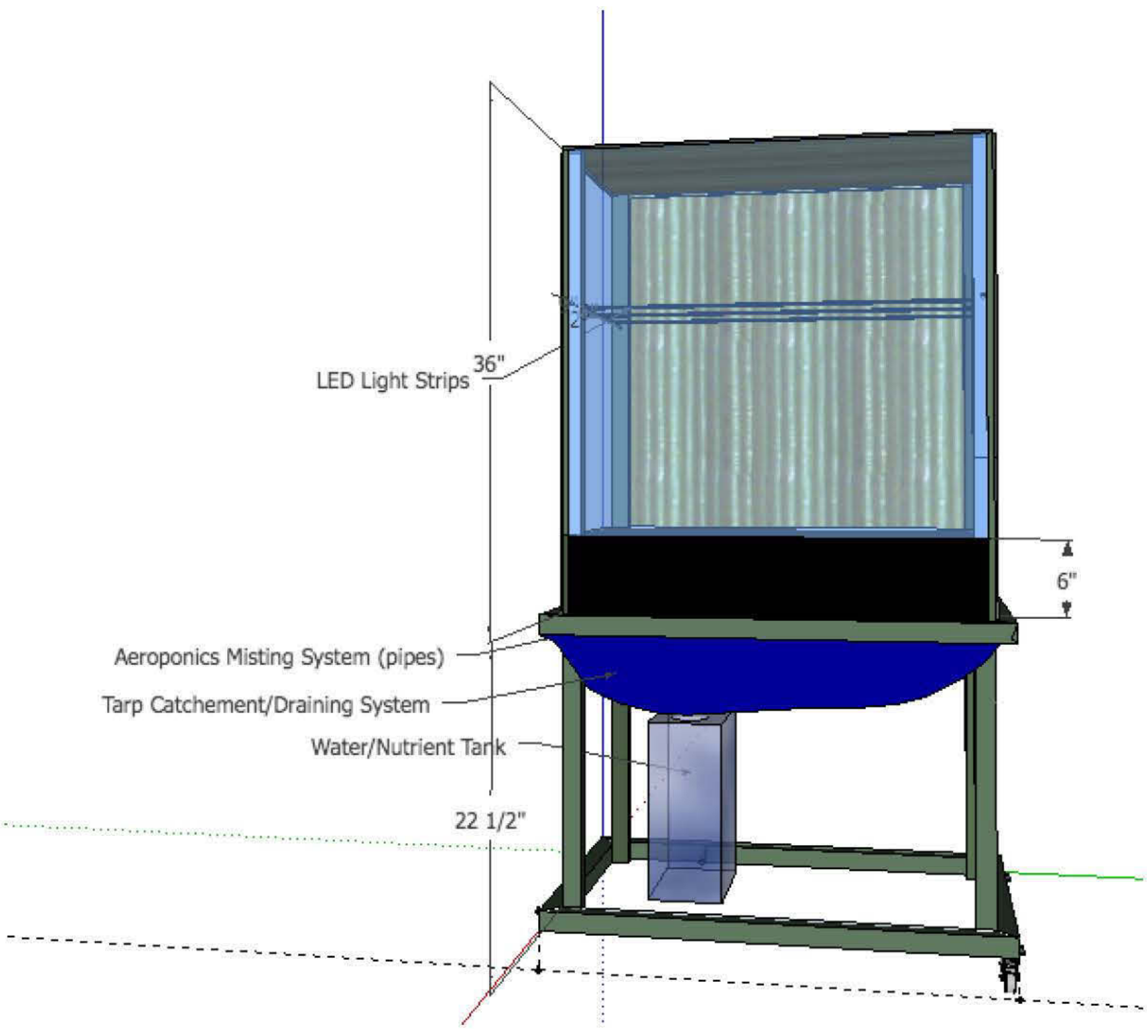


Figure 3 Model of final prototype.

Realistic Constraints

The main constraint for this project was time - plants take a few weeks to mature, so we had to budget our time to maximize the number of trials. This meant building the physical box during the Fall 2015 semester, and ordering components before we returned to start the Spring 2016 semester. We grew greens with minimal time between seed and harvest, but the quickest growing plants available (small Asian greens, cressida, pea shoots and arugula) still take about 21 days to reach maturity under ideal outdoor conditions.

Another major constraint was budget and construction skills. We used the materials and tools that we had available to us for a reasonable cost in both money and labor.

Additionally, we attempted to constrain ourselves to use environmentally low-impact materials. It proved to be difficult to stick to this constraint, as we needed to develop a prototype quickly to begin experimentation. Nonetheless, we did repurpose many scrap materials (all wood components, Unistrut pieces, plastic plant tray, parts of plastic sidings). Additionally, many of these components will be re-usable after this prototype is disassembled.

Overview of setup

Our experimental setup has three rows of plants, with 9 cells in each row. This design decision was driven by the need to have enough plants in each row to produce significant results for experimentation while keeping the size of the overall system within reason. The following table shows how the cells in our growing environment are labeled; this is how they will be referred to in this paper.

A-1	A-2	A-3	A-4	A-5	A-6	A-7	A-8	A-9
B-1	B-2	B-3	B-4	B-5	B-6	B-7	B-8	B-9
C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8	C-9

Figure 4: layout and labeling of plant cells within the growing box.

Design Features

Fluid Components (Aeroponics)

The aeroponic system went through 2 separate iterations - the initial design can be seen in **Error! Reference source not found.** below. Each row of misters is controlled by a solenoid valve. Due to the fact that our system requires high pressure and low volume, finding parts that provided this specification proved difficult. We used simple fluid dynamics relationships (PV=nRT) to estimate reasonable pipe diameters, nozzle opening and pump output requirements. We experimented with the pipes and misters (using hose pressure, valves, and a pressure gauge) to determine a range of pressures to be outputted by the pump for an acceptable mist. Based upon those tests, we purchased a 60 psi diaphragm pump and rigged it up to a 5 gallon tank to distribute water through our misting system to the plant roots. The system delivered 0.008 L/second (0.127 gal/min) to the plants.

The first version of the piping system used ID 3/8" steel piping. The nozzle heads were spaced 6" apart along the row, and only one solenoid was in place (there was no control between individual rows.) After Trial 1, we revised the system to include 4 misting heads per row, solenoids in each row (located outside of the box to protect them from water damage), and 3/8" ID plastic PVC piping. This system proved to work very well. Ultimately, we inserted plastic dividers between each row to enable variation in water and nutrient distribution between rows.

All fluid (water and nutrients) were stored in a plastic Nalgene tank with a spigot at the bottom and a hole on the top. Fluid was caught by the drainage tray, passed back into the tank, and recirculated through the system. We elected to directly mix nutrients into the water system because it was more economical and simpler (no additional valves/tanks were needed in this design.) The tank we started with had a 5 gallon capacity, but we swapped it for a smaller (2 gallon) tank because the larger tank was excessive. Every 10-14 days, we changed the water/nutrient solution to prevent algal build-up and encourage better hygiene in our system.

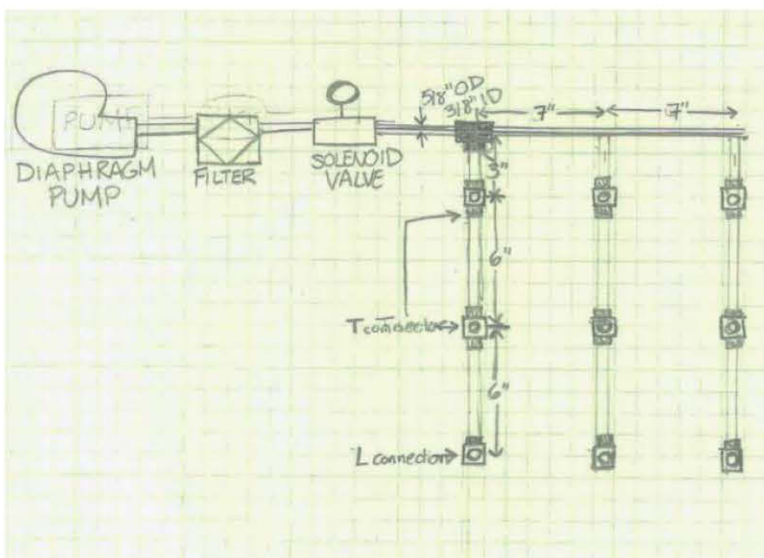


Figure 5 - Initial Aeroponics System Design

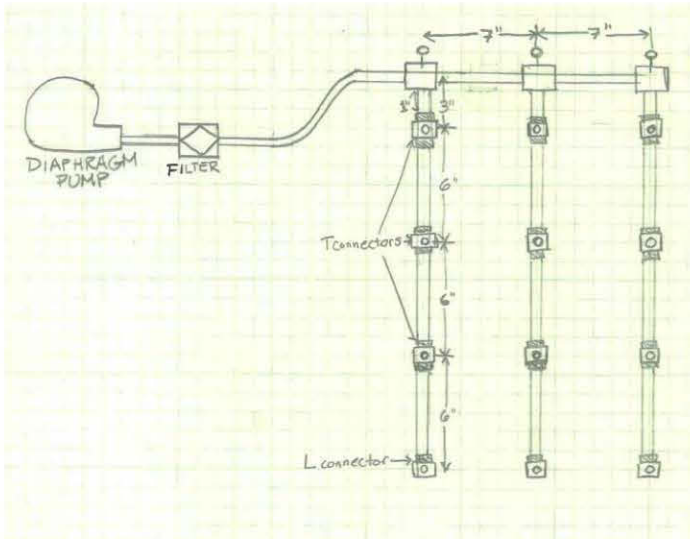


Figure 6: Final Aeroponic System Design

Nutrients

All plants need three primary macronutrients to survive – nitrogen (N), phosphorous (P) and potassium (K). Additionally, they require micronutrients to thrive. Different nutrients contribute to different phases of growth and elements of plant health.[9] After conducting research, we determined that most nutrient solutions are determined and optimized after extensive trial and error. Precipitation formation is a concern when soluble chemicals are mixed, and this is a particularly important thing to avoid in an aeroponics system because salt and precipitate build-up can clog misting nozzles.

We knew that we wouldn't have time to determine the optimal nutrient solution; optimizing the grow box itself was the priority. Consequently, we elected to use the General Hydroponics' Flora Series line of highly concentrated soluble liquid nutrients. These 3 solutions come with recommended feed charts, and are formulated to be mixed together in various ratios for different growing needs. They are also formatted to be completely soluble (no precipitates) for easy use in recirculatory hydroponic systems. We only used 2 of the 3 solutions in the series – FloraGro and FloraMicro. FloraGro has a 2-1-6 NPK ratio and also contains magnesium (Mg, micronutrient.) FloraMicro is intended for use with FloraGro and was used for Trials 3 and 4. It contains a 5-0-1 NPK ratio, as well as calcium (Ca), boron (B), cobalt (Co), copper (Cu), iron (Fe), manganese (Mn), molybdenum (Mo) and zinc (Zn). We followed the provided feed charts for recirculating hydroponics systems.[10]

Conductivity is a measurement of ion concentration in a solution; in other words, it is a measurement of salts. Nutrients are present in the form of ionic salts, so by using deionized water as a starting point, we were able to measure nutrient presence using a conductivity probe. The Hach40d probe that we used measures conductivity in microSiemens/centimeter (µS/cm), but it can easily be converted to parts per

[9] "Aeroponic System Basics." Mother Earth News.
[10] "FloraSeries." General Hydroponics.

million (ppm). The feedcharts that General Hydroponics provided included recommended total dissolved solids (TDS) levels in ppm. TDS is equivalent to conductivity (also known as electrical conductivity, or EC), and the conversion is direct.[11]

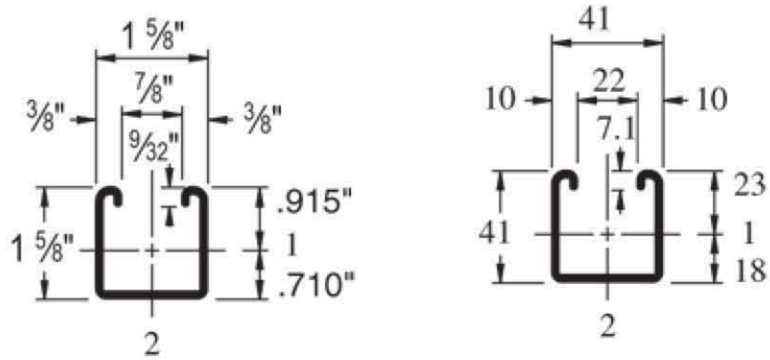
Figure 7 Initial Aeroponics System in Place

Structural Components

For the frame of the grow box, we elected to use 1 ⅝” Single Channel UniStrut (see Figure _). because it was available in large quantities in the shop and because it allowed for structural alterations throughout the project. UniStrut Metal Framing System uses hex-head bolts that thread into spring nuts to connect fittings to channels quickly and easily without any special tools.[12] It was not necessary to perform structural analysis on the framing because it was reasonable to assume that the frame was more than strong enough (the heaviest component of our project is a 2 gallon water tank that sits on the ground). Unistrut was sufficient for the prototype, but ideally, we could have used a much lighter weight (and better looking) material for the frame.

