ACAMP

Automated Care And Monitoring of Plants

Justin Johnson
Electrical Engineering
jurjohns@ucsc.edu

Stark Pister
Computer Engineering
spister@ucsc.edu

Wayland He
Computer Engineering
whe3@ucsc.edu

Daniel Gunny
Electrical Engineering
dgunny@ucsc.edu

Table of Contents

Purpose

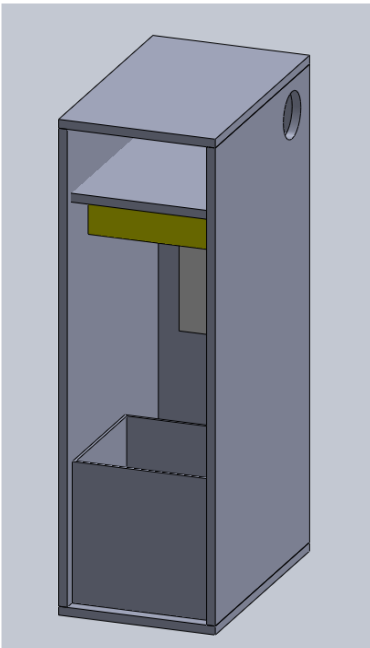
Why Hydroponics

As the supply of fresh water decreases and the amount of arable land is diminished, solutions are needed to continue the production of plants and food. ACAMP's goal is to develop a system that requires little to no interaction by the user while optimizing the growth of the plant. Hydroponics is a particularly environmentally-friendly form of plant cultivation as far less water is consumed in the hydroponics process compared to traditional means of cultivation. Water is recycled and used repeatedly, reducing the amount of water wasted substantially. ACAMP intends on building a custom web interface, where the user will be able to select the settings that best fit the environment the plant thrives in. An exotic food can be grown locally instead of being flown in from across the globe. In addition to an adjustable environment, users can also observe the growth of their plants through a webcam inside the growth chamber.

Why Automation is Important

Automation of the hydroponic growth cycle eases the growth process, opening the door to gardening for a whole range of people: inexperienced growers, people lacking time, people lacking the outdoor space to grow, and people who demand precision in their growth process. Automation ensures that the environment will always be optimized while operating at a much higher level of efficiency. We believe that the automation ACAMP provides will revitalize an interest in subsistence agriculture.

Visual Overview



The graphic on the left is a simplified version of what we imagine our project to be. While the door is missing, the graphic shows the general framework, the reservoir/grow table, and the adjustable light. The dimensions of the structure were intended to allow the structure to be moved through most doors. The two pictures on the right show the progress completed so far in the physical design.

Research Completed

As nobody in our group has any experience gardening with hydroponics, we had a considerable amount of learning to do before beginning our project. We began by talking with professors and persons in industry about general design considerations and possible sensor choices, in addition to doing web research over winter break. We quickly found a fantastic beginners guide to hydroponic gardening which gives a basic overview of the processes of growing indoors. Found locally, Santa Cruz Hydroponics has been a great resource for both sourcing our components and drawing upon their employee's knowledge base.

When asking for recommendations for literature from both Santa Cruz Hydroponics and Steve Fambro there was one book recommended above all other for beginners, "Gardening Indoors with Soil and Hydroponics" by George F. Van Patten. From giving us an idea of the ideal levels for our environmental factors to how often the air in our room needs to be re-circulated, this 384 page book has 670 full color illustrations and has been an invaluable resource for our design. Van Patten's book devotes several chapters to different types of hydroponic setups and various methods of controlling the environment to encourage healthy and happy plant growth. This book has helped us tremendously and has earned its reputation as "The Bible" amongst indoor gardeners.

We identified pH and conductivity sensors as problem points early on, and soon after our project was approved we sent an email to Doug Au of MBARI asking for help in choosing a pH sensor. We received confirmation that in situ pH sensors are notorious for losing their calibration after a few months in solution. It was pretty evident from this point that a commercial system for monitoring pH and total dissolved solids was our only real solution.

We were debating the merits of hacking the data output of a combo pH/TDS meter into our micro when we discovered Atlas Scientific while searching on the internet. Their company specializes in environmental monitoring projects; their sensors can interface easily with a microcontroller via UART for ph/TDS/dissolved oxygen. After another week or so of discussion it seemed pretty clear that there was no other competitor for pH monitoring. Subsequently, we ordered a kit which came with sensor and calibration solutions. After we found how easily used the Atlas sensors were, we ordered the TDS sensor as well for our nutrient monitoring. Several weeks later, we were so happy with our first two Atlas sensors we purchased their submersible temperature sensor as well.

Santa Cruz Hydroponics has consistently been a great source of information and guidance for our project. They have a huge selection of everything one needs to equip an indoor garden. So far, they have sold us the water table, pumps, reservoir, and interior coating that will be used in the final design. We expect to be purchasing the bulk of our remaining equipment from them as well; they have very competitive pricing and are happy to demo whatever we are purchasing in store as well as answer any of our questions.

Justin had a similar web controlled relay project completed in fall quarter which gave some insight into how our final communication between web-server and Raspberry Pi would work for remote control. He has also has been talking with Luca De Alfaro in the Computer Science department about whether it will be best to adapt last quarter's Web2py based project for our uses or start from scratch with another web framework such as Django or Drupal. Additionally, it is still not clear if approaching this with a web server hosted externally is the best solution as the Raspberry Pi can act as a fully fledged web server.

Design

Key Features

ACAMP's ability to effectively monitor and control the environment is due to several key features, first and foremost being the structure that houses our project. The dimensions of the frame are 27 x 43 x 78 inches and it is made out of redwood. Additionally, the structure stands on 2 inch wheels that allow the otherwise heavy structure to be moved with relative ease, putting the total height at 80 inches. These dimensions were chosen such that the structure is small enough to fit through smaller-than-average doorways but large enough to comfortably house a 25 x 25 inch grow tray. The inside of the structure will be coated with a material called "panda plastic" - a type of polyurethane that is black on the outside with a reflective sheet of white inside. The white side of the panda plastic allows for between 75-90% light reflectivity, allowing for greater light absorption by the plants.

In order to maximize plant growth, the project intends to automate the light and water cycles of the plants to the user's specifications. The growth environment will be monitored using a sensor that measures for temperature and humidity. Using the values from these sensors, fans and dehumidifiers can be controlled to match the desired parameters. In addition to environmental control, the water being fed to the plant will also be monitored using pH and total dissolved solids (TDS) sensors. Using solenoid valves, pH solutions and nutrients can be released to ensure that the water in the reservoir is ideal. An additional feature is the ability to detect plant height using a webcam. After the webcam determines the height of the plants, the light suspension system will be raised or lowered to stay a constant distance away from the top of the plants to ensure ideal growth conditions.

One of the most notable features of our project is the ability to monitor and control the growth environment online using a custom web UI. From this interface, the user will be able to set the parameters of their environment: light duration and height, the number of water cycles and duration, temperature, humidity, pH, and TDS. The values read from the sensors will be recorded so the history of the growth cycle can be viewed. Furthermore, the user will be able to view their growth environment via the webcam. We believe that this level of digital interaction truly sets our project apart.

Components

Physical Structure

The goals of our project demand a controlled environment that works to aid the hydroponic growth process. This means our physical design must face several considerations. The structure must be strong enough to hold the water reservoir and grow table, in addition to supporting the light suspension system. We are using redwood to build this structure because it is a strong wood capable of handling the weight of the project and it can handle potential environmental strains. The wood will be treated with a water resistant stain to better control humidity. Lastly, our structure will have a shelf near the top of the structure where most of the control electronics will be stored.

We will be insulating the room using panda plastic, which prevents outside light from entering while preserving the artificial light inside. In comparison to Mylar sheeting for light reflectivity, panda plastic produces fewer hot spots, making the panda plastic a more suitable insulator for our project. We have yet to decide on how to fully insulate the environment from the outside, but we have discussed using fiberglass or expanding foam between two layers of wood paneling.

Data Management Systems

Since our project requires the ability to communicate with the user over the Internet, we needed to pick a communication interface that could connect via Ethernet. After discussing with several graduate students who were working on developing an Ethernet port for a microcontroller, we determined that we needed a pre-built Ethernet connection. For this, we chose the Raspberry Pi. The Raspberry Pi has a built in Ethernet port, 2 USB ports, 17 GPIO pins, and is a fully functioning Debian Linux platform capable of running Python scripts. So far, it has been remarkably easy to connect to the Internet using the Raspberry Pi. When deciding on a microcontroller, we took into account several factors:

- At least 256k Flash memory as we expect our project to process a lot of data
- At least 4 UARTs to connect; one to connect to the Raspberry Pi, and at least 3 for sensors
- Ample GPIO
- Good online support for unforeseen obstacles

Using these criteria, we determined that the Atmel ATMega2560 was the best microcontroller for our needs. The specifications of the microcontroller match the required amount of Flash, UARTs, and GPIO. Additionally, Atmel has a strong online community that has proven itself to be an invaluable resource. For our development board, we decided to use the Arduino Mega 2560 which is written to using SPI from Atmel Studio 6. One benefit of using this board is that we are able to communicate serially through USB with the Raspberry Pi. For all of the aforementioned reasons, the ATMega2560 has been a good choice as our microcontroller.

Sensors

Our project requires that we are constantly monitoring the environment. Four different parameters will be measured: pH, TDS, temperature, and humidity. We will be measuring pH using the aptly named “pH Kit” made by Atlas Scientific. This sensor can detect pH levels from 0.01 to 14.00 and comes with a chip that communicates with the microcontroller via UART. The sensor only needs to be calibrated about once a year and can be run continuously without affecting accuracy. Due to the ease of communication and very low maintenance requirements, we chose to work with this particular pH sensor. After working with Atlas Scientific’s pH sensor, we decided to also use their “Conductivity Kit” as it functions largely the same as the pH sensor while decreasing debugging time spent on other sensors. This sensor can give us three different values: electrical conductivity (μs), total dissolved solids (KCl), and salinity (pss). By parsing this data, we can provide the user with a wealth of data pertaining to the water quality.

To measure both temperature and humidity, we are using the DHT22 temperature/humidity sensor made by Aosong Electronics. The temperature is measured in Celsius with two decimal places for added resolution; the humidity sensor has the same resolution and is a humidity percentage measurement. We picked this sensor because it is accurate and reliable as a long-term sensor. Additionally, there are several libraries already written by the distributor, Adafruit Industries, which communicate directly with the Raspberry Pi.

We plan on using a photo diode to determine whether or not the light in the room is on. Information from this sensor can be used to tell the user if their light bulb is no longer working. Lastly, we plan on using a webcam to determine the height of the plant. The Raspberry Pi will take the image from the webcam and compare the current image of the plant to previous images to determine the change in plant height. From this, we will know how much to adjust the light suspension system.

Actuators

In order to control the environment, we must make use of several actuators that adjust various parameters of the environment. First, we needed a water pump that can fill our grow table quickly but slow enough that it wouldn't overflow. For this, we decided to go with EcoPlus's ECO-264 submersible pump, which pumps water at 290 gallons per hour. The pump only has on-off states, making it easy to control, and was relatively cheap.

Next, to control the pH and TDS of the system, we will be plumbing solutions for each of these into the reservoir using solenoid valves. We are using the Ehcotech BBTF-CD-12VDC solenoid valve for several reasons. First, it has a slow flow rate which means we can distribute the solutions slow enough to ensure we don't overshoot our intended pH and TDS concentrations. Second, it only uses two wires that don't depend upon polarity to turn on, making it simple to use. The other side of the solenoid will be connected by a tube that goes into the water reservoir at the base of the grow table.

To adjust the temperature of the system, we intend on controlling the speed of the exhaust fan which should adequately regulate temperature. Because our room is so small, most inline exhaust fans can circulate air much faster than what we need. We have yet to figure out which fan we intend on using, but the fan must have a minimum speed of 15 CFM and anything above that would work for cooling the system.

In order to adjust the height of the light, we plan on using a winch and pulley system powered by a servo motor. Once the height of the plant is determined, the servo should either raise or lower the light hood. We plan on having the light ballast separate from the light hood so that there will be less weight that this system has to hold.