[11] "Electrical Conductivity (EC)." *Growth Technology*. Web. 06 May 2016.

[12] "About_overview UniStrut." About_overview. Web. 06 May 2016.



Elements of Section - P1000		
Area of Section	0.555 in^2 (3.6 cm^2)	
	Axis 1-1	Axis 2-2
Moment of Inertia (I)	0.185 in^4 (7.7 cm^4)	0.236 in^4 (9.8 cm^4)
Section Modulus (S)	0.202 in^3 (3.3 cm^3)	0.290 in^3 (4.8 cm^3)
Radius of Gyration (r)	0.577 in (1.5 cm)	0.651 in (1.7 cm)

Column Loading - P1000					
Unbraced Height (in)	Allowable Load at Slot Face (lbs)	Max Column Load Applied at C.G.			
		K=0.65 (lbs)	K=0.80 (lbs)	K=1.0 (lbs)	K=1.2 (lbs)
24	3,550	10,740	9,890	8,770	7,740
36	3,190	8,910	7,740	6,390	5,310
48	2,770	7,260	6,010	4,690	3,800
60	2,380	5,910	4,690	3,630	2,960
72	2,080	4,840	3,800	2,960	2,400
84	1,860	4,040	3,200	2,480	1,980
96	1,670	3,480	2,750	2,110	1,660

Figure 8 P1000 1 ⅝ " Single Channel Unistrut spec sheet

Envelope Components

The top portion of the grow box is surrounded by transparent walls. The primary function of these walls is to create a climate-controlled, sanitary environment that allows light through. Three of the walls, as well as the ceiling, are ¼" thick twinwall polycarbonate panels (see Figure _.) The polycarbonate sheeting is held in place and sealed by adhesive foam insulation tape and Gorilla tape (see Figure _.) The largest wall is cut in two - a top (photosynthesis space) and a bottom (root space) piece. The top piece is removable and is held in place on the top with high power magnets and the bottom piece is held in place by velcro. The bottom portion of each wall is painted black to prevent root light exposure. The fourth wall is the viewing window (the twinwall panels are transparent, but distort the view) and is made of thin clear acrylic plastic. The advantage of the twinwall panels is their high R-value (insulation), high light-transparency, durability, and low cost. The roof includes a ventilation hole with a fan (see 0, 0.)

LED Lighting

The LED lighting that provides energy for plant growth is supplemental to the external light entering the box. These lights were chosen to be optimal for a wide variety of plants. Research has shown that different photosynthetic pigments within plants require specific ranges of wavelengths of light to achieve different types of growth.[13] There are two categorical types of growth and development that plants undergo: 1) indefinite growth of the stem, branches, and roots and 2) definite growth of the leaves, flowers, and fruit.

Since the majority of plants that will be grown within our environment will not be flowering and producing fruit, we decided to maximize indefinite growth using red and blue LED lights. Red lights (640nm-680nm) are the most important for the process of photosynthesis. Specifically, the 660nm frequency is the most efficient wavelength for plant growth and promotes stem growth and chlorophyll production. Blue light (430nm – 450nm) helps seedling growth and prevents elongation during the plants germination phase. This is important for growing microgreens that are harvested at an early stage in their growth. We used LED strips with a ratio of 2:1 Red:Blue in order to provide the plants with what research has shown to be optimal for general vegetative growth.[14] Figure 9 shows the distribution of wavelengths that our supplementary lighting provides to the plants.

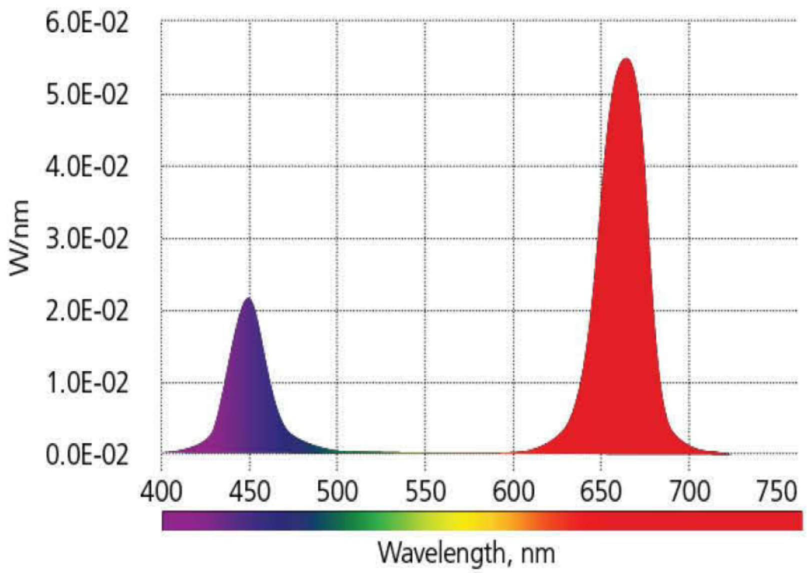


Figure 9: Ideal distribution of wavelengths for suplimentary LED growth lighting.

---

13 "Wavelength Influence on Plants | Illumitex." *Illumitex*. 2014. Web. 04 May 2016.
14 *ibid*

Electrical Components

The electrical components consist of the I/O ports on the raspberry pi, a custom designed PCB transistor control circuit, and a power supply. The I/0 from the raspberry pi is controlled entirely through software and sends a 3.3V signal to the control board. This board has the capability to take six control signals, boost the voltage to 5 volts to control high current transistors, which allow current to flow from the power source to the various hardware components (lights, valves, pump and fan). The power supply operates at 12V and is capable of supplying 12 Amps; however, the system only requires at most 10 Amps when the lights, pumps, and valves are on simultaneously. The following is the schematic for our DC controller circuit:

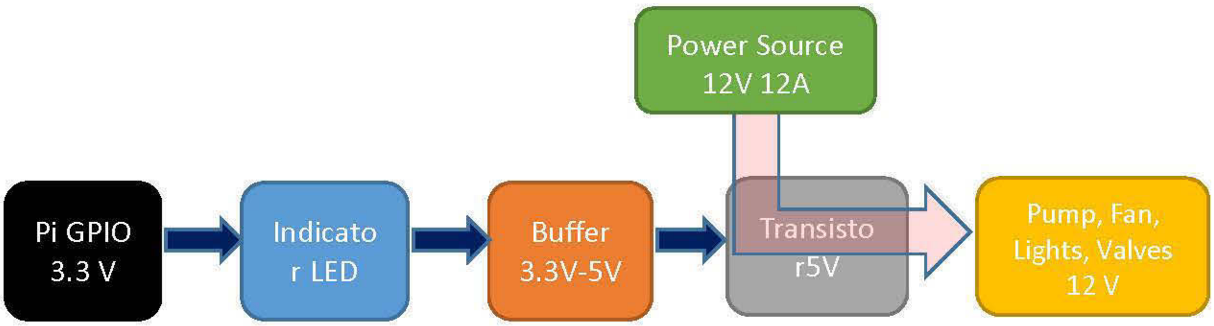


Figure 10: Flow chart illustrating the functionality of the schematic.

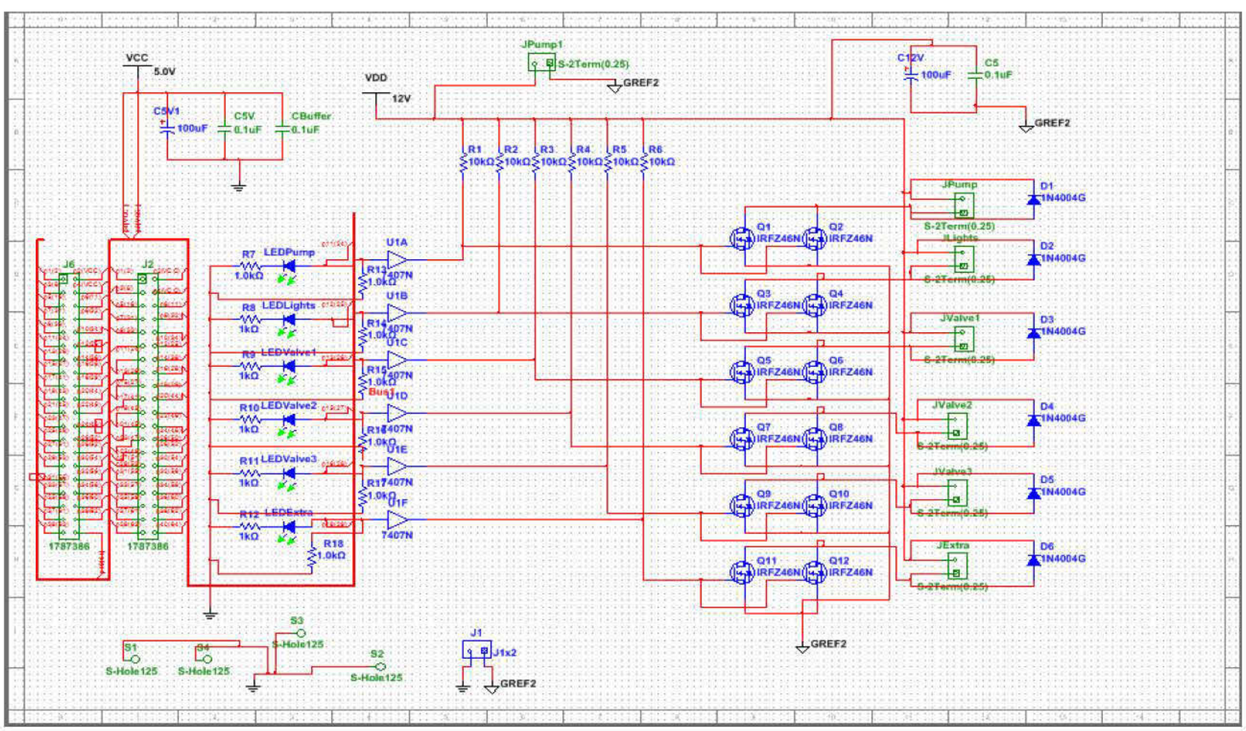


Figure 11: Electric Schematic of Control Board

Figure 10 is a flow chart illustrates the functionality of what is happening in the real circuit in Figure 11. On the left hand side are the pins from the raspberry pi. When the software tells a GPIO pin to go high (at 3.3V) a signal is sent to the buffer, illuminating an indicator LED along the way, which steps the voltage to 5.0V, enough voltage to control the transistors. The system's six hardware components are wired to the ports on the right hand side of the schematic, which receive the appropriate amount of current.

The electrical components that physically change the environment are the pump, valves, fan, and lights. We used a FloJet diaphragm pump from KingPumps that is capable of producing 60 pounds per square inch with a flow rate of 1.6 gallons per minute. This is sufficient to create a mist through a row of four misting nozels as discussed in the Fluid Components (Aeroponics) section. The pump operates at 12V and approximately 6 Amps. The flow of water is directed to each of the rows through solenoid values that operate at 12V and approximately .04 amps. The LED grow lights operate at 12V and approximately 2 amps. Lastly, the fan is a repurposed computer fan that operates at 12V and low amperage.

Software Components

The software for our system can be categorized in three components: the controller, the database, and the web application. These three components allow the system to control the environment, store the data, and let users interact with the system through an online platform. Figure 12 shows the interaction between each component of the system.

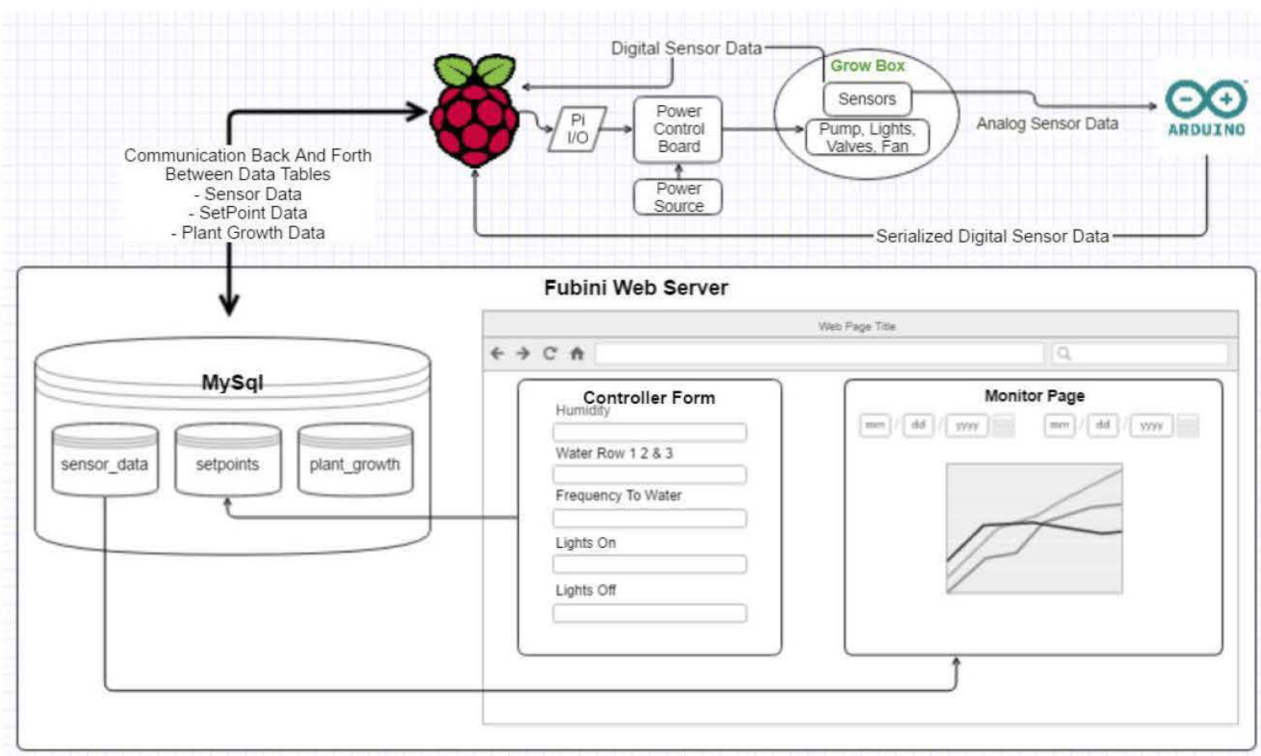


Figure 12: Gliffy diagram of software/hardware interconnectedness throughout entire system

The software for the controller resides on the raspberry pi microcontroller, which maintains individual processes for the fan, pumps, lights, and valves and adds sensory data to the database. The processes which are always running on the pi's linux OS, and are intuitively named, are as follows:

- ➔ light_cycle.py
- ➔ fan.py
- ➔ water_cycle.py
- ➔ data_collection.py

In order for these processes to control the various hardware components, we used object oriented programming to create classes to encapsulate each hardware components functionality. For instance, there is a class for the pump which has methods to set the input/output ports, and turn on and off the pump. The valve class has similar functionality. The fan class has an interface to control the speed of the fan using pulse width modulation (PWM). Each of the four main processes mentioned above instantiate instances of these classes to interact with the hardware as events are triggered. Below is a code snippet from the fan object to further illustrate the importance of this choice in code architecture.

```python
class fan(object):
    def __init__(self,pin):
        self.pin = pin
        self.pulse = GPIO.PWM(pin, 50)

    def duty(self,d):
        self.pulse.ChangeDutyCycle(d)
        return

    def freq(self, f):
        self.pulse.ChangeFrequency(f)
        return
```

The fan class allows the user to set the frequency and the duty cycle of the fan without considering the raspberry pi's PWM interface or GPIO pins. Using this style of programming eliminates redundant code as well as abstracts the functionality of the fan into an easy to use interface. The classes for the other hardware components achieve the same goal

The fan is operated based on a desired setpoint for the humidity. This setpoint can be established using the website, and setpoint values are stored in the database. The fan.py process on the pi reads the setpoint from the database and calculates the speed of the fan accordingly. If the humidity is lower than the setpoint, the fan remains off. If the humidity is above the setpoint, then the speed of the fan is calculated as follows:

$$\%\ of\ full\ speed = \max\left(\frac{current\ humidity - setpoint}{current\ humidity}, 0\right) * 100$$

The result of this calculation is used to establish a pulse width for which the voltage supplied to the fan should be modulated. The sample graph below illustrates how the humidity data changes with a setpoint of 85% relative humidity:

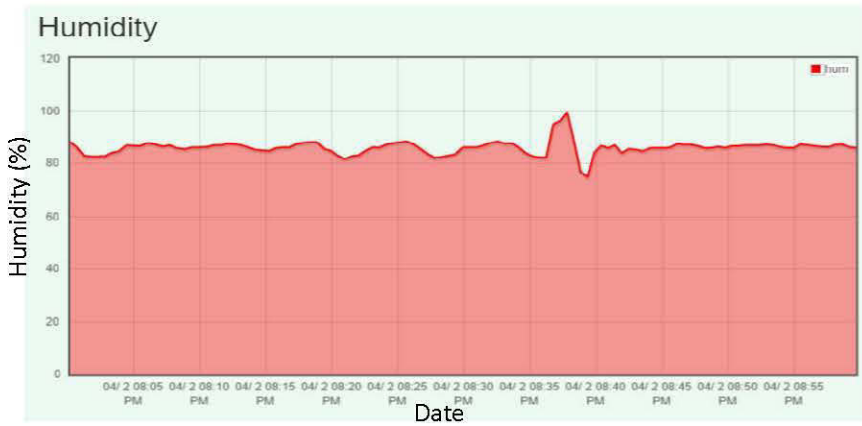


Figure 13: Sample data illustrating controller response to humidity with set-point at 85.

As you can see, the humidity hovers around 85% as a result of the fan adjusting to feedback from the sensor. The response is not perfectly even because humidity data is only collected every 30 seconds, so the fan control is a bit delayed. At about 8:37PM, there is a large spike in humidity caused by the watering cycle. As you can see, the fan system is fairly quick to respond in bringing the humidity back to the set-point.

The water_cycle process simply waters each row individually for a specified number of seconds. Then, the process calculates the approximate number of liters that is applied through the misting nozzles and submits the update to the database. Each row outputs approximately 0.008 liters per second. The watering process occurs at a frequency that also can be specified on the website. The light cycle simply turns on the lights at a certain time of the day and shuts them off at a certain time. These times are specified by a user on the controller form of the website.

Lastly, the data_collection process reads the sensor data (lux, light intensity, humidity, temperature) and submits them to a table that holds all the sensor data in the database. This process occurs once every 30 seconds.

The database that stores information for the website as well as data for the growing environment is built using mySQL and hosted on a virtual server that the Swarthmore ITS department maintains. At the growing environment, the Raspberry Pi microprocessor, which is connected to the internet, uses SQL statements to insert and select information for the database schema. On the other end, the website uses Django Models to interface with the database through the django mysql API.

The web application hosted at "fubini.swarthmore.edu/grow", serves three main functions. First, the website informs visitors about the goals of the project as well as explains some of the broader implications of vertical farming. Second, the website allows users to visualize the current, live data to see how the system responds to changes made by the controls. The monitor page uses jquery flot to plot data for a specified time frame. By default, only the previous day's data is displayed; however, the user can adjust this parameter to see whichever timeframe they are interested in. Lastly, the database is used to set the parameters for the growing environment including (lights on and off time, humidity setpoint, and the amount of time to water each row). These controls are connected through the database using a django ModelForm, which allows a Django Form to be directly tied to a Django Model, and interfaces neatly with the SQL database.

Chapter 3: Trials

Purpose:

These trials were designed to troubleshoot the prototype. After (and during) each trial, changes were made to improve or modify the grow box. Ideally, our system will reach a point where we can automatically adjust our controllable growth variables to optimize plant growth.

The first two trials did not yield quantitative results, but they were immensely helpful in informing our modifications. Trial 3 yielded quantitative results and was relatively successful - we feel that at this point our box finally reached a point where we could begin to adjust parameters to optimize crop growth without having to significantly alter the physical design of the system.

Trial 1

Significant Dates:

Started Seeds (in greenhouse): 01/19/2016

Transplant to Box: 01/29/2016

Add Fertilizer (end "germination" phase): 01/29/2016

End Trial: 02/06/2016

Parameters:

Watering Time	5 seconds
Watering Period	30 minutes
Water Quality	Untested tap water (later tested to reveal high conductivity)
Growth Medium	Natural cellulose sponge
Plant	Arugula
Light	2 strips LED grow lights/row
Humidity	No control (no walls)
Nutrients	Dilute Miracle-Gro LiquaFeed: Tomato, Fruits & Vegetables (applied bottom watering, 02/29)

Purpose:

To test the earliest form of our prototype.

- Can we grow arugula in this environment (3 misters per row, no walls, limited light)?
- Can the aeroponics system reliable be run remotely?
- What problems are limiting plant growth that we need to fix?

Set Up:

The arugula seeds were started in natural cellulose sponges in the greenhouse. The sponges were sanitized with heat, and the greenhouse is a relatively sanitary environment. 10 days after starting the seeds, the sponges were cut into sections and transplanted into the grow box. The grow cups were not sanitized for this trial, nor was the water tank, drainage tray or piping.

The grow box at this time included our initial aeroponic design (3 rows with 3 misters, metal piping, see Figure 7 Initial Aeroponics .) The plant tray was plywood with 9 plants per row, and each row received the same amount of water. The box did not have walls installed for this trial, and none of the sensors were in place.

Results:

Some of the plugs dried up during the course of trial. Some of them remained constantly soaked and developed black mold. Towards the end of the trial, white fuzzy mold developed around the plastic grow cups and the holes in the plant tray. Ultimately, the plants died and very little growth was observed.

Additionally, there was significant rust build up in the aeroponics system and in the water tank. The steel pipes that we used in our aeroponics system design corroded and deposited rust into the recirculating watering system.

Modifications:

We intended to add walls from the initial planning phase, but we were not able to complete this part of the construction until after this trial was complete. These walls help to prevent drafts and create a uniform climate in the box.