The last actuator we intend on using is a dehumidifier. We haven't decided upon a dehumidifier yet but our main design concerns are size and cost. The reservoir of whatever dehumidifier we purchase will be connected back into the main water reservoir for the growth tray, so we do not need a very large dehumidifier. Instead, we need to ensure there is enough space for it in the system, so we need a dehumidifier that is strong enough to handle the humidity of our system but small enough to fit comfortably.

Peripherals

The success of this project relies on several other components not directly linked to the control structures. To begin with, we will be oxygenating the main water reservoir with the EcoPlus ECO-AIR-1 air pump. This air pump was chosen as it was recommended by Santa Cruz Hydroponics to work well with our water pump. Next, we plan on using the ATX power supply to provide the power for our project. The ATX power supply is capable of supporting three different DC voltages: 3, 5 and 12 volts. Additionally, the power supply's current ratings are more than capable of handling the load we intend on using. To power the light, we plan on using a digital ballast that is capable of supporting both high pressure sodium and metal-halide lights. Lastly, we plan on installing an oscillating fan near the middle of the structure. The oscillating fan strengthens the stems of plants by recreating the wind. For this, we need a cheap 12V oscillating fan that doesn't blow too fast, as that can damage the plants.

Diagrams

Found in **Appendix A** is a System Overview Block Diagram which shows the general principles behind how our project works. The microcontroller will be in charge of reading most of the data, controlling all of the actuators, and sending this data to the Raspberry Pi. From there, the Raspberry Pi will upload all of this information online and check for changes in the user's settings. Each subsystem can be viewed in the subsequent pages of **Appendix A**. Additionally, a flow chart demonstrating our software plan can be found in **Appendix B**.

Division of Labor

Name	Lead	Assisting In
Justin Johnson	• Physical Design • Gannt Chart • PCB Design • Web Design	• Solidworks
Daniel Gunny	• Temperature/Humidity sensor • Exhaust Fan • Dehumidifier • Power Supply	• pH/TDS sensors
Wayland He	• Solidworks Design • Webcam Image Processing • Co-Lead Light Adjustment System	• Code Supervision
Stark Pister	• pH/TDS sensors • Solenoid Release Valves • Co-Lead Light Adjustment System • Code Supervision	• Web Design

Tests Conducted

We conducted several simple tests this quarter to check the functionality of each of the sensors we got working. In addition we tested the pump timers and control implemented on the ATMega2560. To test the pump, we simply timed the duration of the power cycle after the power switch had been flipped and it measures correctly to the time set in the program. To test the pH and TDS sensors, we first had to calibrate them using the calibration solutions provided by Atlas Scientific. After receiving confirmation from the sensor chips that they are calibrated, we set up a neutral water bath for the sensors and added different solutions to check their effects on the sensor data we received online. We added vinegar to lower the pH and baking soda to raise it again. Similarly for the TDS we added salt to raise the salinity and purified water to bring it back down. The DHT22 sensor came pre-calibrated; we tested it by raising and lowering the temperature with a space heater. This also changed the humidity by blowing hot dry air over the sensor.

Progress

Goals Accomplished

This quarter, we managed to finish most of our intended goals laid out in our Gantt Chart, which can be found in **Appendix D**. To begin with, we managed to get our pH, TDS, and temperature/humidity sensors to provide us with information. Additionally, we managed to get the ATMega2560 to communicate with the Raspberry Pi. Once this was accomplished, we were able to upload all of the sensor data to a spreadsheet in Google

Docs via the Raspberry Pi. We have also managed to automate the water and air pumps to run for specified periods of time. Lastly, we have managed to build the frame and finished the preliminary design of the physical structure.

Goals Currently Being Worked On

We are currently working on three different aspects of our project. First, we are learning how to operate the solenoid release valves. This will allow us to control the pH and TDS content in the water reservoir. Second, we are working on enabling the ATX power supply so that we can begin creating power rails for the whole project. Third, we are planning our PCB layout so we can start system integration in the coming months.

Budget

Our original budget was for just over two thousand dollars at the beginning of the quarter. Of that we have so far spent close to nine hundred dollars. As the quarter progressed however, the requirements for a complete hydroponics system became clearer and we have been forced to reevaluate our budgeting. We will have a new budget for the next quarter as a comparison in order to show the final projections based on the fluid design and pricing we found. For example, our pH sensor was cheaper than expected but the TDS sensor was more. We vastly underestimated the price of lighting hoods and budgeted extra for the microcontroller. We hope to keep the new budget around two thousand but anticipate closer to two thousand five hundred as the total cost. The budget tracks the amounts we have spent so far in each category and displays overages and under budgeting to give a good idea of what we have left. We also keep track of who purchased which items to ensure we can come to a fair division of costs. The budget can be found in **Appendix C**.

Accomplishments

We are proud of many of our accomplishments this quarter. The first challenge we faced was getting the pumps to work. We spent many hours trying to debug the switch setup and interface to the microcontroller. We ran into issues we didn't understand at first and so overcoming them boosted morale. Once we put everything together and solved our issues, it was incredibly gratifying to see the pump respond as expected to our commands and fill up the grow tray appropriately.

The serial communication was initially a daunting task, but we pored over all the documentation for both the Atlas Scientific parts and the ATMega2560 UART app notes. After spending hours trying to correct bugs like forgetting to append carriage returns to each transmission, we got the serial communication between ATMega2560 and the Atlas Scientific pH sensor working. Seeing the output from the sensor show up in the terminal on the computer was the first big step in our basic communications.

Once we got the pH sensor working, it was simpler to set up the TDS sensor, but the temperature/humidity sensor gave us special trouble due to the complications of timing using the single wire interface. We spent too much time trying to debug the associated C codes; when we realized we could just talk to it using the Raspberry Pi, several sighs of relief and high fives were exchanged.

The most tangible of our accomplishments was getting a frame put together so we could begin to better visualize the project. Wayland's father came to Santa Cruz at the last minute and put together the framework for our enclosure. He pre-cut all the wood and assembled it in our lab. We now have a place to begin placing components and ensuring they match the Solidworks designs.

Problems

Plant Height Detection System

Our initial design for the plant height detection system involved using a laser to determine the highest point that the plant blocked the light. The laser would sweep horizontally until it had scanned the length of the grow tray and then decrement vertically and perform the sweep again. However, this system turned out to be unfeasible and an alternate solution was required. We have now decided to determine the height of the plant by processing images acquired from a webcam using the Raspberry Pi. This benefits our project as we no longer have the space to fit in a scanning system as the dimensions of our project have changed. However, none of us have experience with image processing; subsequently, much of next quarter must be spent getting this to work.

DHT22

When we began work with the DHT22, we tested the chip using Arduino code provided by Adafruit Industries. However, when we implemented this code using Atmel Studio 6 and writing it in C, the sensor would not respond to our requests for data. After five different total overhauls of the code, we determined that too much time was being wasted on getting this sensor to work. Instead, we found a Python library that works well with the Raspberry Pi and got the sensor working within 20 minutes of programming.

Door Restrictions of Our Enclosure

Originally, we had planned for a larger enclosure so we could comfortably fit a 3x3 ft. grow table in our enclosure. We realized that in order to fit the enclosure through a door, we had to make it smaller. We decided instead to use a 2 x 2 ft. grow table, allowing us to make the enclosure a little smaller. With some careful measurements, Wayland's dad helped us design an enclosure that fits through all standard doors. Additionally, wheels were added, allowing the otherwise heavy structure to become mobile.

Counter EMF of Pumps

Once we had written code to get the water pump working, we tried testing it by connecting the power of the pump through a switch-relay that was controlled by our board. A switch was then flipped on and the water pump ran for the desired ten seconds of testing. However, once the timer shut off the pump, the switch (which was being read as "high") went low briefly and the system would reset itself once it acknowledged the switch's true state. After researching relays, we determined that there was a harsh counter EMF that was being generated when the relay was suddenly shut off. In an effort to fix this error and protect our board from damage, we created a circuit that uses two diodes and a bypass capacitor to prevent most of the counter EMF. We also created new code to deal with this occurrence which doesn't go back to the initialized "OFF" state without polling the switch a few seconds after the pump shuts off.