Furthermore, we decided to use a different growth medium for the remainder of the trials. We used cellulose sponges for this trial because it was readily available and because Connie had successfully run trials over the summer with cellulose as a growth media for arugula and other materials. Unfortunately, cellulose is not well suited for closed environments because it retains water too well. For the remainder of the trials, we elected to use Grodan A-OK Stonewool Start Plugs. Stonewool retains some moisture for better root-nutrient-absorption, but it drains well enough to prevent mold build up.

An unexpected change that we made after this trial was our aeroponics system revisions. Prior to this trial, we ran tests with our misters to ensure that the cones of fine mist would hit every hole in the plant base. The mist seemed to be relatively even and well dispersed in our preliminary tests, but during the trial, we had problems with plugs drying out. Moreover, we had issues with rapid pipe corrosion and rust sediment buildup. Clogged misting heads was a big concern, and we wanted to completely control the nutrients being added to our plants, so we revised our aeroponics system. The second iteration of the system (see 0) includes 4 misting heads per row, and it is completely PVC plastic (except for the stainless steel misting heads.)

Water quality became an obvious concern as we watched the dramatic reaction between the tap water and the metal pipes. Quality testing (pH and conductivity) were always components that we planned to measure in our box, but this trial spurred us to prioritize water testing. With Professor McGarity's help, we obtained a Hach40d water monitor with pH and conductivity probes. We ran some basic tests on the tap water and concluded that, based upon high conductivity, we should be using deionized water for the remainder of the trials (to give us complete control over the nutrients being added to the plants.)

A final major modification that was made after this trial was replacing the transistors in our control board. The high current required for pump (6 amps) was too much for the transistor to handle; in order to prevent our circuit from burning out, we were limited to 5 second water periods with cool-down breaks of at least 10 minutes. This is not an efficient way to water (and we did not want to melt the bread board), so we replaced the transistor with a several stronger transistors in parallel.

Trial 2

Significant Dates:

Started Seeds (in greenhouse): 02/02/16

Transplant to box: 02/12/16

Add Fertilizer: 02/12/16

End Trial: 02/29/16

Parameters:

Watering	5 seconds/60 minutes (+10 seconds to relieve pressure) 2 x daily (30s/12 hr)
Water Quality	DI water, see pH chart
Growth Medium	Grodan A-OK stonewool starter plugs
Plants	Assorted – Cressida, Tatsoi, Kyoto, Pea Shoots, Nasturtium
Light	2 strips LED grow lights/row
Humidity	No control (>95%)
Nutrients	FloraGro nutrient solution – dilute bottom feed (10mL Floragro/1 L DI water)

Purpose:

To test the updated prototype.

- What plants seem to grow best (fastest, healthiest) in this environment (4 misters per row, walls, no ventilation, limited light)?
- Will our new aeroponics system function better?
- Do we need ventilation?
- What factors are limiting plant growth/need fixing?

Set up:

All of the seeds were started in the greenhouse. A variety of quick-growing greens were planted because we needed to determine which plants would grow the fastest (to minimize trial time and therefore enable us to run the maximum number of trials.) Each seed variety was selected based upon predicted germination and seed-to-harvest time (forecasts were made for ideal outdoor conditions).

The grow box at this time included a PVC aeroponics systems with 3 rows of 4 misters per row, a sanitized and oiled plywood plant tray, sanitized (with dilute bleach) plastic grow cups, sanitized plastic polycarbonate and acrylic walls, and a removable wall. Nothing was light proofed at this point.

Results:

The walls were effective in blocking drafts and preventing drying out. However, they were almost too effective in retaining moisture. We implemented a humidity monitor towards the end of this trial, and we determined that the humidity hovered between 95% and 100% when the box was left closed for extended periods of time (over an hour.) Additionally, the box took on a very musty smell, which we believe was due to mildew and mold growth (the high humidity environment likely contributed to this.)

Everything in this trial exhibited growth, but only the pea shoot reached harvest-ready maturity. Black mold grew on the plywood plant tray and began to spread to the plants. All of the plant growth stalled about two weeks into the trial; it appeared that the plants went into shock. Some of the leaves began to develop white spots towards the end of the trial. These plants died in a matter of days – we are not sure exactly what disease struck, but something happened.

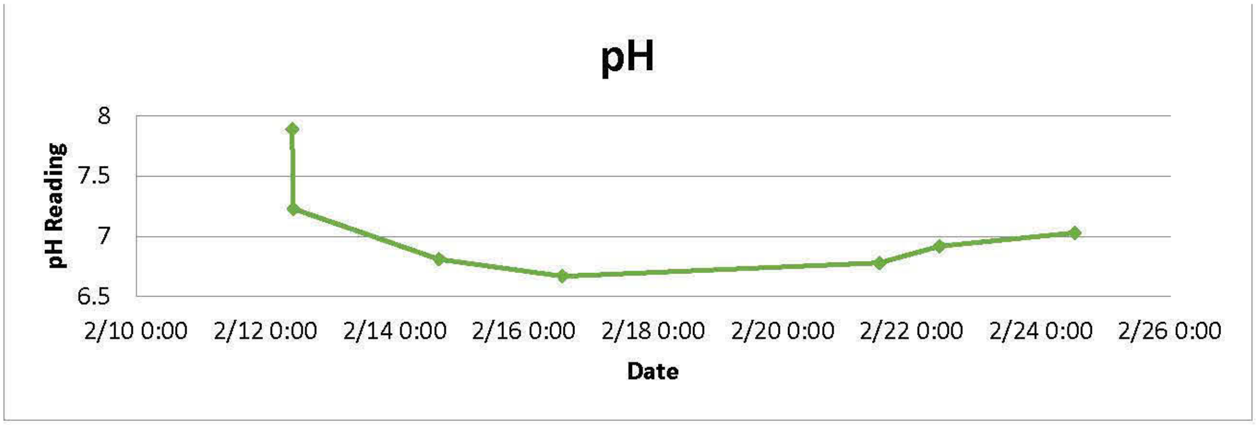


Figure 14 - pH of water/nutrient solution

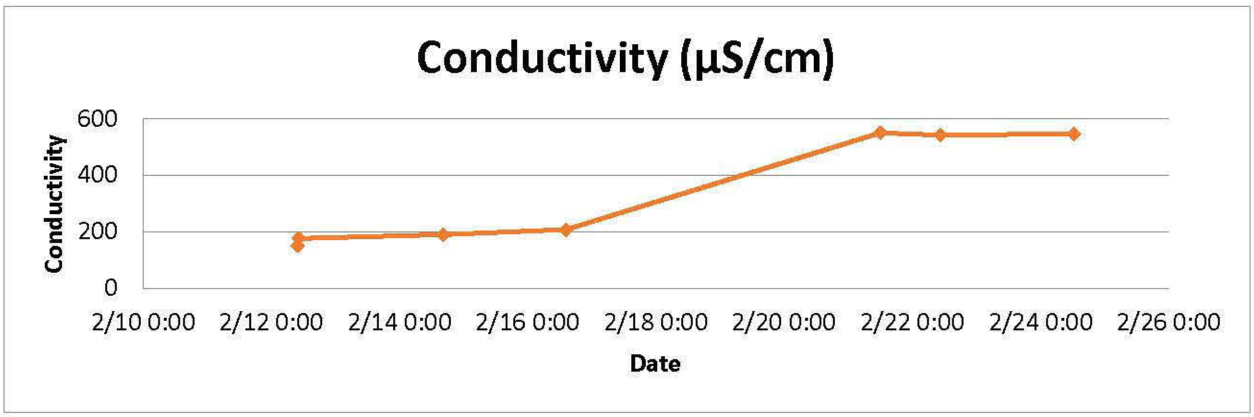


Figure 15 - Conductivity of water/nutrient solution

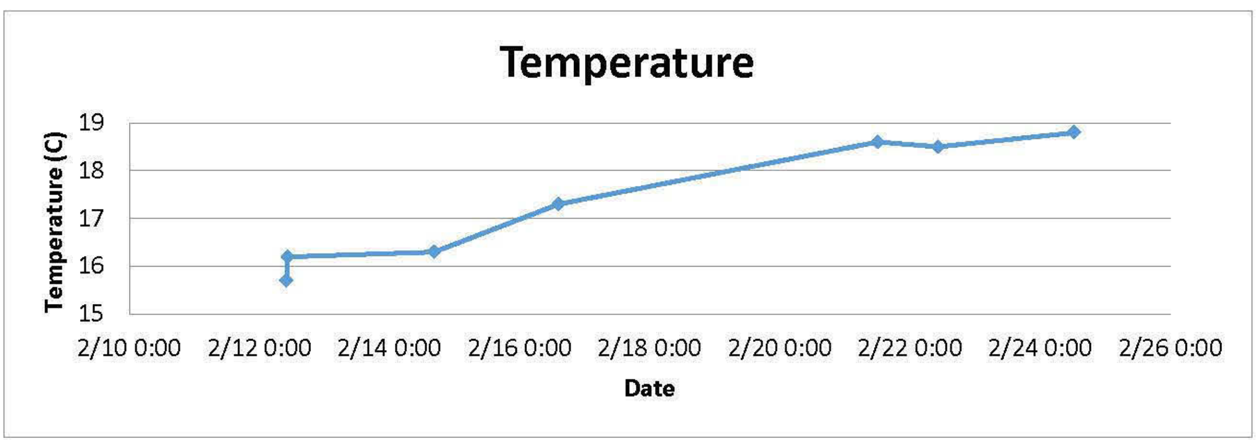


Figure 16 - Temperature of water/nutrient solution

Modifications:

To combat mold and mildew issues, we made two major changes. Firstly, we added a vent with an automated fan programmed to be controlled by the humidity/temperature sensors (see Software Components). Additionally, we replaced the plywood plant tray with a plastic plant tray. The plywood had been permeated by water, and black mold was spreading all over the back of the tray. The plastic tray was water-proof and easier to effectively sanitize between trials.

After doing some research, we hypothesized that the plants may have gone into shock due to root-light-exposure. When exposed to lights, plants release a burst of reactive oxygen species, potentially causing plants to enter a state of shock.[15,16] To combat this problem, we light-proofed the root section of the box with black latex paint. We also adjusted the drainage tarp to keep the area more light-tight (this was also necessary because the drainage tray was having some leakage problems over the course of Trial 2.)

Additionally, we added a second set of lights – we decided (based upon low lux-monitor readings) that our plants would benefit from more light, and the only reason we waited until the end of this trial was shipping time.

Finally, we added dividers between the misting rows. This enabled us to vary the quantity of water and nutrients delivered to each row.

Trial 3

Significant Dates:

Started Seeds (in grow box): 03/14/16

Add Fertilizer (end "germination" phase): 03/25/16

Move Upstairs: 04/09/16

End Trial: 04/13/16

Parameters:

Watering	Row A: 10 sec every 4 hours Row B: 20 sec every 4 hours Row C: 30 sec every 4 hours
Water Quality	DI water with nutrient solution, see pH/conductivity charts
Growth Medium	Grodan A-OK stonewool starter plugs

[15] Yokawa, Ken et al. "Light as Stress Factor to Plant Roots – Case of Root Halotropism." *Frontiers in Plant Science* 5 (2014): 718. *PMC*. Web. 6 May 2016.

[16] Yokawa, Ken et al. "Illumination of Arabidopsis Roots Induces Immediate Burst of ROS Production." *Plant Signaling & Behavior* 6.10 (2011): 1460–1464. *PMC*. Web. 6 May 2016.

Plants	Cressida
Light	4 strips LED grow lights/row
Humidity	85%
Nutrients	FloraGro and FloraMicro nutrient solution

Purpose:

To test the effect that varied watering (and therefore nutrient feeding) would have and to test the prototype and make further improvements.

Set up:

Trial 3 was our first official trial in which we tested an independent variable, namely water quantity. It was also the first trial in which we quantitatively measured plant growth using open source computer vision software in order to find meaning in our results.