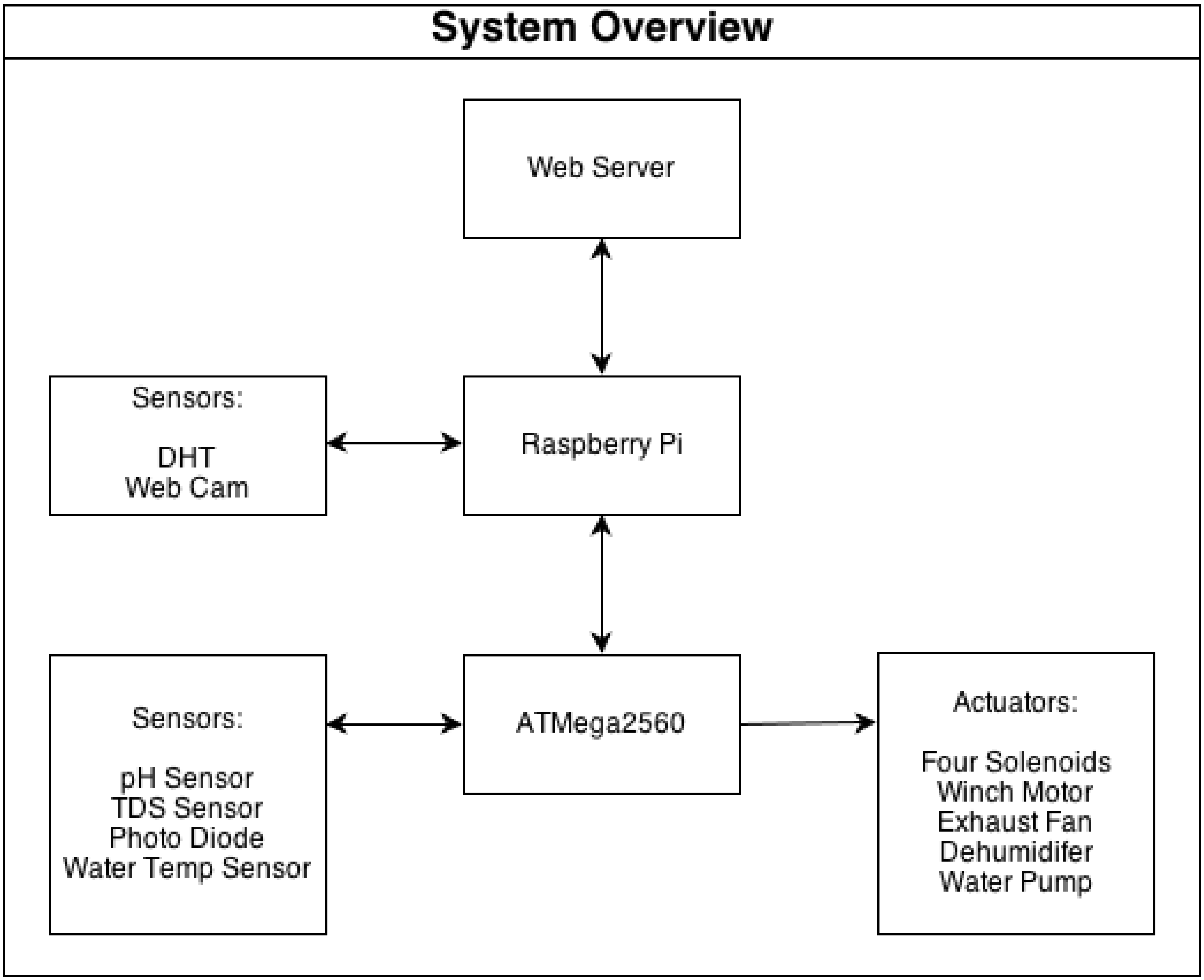
Conclusion

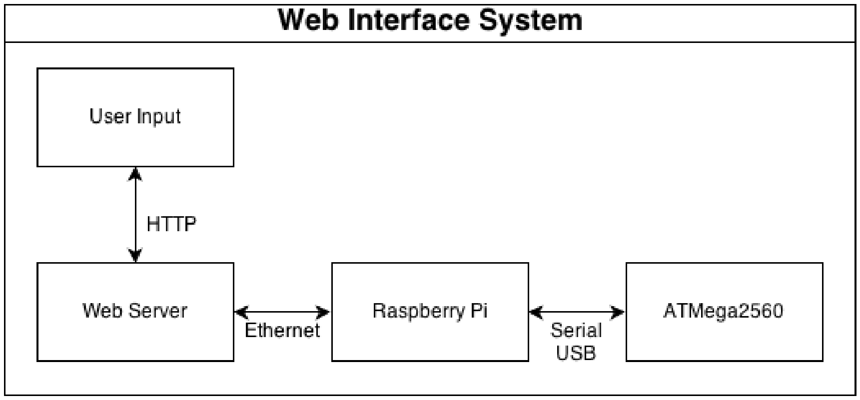
Next quarter, we are planning on installing the light system for our enclosure which will be held up by a winch and pulley system. We are also planning on building the exhaust system, complete with a 3-4 inch inline fan. In order to control the temperature and humidity of the project, we will begin controlling the exhaust fan as well as a dehumidifier. We have already purchased solenoid valves for our pH and nutrient release system and plan on connecting those to our water reservoir. Additionally, we will begin setting up the web interface to communicate with the Raspberry Pi. Lastly, we will begin image processing using a webcam. Most of our time next quarter will be spent integrating all of our individual systems.

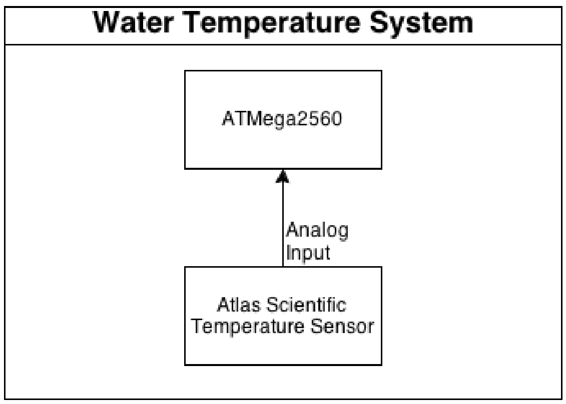
If time permits, we would like to add an LCD screen to our enclosure so that someone can control the parameters of the system without having to go on the website. We also would like to add a carbon dioxide sensor as well as a water level sensor in the grow tray. Lastly, we'd like to have the oscillating fan move with the light system so that the fan remains focused at the central height of the plants perpetually.

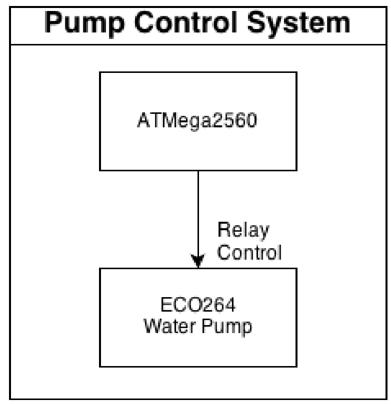
We believe that we are on track for completing this project. We finished all of our goals for this quarter. Our main concern is the lack of external funding for this project. We are hoping to be able to get some sort of funding so that we don't have to pay everything out of our pockets. We are trying to recuperate these costs by entering it into an entrepreneurship competition at the end of May. We are excited to begin the second half of our project.

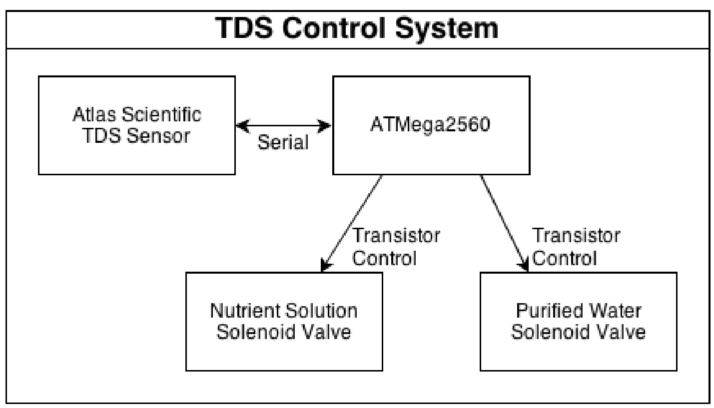
Appendix A

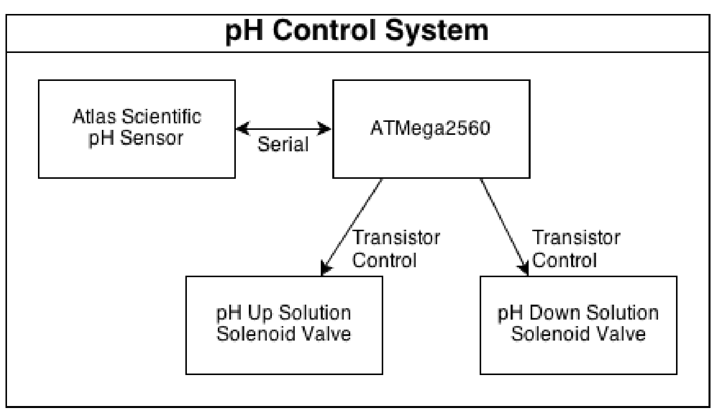

System Overview
Web Server
Sensors:
DHT
Web Cam
Raspberry Pi
Sensors:
pH Sensor
TDS Sensor
Photo Diode
Water Temp Sensor
ATMega2560
Actuators:
Four Solenoids
Winch Motor
Exhaust Fan
Dehumidifer
Water Pump