The plants grown in this trial were watercress. We varied the independent variable (water and nutrient solution) by applying 0.24, 0.16, and 0.08 liters of water (with nutrients) to row A, B, and C respectively every 4 hours. This resulted in an accumulative 14.976, 10.016, and 5.056 liters of water over a 3 week period shown in Figure 24. (Note, the trial was run for longer than three weeks, but we did not record water utilization during germination and the initial growth.) The significance of the method of application was that row A received three times as much water as row C, and row B received twice as much water as row C. The beginning of this test took place in the basement of hicks where the system was previously located.

Methods

We measured the plant growth using overhead leaf area. The first 3 days of measurements relied upon using ImageJ, "an open source Java image processing program inspired by NIH Image.[17] E14 students took the first 3 days of data for us, and processed it using ImageJ. The students were finished with their reporting at this point, so we took over in data collection. We found ImageJ to be very tedious and difficult to automate, so we started using Easy-Leaf-Area. This application requires only high contrast and a red scale, and it allows us to run fairly accurate batch measurements.[18] We used Easy-Leaf-Area for the remainder of the trial, and plan to continue using it in the future.

Results

Plant growth was delayed - we concluded that this was probably due to insufficient amounts of light received by the plants in Hicks basement. Consequently, we decided to move the system to the third floor of Hicks near a south facing window to increase the magnitude of the parameter from external sunlight. Immediately after the move, we observed a qualitative and quantitative

[17] http://imagej.net/Welcome

[18] https://github.com/heaslon/Easy-Leaf-Area

increase in growth rates of the watercress, though a few plants did die (presumably due to excess stress). This leads us to believe that the low level of light was a large factor in growth stagnation during the first 3 weeks of Trial 3.

After averaging the growth for each individual plant over each row, we found that Row B exhibited the fastest growth over the entire trial period. However, the standard deviation and variance of these results indicate that they cannot be taken too seriously (see Table 1- Statistical analysis of growth between rows.) The Easy Leaf Area program required some manual adjustments, and the photos that we used to make these measurements were not perfect. It was almost impossible to perfectly align the red scale square in the plane of the leaves.

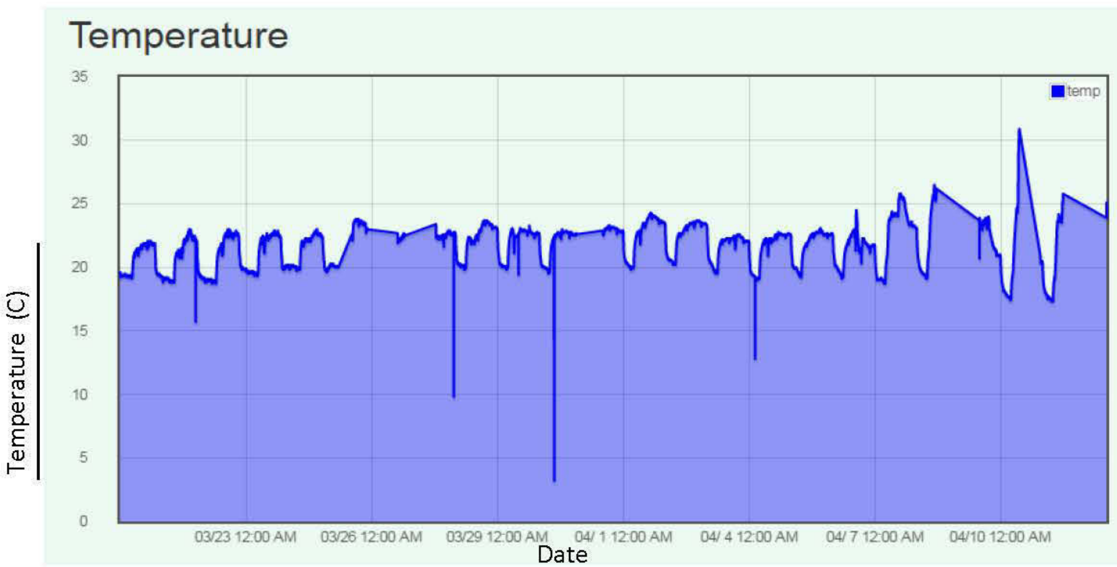


Figure 17 - Temperature in grow box (measured by humidity sensor on plant tray).

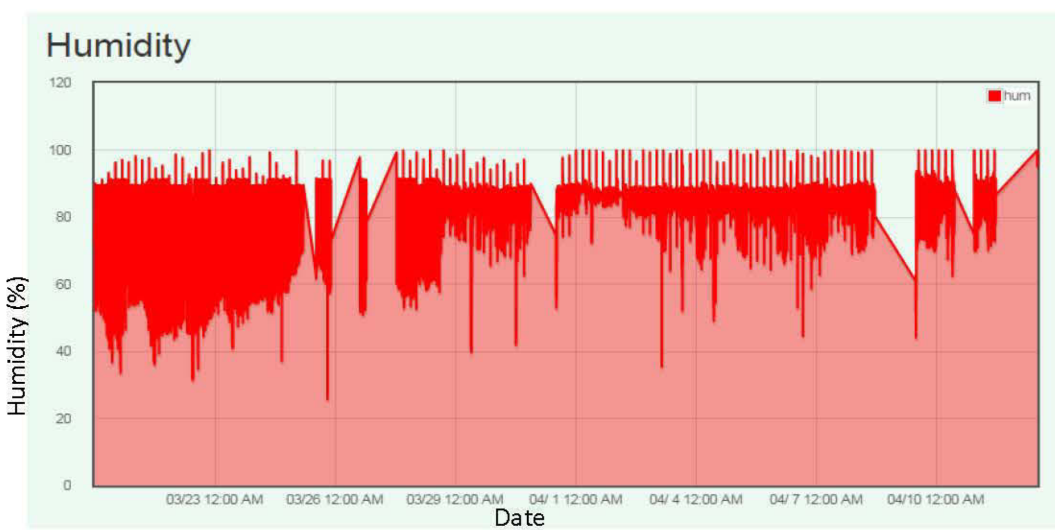


Figure 18 - Percent humidity in grow box (measured by humidity sensor on plant tray).

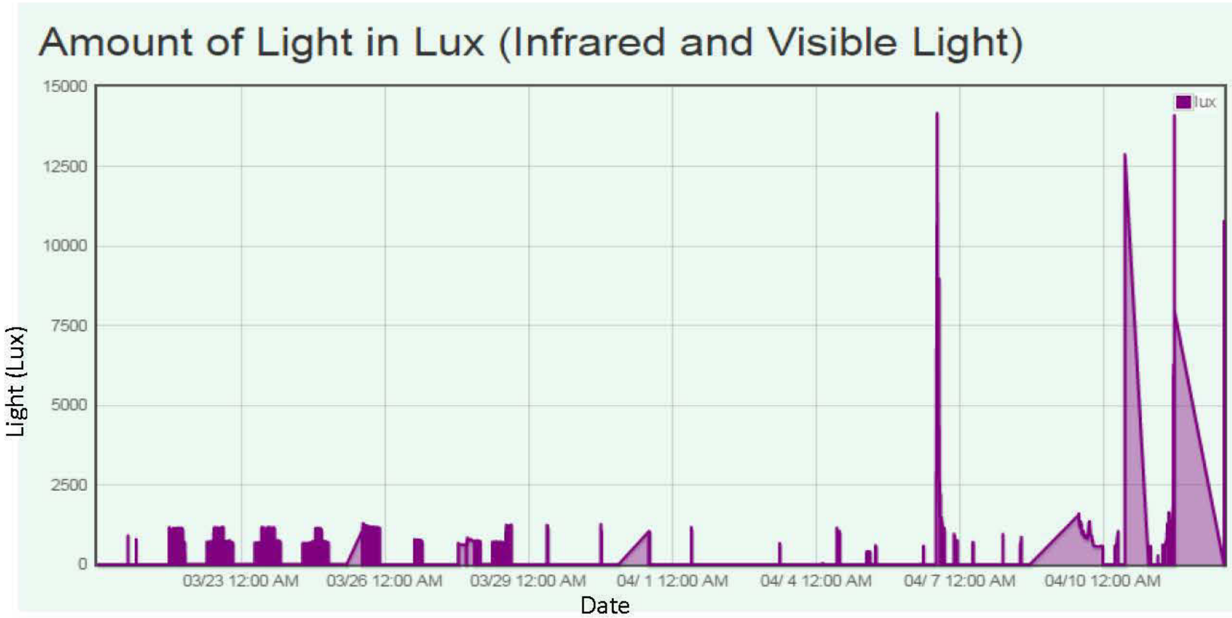


Figure 19 - Light measured by infrared/visible spectrum light sensor (Lux, or lumens per meter).

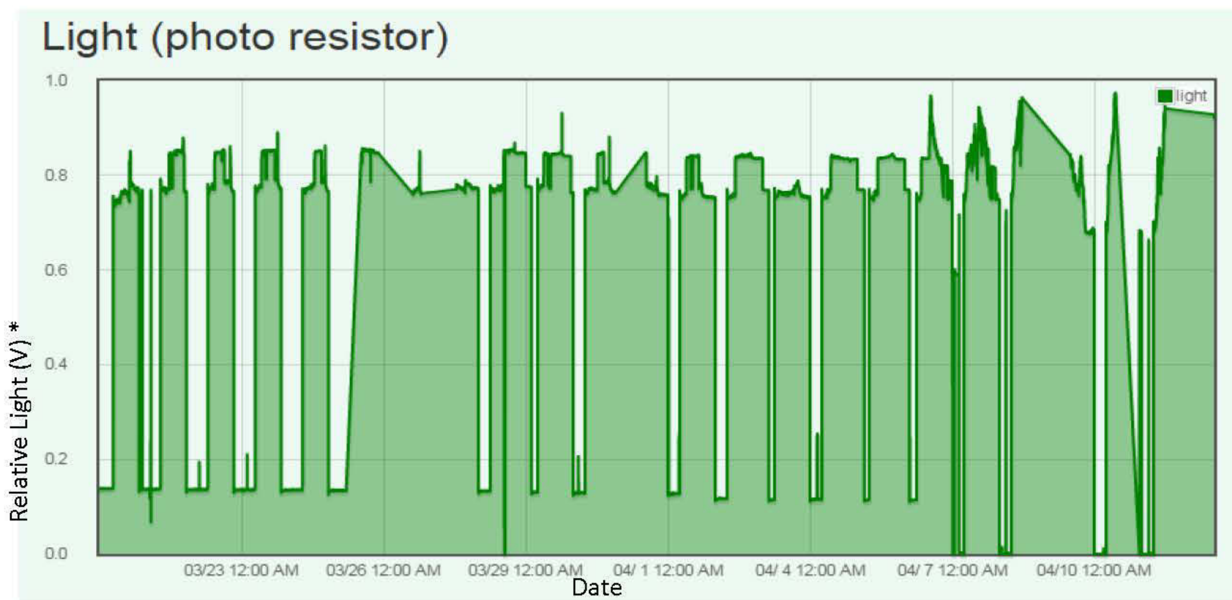


Figure 20 - Light measured by simple photoresistor (on plant tray)

*No units – the resistance of the photoresistor decreases as light intensity increases. The resistor is in a circuit with a voltage meter, and the voltage output is displayed on this plot. Comparing Figure 19 and Figure 20 provides a sense of meaning to the photoresistor data. This is useful because the lux-sensor stopped functioning for brief periods of time (more of an issue in Trial 4.)

Figure 21 - pH of water/nutrient solution.

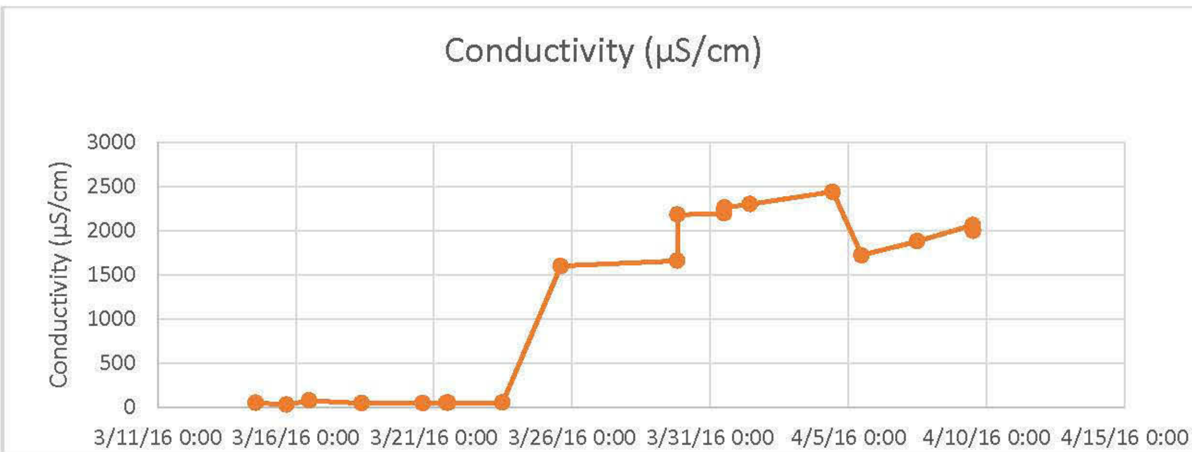


Figure 22 - Conductivity of water/nutrient solution.

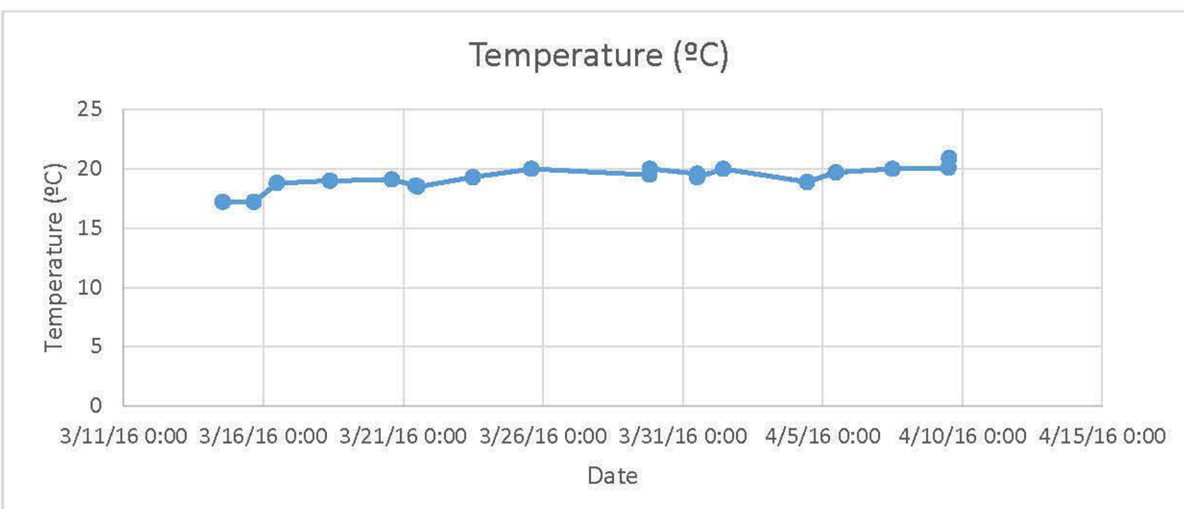


Figure 23 - Temperature of water/nutrient solution.

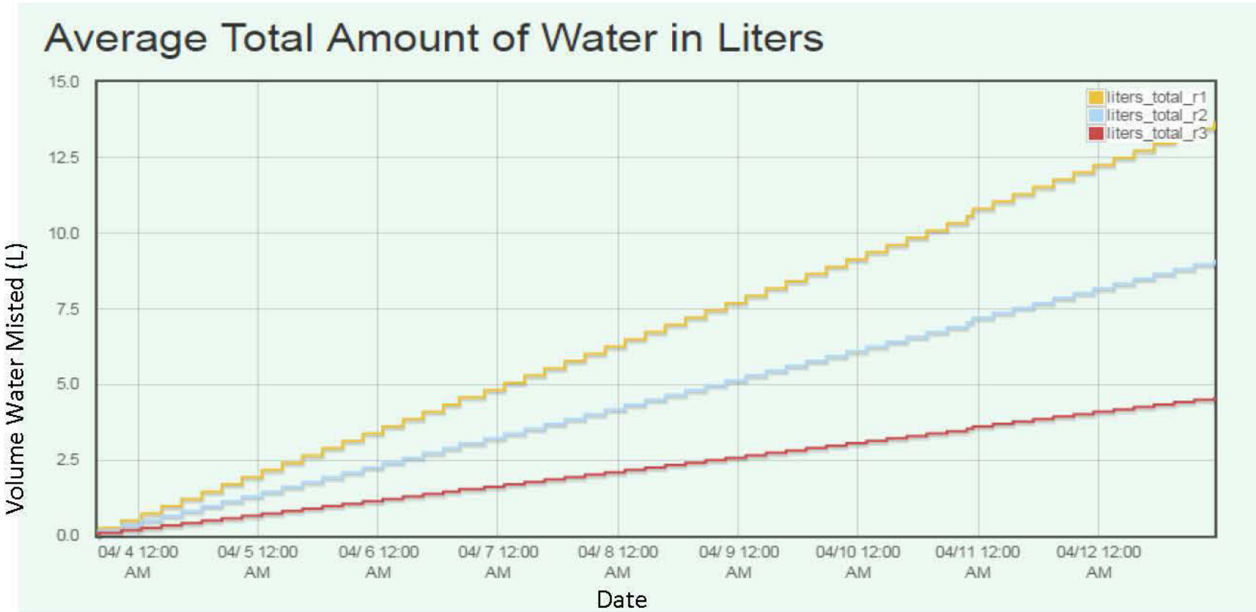


Figure 24- Visual of volume of water/nutrients distributed to each row over Trial 3.

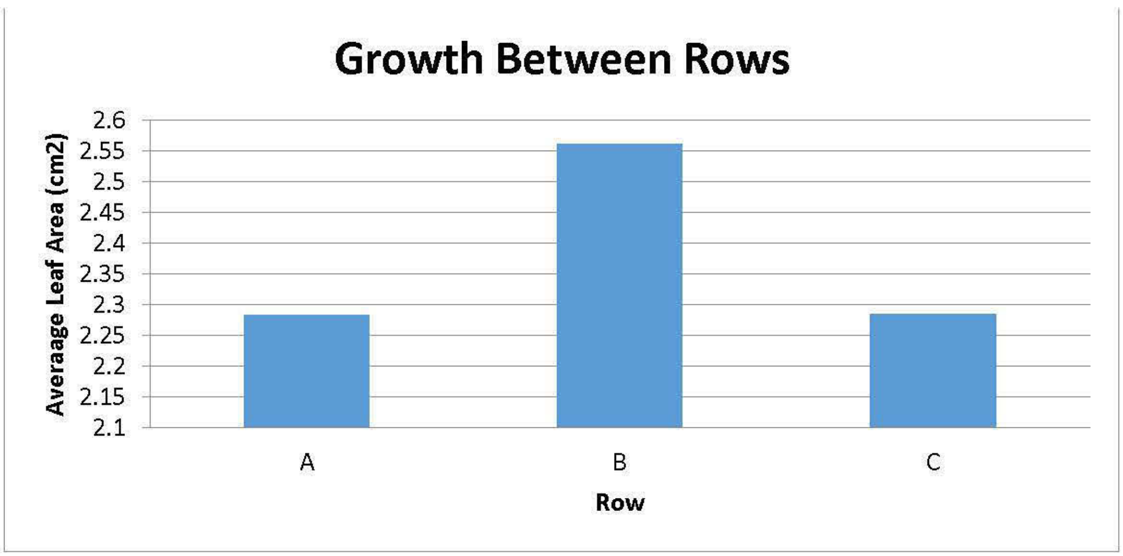


Figure 25 - Average growth between plant rows (Row A, B and C received 10, 20 and 30 seconds of water (respectively) every 4 hours.

Table 1- Statistical analysis of growth between rows

Row	Average (cm2)	Std Deviation	Variance (p-test)
A	2.28375	0.437890897	0.191748438
B	2.561666667	0.599149861	0.358980556
C	2.285714286	0.844610767	0.713367347

Modifications

The drainage tarp and tray system was replaced with a higher quality, more opaque tarp. We'd been having leakage issues, and the green plastic tarp that was in use for trials 1-3 was somewhat transparent, contributing to root-light-exposure.

The rest of the modifications that we made were control-based, instead of physical. The humidity and temperature controls were adjusted for the next trial, as was the watering regiment (shorter, more frequent watering.)

Trial 4

Significant Dates:

Started Seeds (in grow box): 04/13/16

End Trial: Still going!

Parameters:

Watering	**Row A: 15 sec every 90 minutes** **Row B: 20 sec every 90 minutes** **Row C: 25 sec every 90 minutes**
Water Quality	DI water with nutrient solution, see pH/conductivity charts
Growth Medium	Grodan A-OK stonewool starter plugs
Plants	Tatsoi
Light	4 strips LED grow lights/row
Humidity	72%
Nutrients	FloraGro and FloraMicro nutrient solution

Results:

The plants in this trial are doing fairly well, and are still growing. As of May 6, 2016, this trial is in its fourth week, and the plants are still exhibiting growth. Transporting the box to and from our E90 presentation was hard on the box (the tarp now leaks more and a few individual plants died.) The lux-monitor stopped working early in this trial, so we don't have that data for this trial.

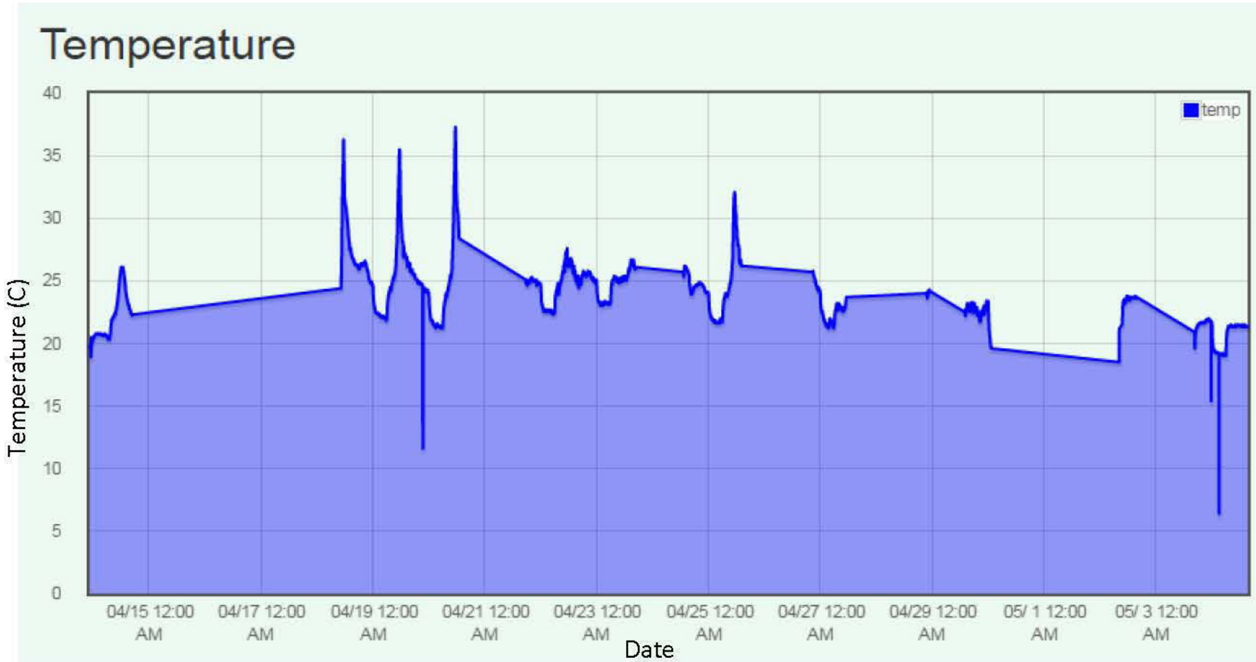


Figure 26 - Temperature in grow box over Trial 4.