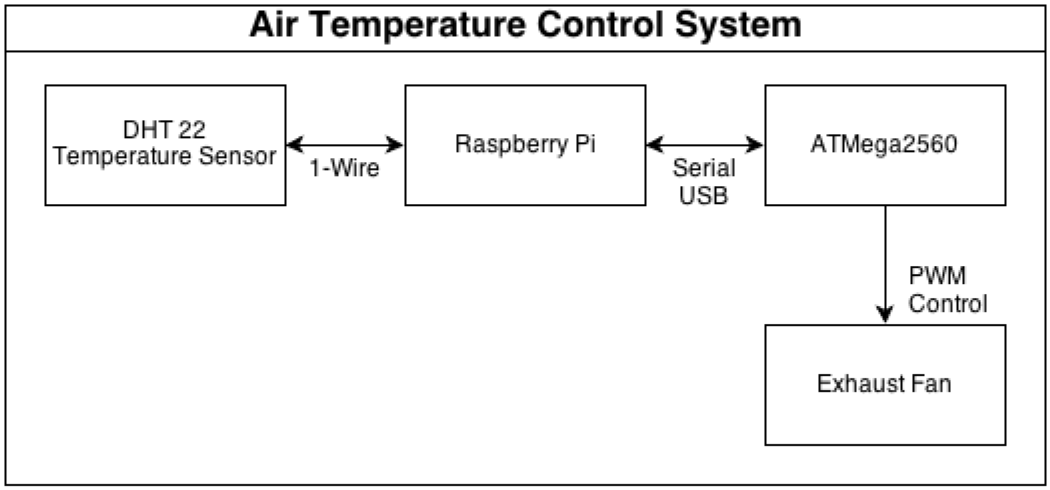

Web Interface System
User Input
HTTP
Web Server
Ethernet
Raspberry Pi
Serial USB
ATMega2560

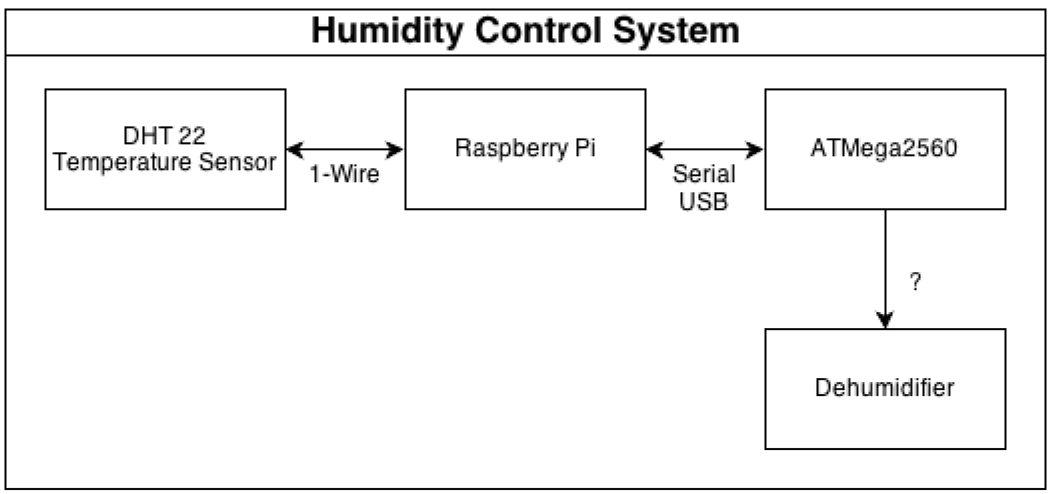

Water Temperature System
ATMega2560
Analog Input
Atlas Scientific Temperature Sensor

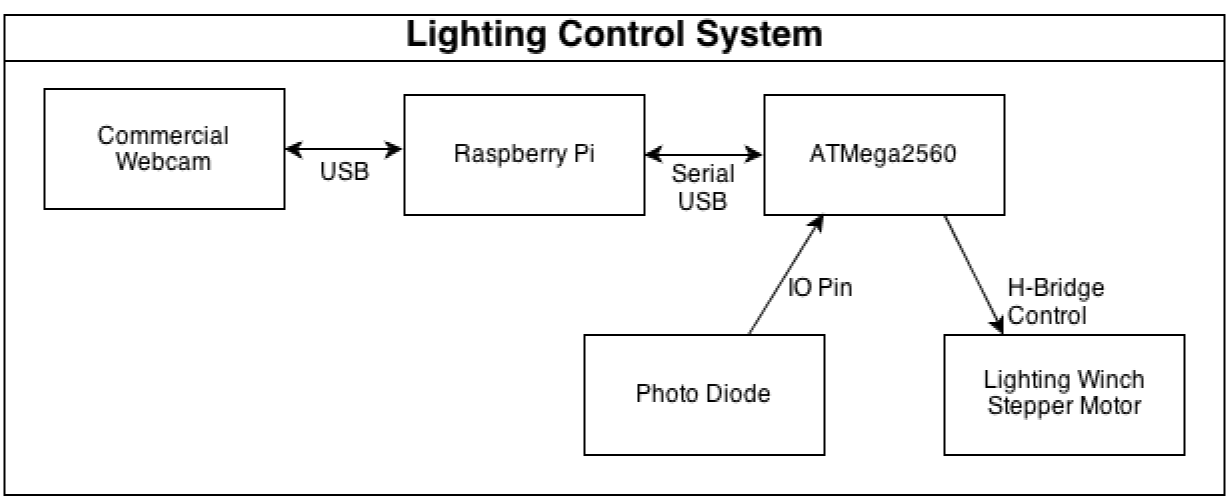

Pump Control System
ATMega2560
Relay Control
ECO264 Water Pump


TDS Control System
Atlas Scientific TDS Sensor
Serial
ATMega2560
Transistor Control
Transistor Control
Nutrient Solution Solenoid Valve
Purified Water Solenoid Valve