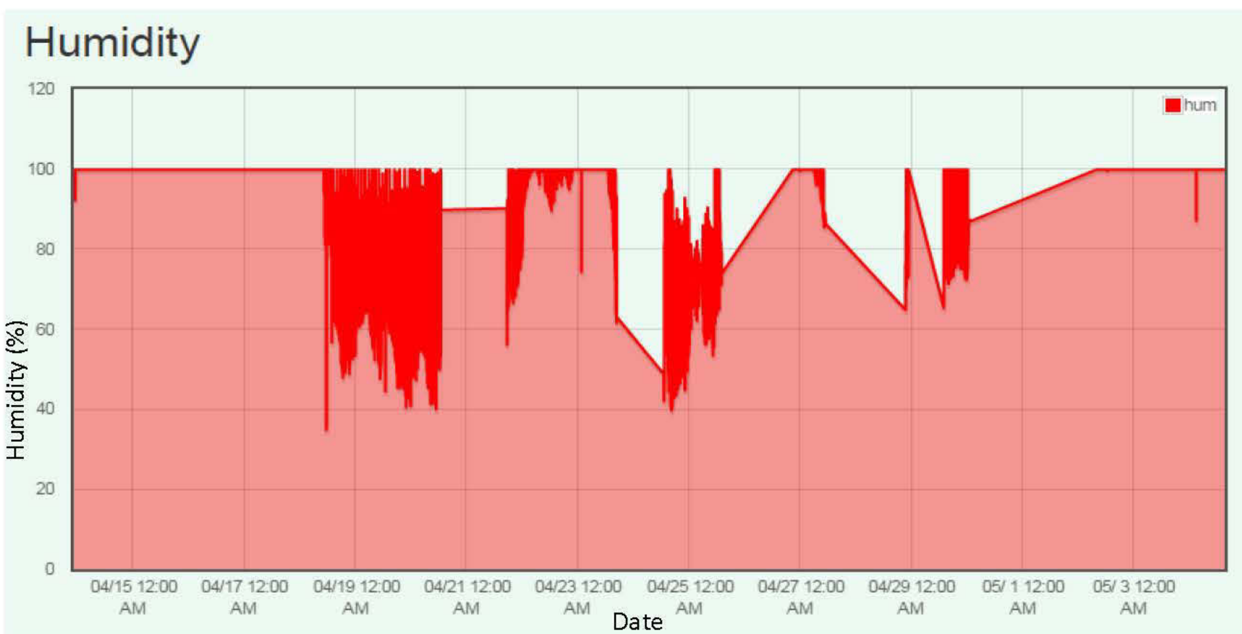


Figure 27 - Humidity in box over Trial 4.

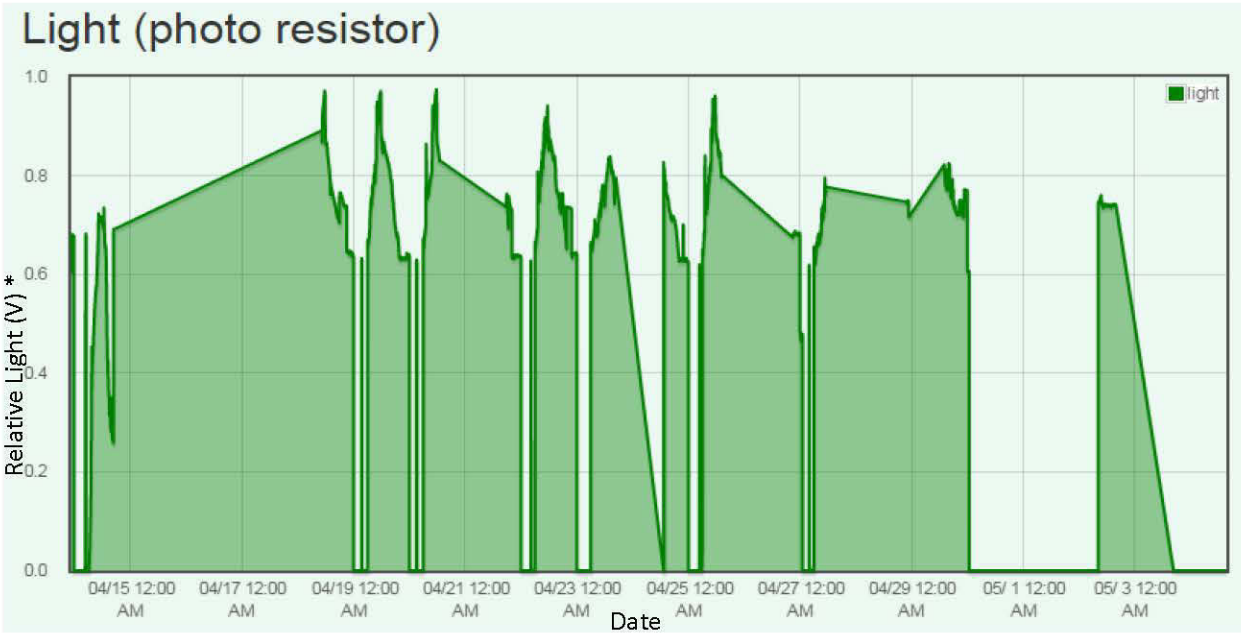


Figure 28 - Photoresistor light reading detected in box over Trial 4. Note: during E90 presentations, we moved the box to Sci199 and turned off the lights so as to avoid disturbing other presenters (accounts for low voltage readingat 05/01.)

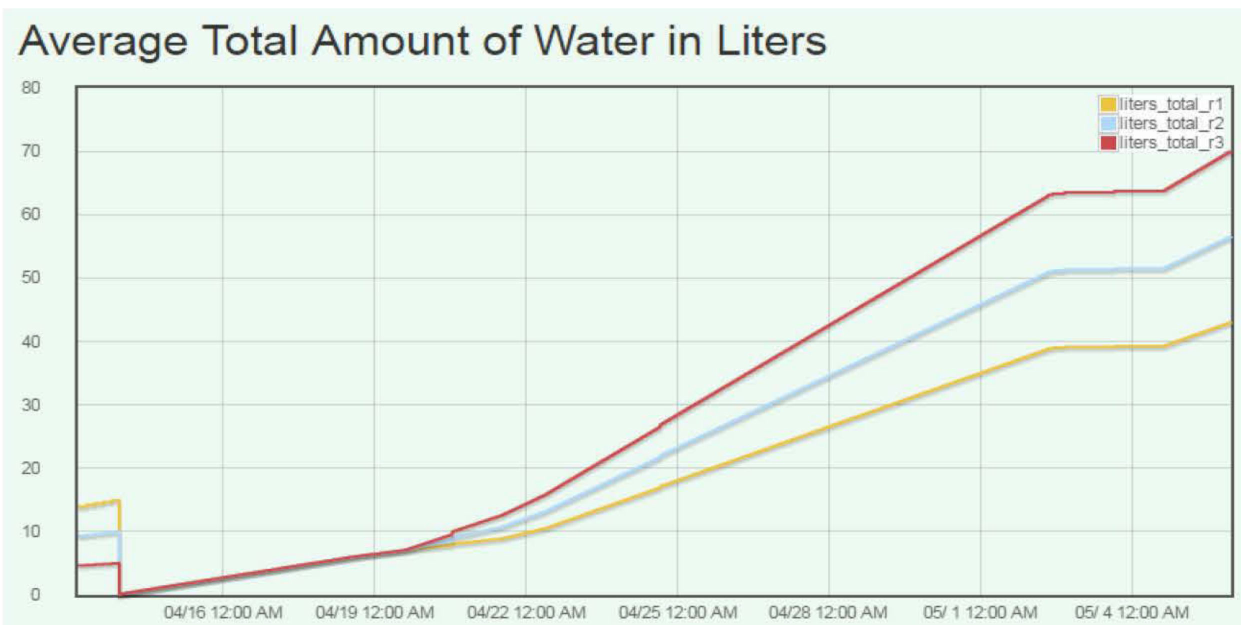


Figure 29 - Visual of volume of water/nutrients distributed to each row over Trial 4.

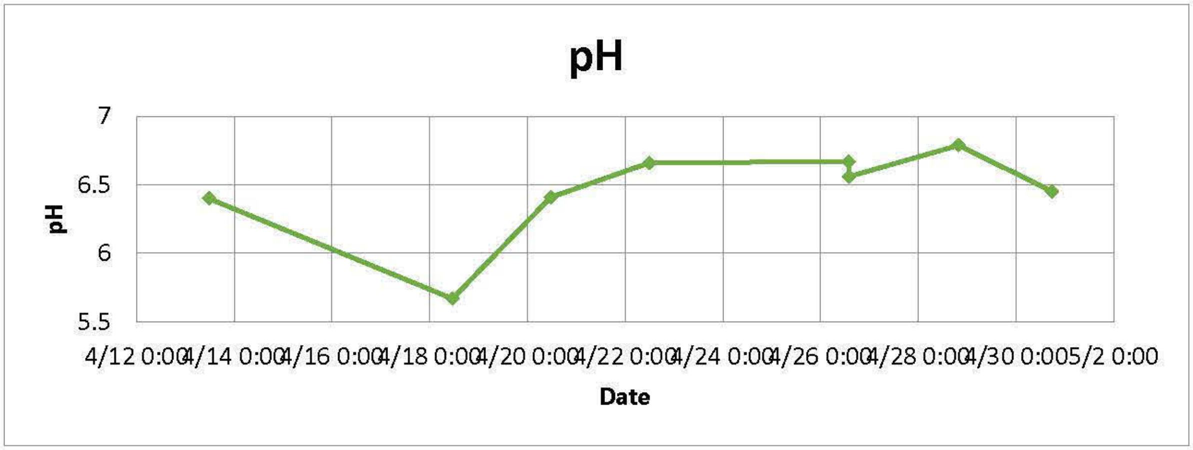


Figure 30 - pH of water/nutrient solution

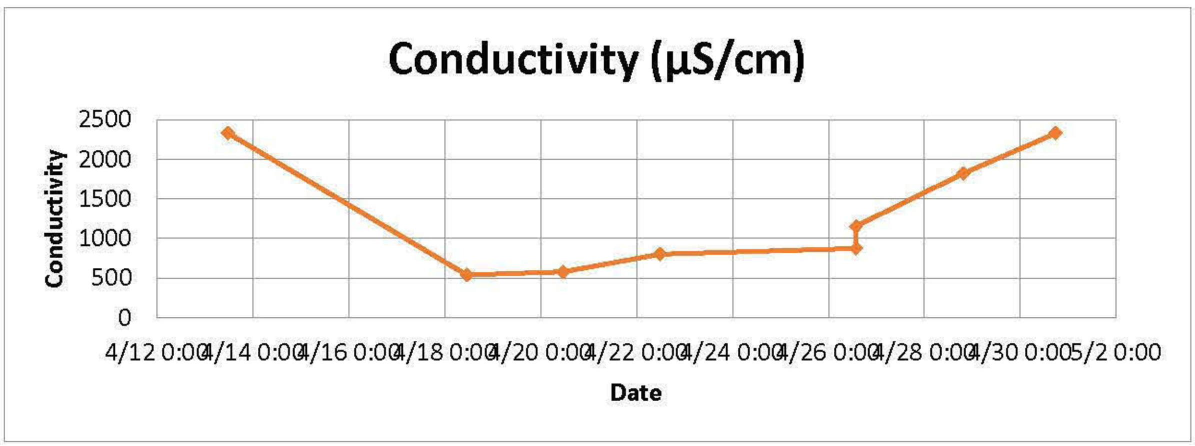


Figure 31 - Conductivity of water/nutrient solution

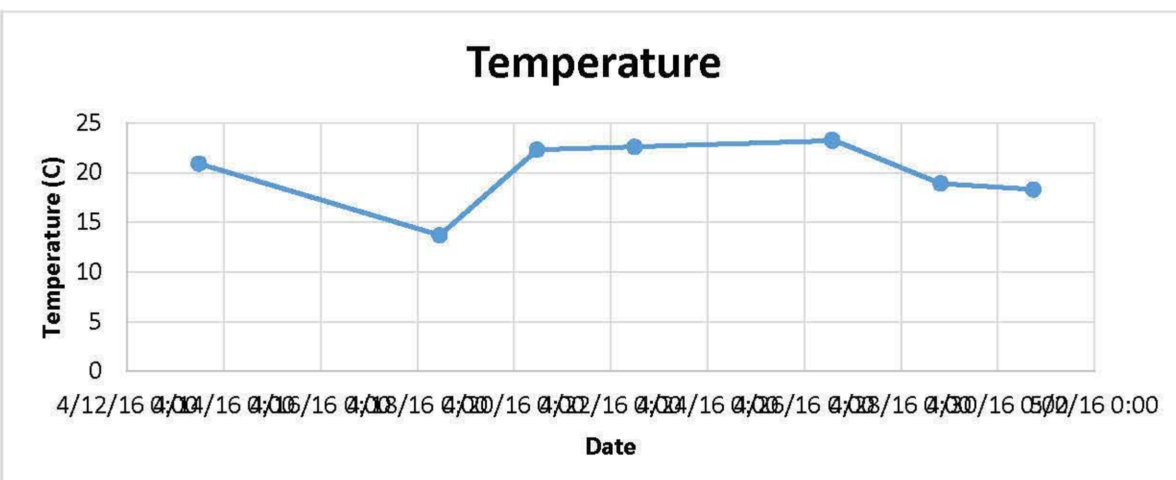


Figure 32 - Temperature of water/nutrient solution

As of May 6, 2016, Trial 4 is still running. The plants are continuing to exhibit signs of healthy growth, and we are continuing the watering regime as described above (see Parameters.)

Chapter 4: Conclusions and Discussion

Potential Improvements

Water Catchment Tray

We found it difficult to design a water basin with limited resources that fully retained all the water runoff from the misting system. We went through several design iterations, and ended up simply cutting a tarp to fit the shape of the bottom. The primary issue with the tarp is that it is not entirely light tight; this can cause algal buildup to occur, which is undesirable and may be harmful to the plants. However, our final implementation works sufficiently.

Configuration to Test More Parameters

The system can be used to research more parameters than simply amount of water by making some physical changes to the system. For instance, a separate hose can be connected to each of the tree valves with three different water reservoirs to test the effects of varied concentrations of nutrients. This would allow a researcher to optimize nutrient levels for a given plant type. A system could also be constructed where the quantity or spectrum of light

Configuration to Test More Plants

In order to test more plants, one could modify the PVC plant trey to contain three long rows that are not separated by pots. Rather, the rows could hold a continuous growth medium, which many more seeds are germinated in. This is similar to sprinkling seeds in a line of traditional garden beds. This design would maximize the number of plants that are growing.

Acknowledgements

We would like to thank many people for their contributions to the success of this project. Professor Nelson Macken, was very helpful in advising us and kept us on track for the duration of the project. J. Johnson was hugely helpful in providing expertise and assistance in building the actual frame and pipe system. Professor Carr Everbach aided us in initial concept design, as well as in assorted components of the project. Professor Cheever was crucial in assisting us with electronic components of the project, and Edmond Jaoudi was also helpful in providing us with electronic materials. Professor McGarity helped us with water quality monitoring and environmental lab access. Matt Powell from the Biology Department lent us a time-lapse camera. Kevin Webb also offered valuable insight on valve selection and microcontroller gardening. Finally, thank you to the ENGR014 students, Malia Scott, Angelina Abitino, Kyrstyn Ong, and Minseo Park, who helped us to measure leaf growth are during Trial 3 using Image-J.

REFERENCES

"Aeroponic System Basics." Mother Earth News.

"About_overview UniStrut." About_overview UniStrut. Web. 06 May 2016.

"Controlled Environment Agriculture." *Cornell : Research*. Web. 06 May 2016.

Despommier, Dickson D. *The Vertical Farm: Feeding the World in the 21st Century*. New York: Thomas Dunne Books/St. Martin's Press, 2010.

"Electrical Conductivity (EC)." *Growth Technology*. Web. 06 May 2016.

"FloraSeries." General Hydroponics. 06 May 2016.

"Food Computers." OpenAG Initiative RSS. Web. 06 May 2016.

Gunders, Dana. *Wasted: How America Is Losing Up to 40 Percent of Food from Farm to Fork to Landfill.* Publication no. IP:12-06-B. 2012 NRDC Issue Paper Wasted How America is Losing Up to 40 Percent of Its Food from Farm to Fork to Landfill (3).pdf.

"If You Think the Water Crisis Can't Get Worse, Wait Until the Aquifers Are Drained." National Geographic. National Geographic Society. Web. 06 May 2016.

"Wavelength Influence on Plants | Illumitex." *Illumitex*. 2014. Web. 04 May 2016.

Appendices

A: Vertical Farming Image

http://www.skygreens.com/

AeroFarms (Newark)

B: Computer Code

Data Collection Script

```
#!/usr/bin/env python
from serial import Serial
from pyfirmata import Arduino, util
from tsl2561 import *
import time
import sys
import os
import MySQLdb
import Adafruit_DHT

serial_port = '/dev/ttyACM0'
board = Arduino(serial_port)
serial_bauds = 9600;

db = MySQLdb.connect(host="fubini.swarthmore.edu",
        port = 3306,
        user="(enter a username)",
        passwd="(enter a password)",
        db="grow")

cur = db.cursor()

def main():
    sensor = Adafruit_DHT.DHT22
    tsl = TSL2561()
    pin = 25
    photo_resistor = board.get_pin('a:1:i')
    it = util.Iterator(board)
    it.start()
    photo_resistor.enable_reporting()
    sql = ("insert into grow.base_sensors (time, temp, hum, light, lux) values (%(time)s,
%(temp)s , %(hum)s , %(light)s, %(lux)s)")

    while 1:
        humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)

        if humidity is not None and temperature is not None:
            hum = humidity
            temp = temperature

        else:
            print 'Failed to get hum/temp reading. Try again!'
            hum = None
            temp = None

        light = photo_resistor.read()
        if light is None:
            light = 0
        lux = tsl.lux()
        values = {
                'time': time.time(),
                'temp': temp,
                'hum': hum,
                'light': light,
                'lux': lux,
        }
        print 'Temp={0:0.1f}*  Humidity={1:0.1f}%'.format(temp, hum)
        print 'Light amount: ',light
        print 'Light in Lux: ', lux
        cur.execute(sql, values)
        db.commit()

        time.sleep(30)

    s.close()
if __name__ == "__main__":
    try:
        main()
    except KeyboardInterrupt:
        print 'Interrupted'
        try:
```

Light Cycle Script

```
import RPi.GPIO as GPIO
import schedule
import time
from valve import *
from lights import *
import threading
import MySQLdb

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(11, GPIO.OUT)

db = MySQLdb.connect(host="fubini.swarthmore.edu",
        port = 3306,
        user="(enter username)",
        passwd="(enter password)",
        db="grow")

cur = db.cursor()


def main():
    l = lights(11)
    l.on()
    t = threading.Thread(target=set_schedule, args=(cur,l,))
    t.start()
    try:
        while True:
            schedule.run_pending()
            time.sleep(1)
    except KeyboardInterrupt:
        print "exiting"

    finally:
        GPIO.cleanup()

def set_schedule(cur,l):
    sql = "select lights_on, lights_off from grow.base_controller_setpoints order by id desc
limit 1"
    while True:
        result = cur.execute(sql)
        data = cur.fetchall()
        on = data[0][0]
        off = data[0][1]
        on_str = str(on)
        off_str = str(off)

        schedule.every().day.at(on_str[:4]).do(l.on)
        schedule.every().day.at(off_str[:4]).do(l.off)
        time.sleep(300)


if __name__ == "__main__":
    main()
```

Fan Cycle Script

```
import RPi.GPIO as GPIO
import schedule
import time
from fan_obj import *
import MySQLdb

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(36, GPIO.OUT)

db = MySQLdb.connect(host="fubini.swarthmore.edu",
        port = 3306,
        user="(enter username)",
        passwd="(enter password)",
        db="grow")

cur = db.cursor()


def main():
    f = fan(36)
    f.start_pwm()

    try:
        while True:
            print "making query..."
            sql = "select hum,temp from grow.base_sensors order by time desc limit 1"
            sql_set = "select humidity from base_controller_setpoints order by id desc limit 1"
            result = cur.execute(sql)
            data = cur.fetchall()
            temp = data[0][1]
            hum = data[0][0]

            result = cur.execute(sql_set)
            data = cur.fetchall()
            hum_set = data[0][0]


            print "The current humidity is: ", hum
            print "The humidity setpoint is: ", hum_set
            print "The current tempurature is: ", temp
            percent = (1 - abs(hum_set/hum) )*100
            print "calcualted percent is: ", percent

            if (percent > 0):
                #give fan a little burst to get going, electric motors need this when duty cycle
is very low
                f.duty(50)
                time.sleep(.25)
                # set the duty cycle, add a little extra so it is not too small
                f.duty(percent+3)
                print "Fan duty cycle is: ", percent + 3
            else:
                f.duty(0)
                print "Fan off"

            time.sleep(15)
    except KeyboardInterrupt:
        print "exiting"
    finally:
        GPIO.cleanup()


if __name__ == "__main__":
    main()
```

Water Cycle Script

```
import RPi.GPIO as GPIO
import schedule
import time
import MySQLdb
import schedule
from pump import *
from valve import *

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
GPIO.setup(12, GPIO.OUT)
GPIO.setup(13, GPIO.OUT)
GPIO.setup(16, GPIO.OUT)
GPIO.setup(18, GPIO.OUT)

db = MySQLdb.connect(host="fubini.swarthmore.edu",
        port = 3306,
        user="(enter username)",
        passwd="(enter password)",
        db="grow")

cur = db.cursor()
frequency = 5400
v1 = valve(12)
v2 = valve(16)
v3 = valve(18)
valves = [[v1,7],[v2,7],[v3,7]]

def main():
    p = pump(13)
    p.off()
    v1.close()
    v2.close()
    v3.close()

    time.sleep(5)

    try:
        while True:
            print "checking system params"
            update_params()
            print "Watering"
            p.on()
              open_each_valve(p)
            p.off()
            # drain the pressure for 10 secconds
            v1.open()
            v2.open()
            v3.open()
            time.sleep(10)
            v1.close()
            v2.close()
            v3.close()
            #push most recent action to database
            push_data()
            time.sleep(frequency)


    except KeyboardInterrupt:
        print "exiting"
        p.off()

    finally:
        p.off()
        GPIO.cleanup()

def push_data():
    sql_insert = ("insert into grow.base_water_amount (time, \
            liters_added_r1, liters_added_r2, liters_added_r3, \
            liters_total_r1, liters_total_r2, liters_total_r3) \
            values (%(time)s, %(a_r1)s, %(a_r2)s, %(a_r3)s, %(t_r1)s, \
```

Web Application Repository

- https://github.com/jpk0727/growApp