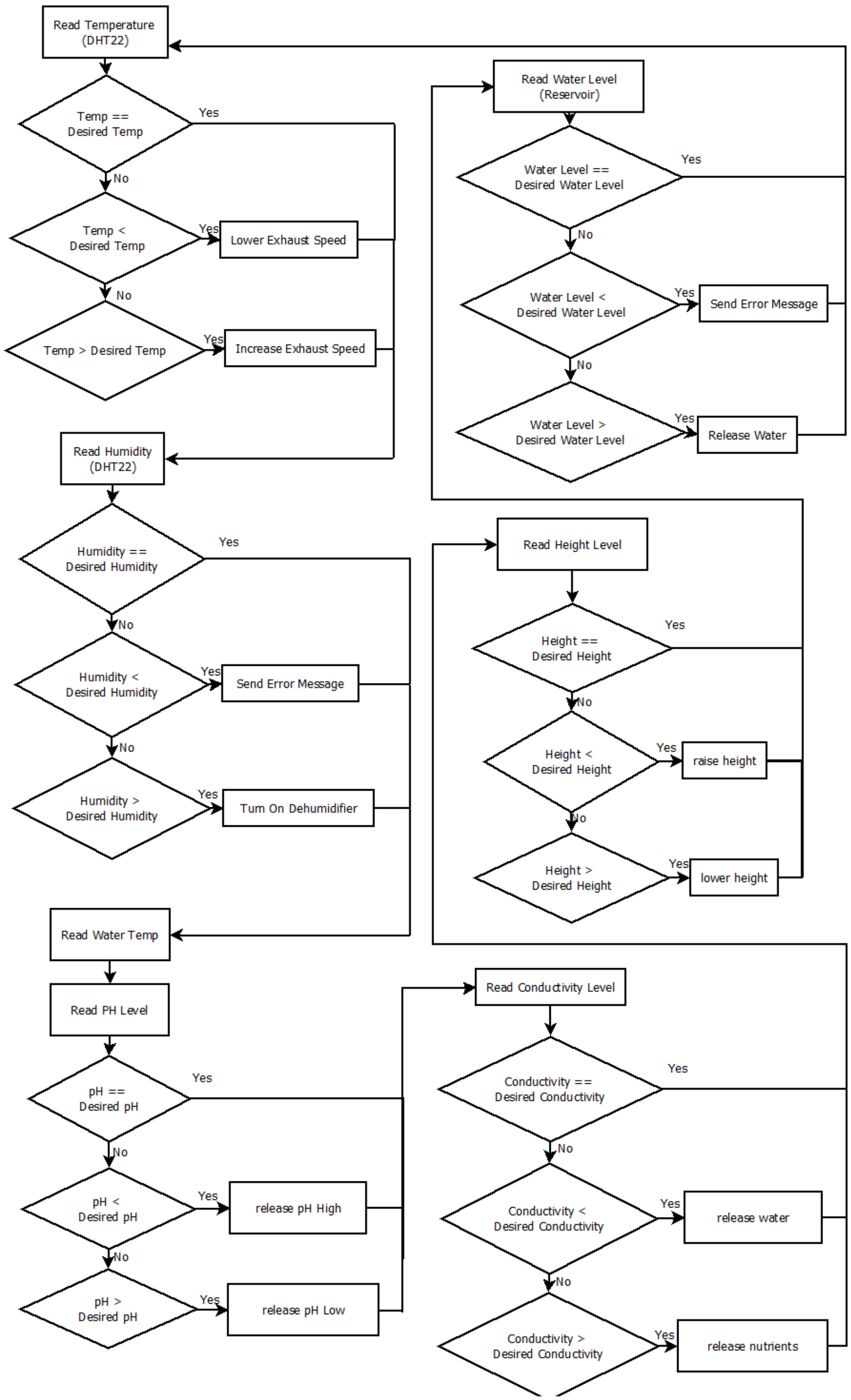

pH Control System
Atlas Scientific pH Sensor
Serial
ATMega2560
Transistor Control
Transistor Control
pH Up Solution Solenoid Valve
pH Down Solution Solenoid Valve


Air Temperature Control System
DHT 22
Temperature Sensor
1-Wire
Raspberry Pi
Serial
USB
ATMega2560
PWM
Control
Exhaust Fan


Humidity Control System
DHT 22
Temperature Sensor
1-Wire
Raspberry Pi
Serial
USB
ATMega2560
?
Dehumidifier


Lighting Control System
Commercial
Webcam
USB
Raspberry Pi
Serial
USB
ATMega2560
IO Pin
H-Bridge
Control
Photo Diode
Lighting Winch
Stepper Motor

Appendix B


Read Temperature (DHT22)
Temp == Desired Temp
Yes
No
Temp < Desired Temp
Yes
Lower Exhaust Speed
No
Temp > Desired Temp
Yes
Increase Exhaust Speed
Read Humidity (DHT22)
Humidity == Desired Humidity
Yes
No
Humidity < Desired Humidity
Yes
Send Error Message
No
Humidity > Desired Humidity
Yes
Turn On Dehumidifier
Read Water Temp
Read PH Level
pH == Desired pH
Yes
No
pH < Desired pH
Yes
release pH High
No
pH > Desired pH
Yes
release pH Low
Read Water Level (Reservoir)
Water Level == Desired Water Level
Yes
No
Water Level < Desired Water Level
Yes
Send Error Message
No
Water Level > Desired Water Level
Yes
Release Water
Read Height Level
Height == Desired Height
Yes
No
Height < Desired Height
Yes
raise height
No
Height > Desired Height
Yes
lower height
Read Conductivity Level
Conductivity == Desired Conductivity
Yes
No
Conductivity < Desired Conductivity
Yes
release water
No
Conductivity > Desired Conductivity
Yes
release nutrients

Appendix C

ACAMP Budget

	Part	Qty	PPU	Total	Spent	Remaining
Controller						
	ATmega 2560	2	20	$40.00	0.00	$40.00
	Arduino	2	50	$100.00	-42.95	$57.05
	Raspberry Pi	2	60	$120.00	-59.95	$60.05
	PCB Runs	2	150	$300.00		$300.00
	Web Hosting	1	50	$50.00		$50.00
Casing						$0.00
	Enclosure	1	100	$100.00		$100.00
	Threaded Rods	6	5	$30.00	-6.90	$23.10
	Ducting	1	20	$20.00		$20.00
	Tubing/Plumbing	1	20	$20.00	-9.17	$10.83
	Plant Tray	1	50	$50.00	-53.84	($3.84)
	Reservoirs	3	20	$60.00	-16.26	$43.74
	Discrete Electronics	1	100	$100.00	-49.46	$50.54
	Assorted Structure	1	50	$50.00		$50.00
Systems						$0.00
Plant Height:	Laser Diodes	2	5	$10.00	-22.95	($12.95)
	Receivers	2	5	$10.00		$10.00
	Stepper Motor	2	25	$50.00	-20.66	$29.34
	Controller	2	5	$10.00		$10.00
Temp/Humidity:	Temp/Humidity Sensor	1	15	$15.00	-14.95	$0.05
	Heating Element	1	10	$10.00		$10.00
	Oscillating Fan	1	15	$15.00		$15.00
	Exhaust Fan	1	150	$150.00		$150.00
	Humidifier	1	50	$50.00		$50.00
	Dehumidifier	1	70	$70.00		$70.00
pH:	pH Sensor	1	120	$120.00	-105.95	$14.05
	Servo Motor	2	10	$20.00	-29.70	($9.70)
	pH Solution	1	20	$20.00		$20.00
TDS:	TDS Sensor	1	50	$50.00	-166.46	($116.46)
	Servo Motor	2	10	$20.00	-29.70	($9.70)
	Nutrients	1	40	$40.00		$40.00
Light:	Photosensor	1	5	$5.00		$5.00
	Light Enclosure	1	10	$10.00		$10.00
	Light Bulb	2	40	$80.00		$80.00
	Reflective Coating	1	20	$20.00		$20.00
	Servo Motor	2	20	$40.00		$40.00
Pump:	Ebb/Flow Pump	1	30	$30.00	-22.14	$7.86
	Air Pump	1	10	$10.00	-9.29	$0.71
Totals					-660.33	Categorized
	Subtotal:			$1,895.00	660.33	Total Spent
Padding		10%	0.1	$189.50		
	Total Budget ---			**$2,084.50**	**Remaining ---**	**$1,424.17**
			Each:	$521.13	Under:	$179.77
					Over:	($152.65

Appendix D

Task Name	Duration	Start	Finish	Predecessors	Resource Names
Complete Gantt Chart	2 days	Sat 1/12/13	Sun 1/13/13		
Solid Works Design	16.75 days	Thu 2/7/13	Sat 2/23/13		
Woodworking	3 days	Sun 2/24/13	Tue 2/26/13	2	
Case Assembly	5 days	Wed 2/27/13	Sun 3/3/13	3	
Case Prototype Done	0 days	Sun 3/3/13	Sun 3/3/13		
Hydroponic System Integration	**7 days**	**Fri 2/8/13**	**Thu 2/14/13**		
Reservior Integration	2 days	Sat 2/9/13	Sun 2/10/13		
Water Table Integration	3 days	Sun 2/10/13	Wed 2/13/13	7	
pH Sensor	**19.75 days**	**Sat 2/16/13**	**Thu 3/7/13**		
pH Sensor Design	10.75 days	Sat 2/16/13	Tue 2/26/13		
PH Sensor Prototype	3 days	Tue 2/26/13	Fri 3/1/13	10	
pH Sensor Integration	6 days	Fri 3/1/13	Thu 3/7/13	11	
Laser System	**14 days**	**Mon 2/11/13**	**Sun 2/24/13**		
Laser System Design	7 days	Mon 2/11/13	Sun 2/17/13		
Laser System Prototype	3 days	Mon 2/18/13	Wed 2/20/13		
Laser System Integration	2 days	Thu 2/21/13	Fri 2/22/13	15	
Microcontroller	**20.75 days**	**Sun 1/20/13**	**Sat 2/9/13**		
Decide uC	4 days	Sun 1/20/13	Wed 1/23/13		
Microcontroller Procurement	5 days	Thu 1/24/13	Tue 1/29/13	18	
Temperature / Humidity	**4 days**	**Sat 3/2/13**	**Tue 3/5/13**		
Temperature Sensor Prototype	1 day	Sat 3/2/13	Sat 3/2/13		
Humidity Control Integration	2 days	Sun 3/3/13	Mon 3/4/13		
Humidity Control Prototype	1 day?	Sat 3/2/13	Sat 3/2/13		
Temperature Sensor Integration	2 days	Sun 3/3/13	Mon 3/4/13		
Nutrient Sensing	**6 days**	**Fri 3/8/13**	**Wed 3/13/13**		
TDS Sensor Prototype	2 days	Fri 3/8/13	Sat 3/9/13		
TDS Sensor Integration	2.25 days	Sat 3/9/13	Sun 3/10/13		
All Sensors Integrated	0 days	Sun 3/10/13	Sun 3/10/13		
Pumps & Irrigation	**8 days**	**Thu 2/21/13**	**Thu 2/28/13**		
Air Pump Integration	1 day	Fri 2/22/13	Sat 2/23/13		
Water Pump Impementation	2 days	Thu 2/21/13	Fri 2/22/13		
Irrigation System Implementation	3 days	Fri 2/22/13	Mon 2/25/13		
Water Filter Integration	2 days	Thu 2/21/13	Fri 2/22/13		
Lighting	**14 days**	**Tue 3/5/13**	**Mon 3/18/13**		
Ballast Design	7 days	Tue 3/5/13	Mon 3/11/13		
Ballast Integration	3 days	Thu 3/14/13	Sat 3/16/13		
Lighting Control Design	3 days	Fri 3/8/13	Sun 3/10/13		
Lighting Control Integration	4 days	Thu 3/14/13	Sun 3/17/13		
Fans & Exhaust	**5 days**	**Fri 3/15/13**	**Tue 3/19/13**		
Fan Integration	3 days	Fri 3/15/13	Sun 3/17/13		
Air Filter Integration	2 days	Fri 3/15/13	Sat 3/16/13		
Printed Circuit Board	**15 days**	**Sat 3/23/13**	**Sat 4/6/13**		
PCB Layout	9 days	Sat 3/23/13	Sun 3/31/13		
PCB Acquisition	3 days	Mon 4/1/13	Wed 4/3/13		
PCB Integration	3 days	Thu 4/4/13	Sat 4/6/13		
Web Interface	**30 days**	**Sun 4/7/13**	**Mon 5/6/13**		
Web Control Design	5 days	Sun 4/7/13	Thu 4/11/13		
Website Construction	16 days	Fri 4/12/13	Sat 4/27/13		
Website Integration	9 days	Sat 4/27/13	Sun 5/5/13		
Physical Prototype Done	0 days	Sun 3/17/13	Sun 3/17/13		
Rasperry Pi Integration	12.13 days	Mon 3/25/13	Fri 4/5/13		
Web Cam Design	4 days	Sun 4/7/13	Wed 4/10/13		
Web Cam Integration	2 days	Sat 4/13/13	Sun 4/14/13	52	
Freeze Physical Design	0 days	Sat 3/30/13	Sat 3/30/13		
Freeze Software Design	0 days	Sun 4/21/13	Sun 4/21/13		
Integration and Testing	21 days	Thu 5/16/13	Wed 6/5/13