P-QUAD: Portable Plant Preservation Pod

University of Central Florida

Department of Electrical Engineering and Computer Science

Senior Design I 10/07/2018

Dr. Lei Wei & Dr. Samuel Richie

Group 4

Alexander Meade – Computer Engineering

Kareem Rahounji – Electrical Engineering

Syed Bukhari – Electrical Engineering

Nicholas Poidomani – Computer Engineering

Table of Contents

1. Executive Summary

The transportation of plants is often an overlooked aspect of horticultural industries such as nurseries or botanical gardens. To mitigate this problem, people often treat it as a logistics issue and ship multiples of the same plant in a shipment to ensure that if some are damaged or perish in the journey then the customer still can receive the plant they ordered, but what if the plant they ordered is of a rare and expensive variety or if someone interested in transporting a particular specimen for scientific inquiry. These cases it may be cost inefficient or outright impossible to take the standard approach of just shipping multiples of the plant. We are proposing instead of treating the issue like it traditionally is to take a more technologically sound approach via our proposed Portable Plant Preservation Pod (P-Quad). This preservation system acts as a safe housing enclosure and monitor and response system in order to assure the safe relocation of the plant in question. This project could help revolutionize the way industries often attack these issues surrounding plant management by encouraging an approach not from the logistical side, but rather as quality assurance issue. While this may start as an increase overhead it did ultimately lead to a cheaper and more secure method which can become normalized due to intended scalability of our design and product.

Traditionally the transpiration of plant life in many ways is a fragile and delicate process. The plants that need to undergo transport are typically exposed to little or no light, a lack of water, improper humidity levels, and otherwise unfavorable conditions. Our objective is to build a prototype of a P-Quad for a small plant to be sustained and monitored over an extended time interval. Most plant monitoring systems are connected to 120V AC current sources and our goal is to provide a cost efficient, scalable, and portable version that can not only monitor the plant's vitals but also sustain the plant for a short time period.

The brains of our pod are the electronic sensors. The sensors job is to keep track of all different environment variables such as: temperature, humidity, soil moisture, and light intensity. These sensors then relay the data to a microcontroller that regulates different pumps and lights to satisfy the plants needs at a given time. Our power to the pod and various subsystems are utilizing AC power when the pod is stationary. This is interchangeable with a secondary power system integrated in the pod that is able to use DC power when the pod is being transported. The backbone of P-Quad's design is the wood housing enclosure. The design for the housing is a flat box housing all the electronics with a circular depression to hold the soil and plant then surrounding this depression is another box shaped enclosure that stands upright and house the plant and hold the sensors and lighting fixtures.

Currently there is nothing comparable to what we are proposing on the market today as far as portable solutions. In terms of transportation of plants, the only carrying devices are all mechanical, usually a bag or box with handles that cradles the plants as they are transported. Some models have hard cases with wheels on the bottom but none that have a full enclosure for the plant being relocated. Monitoring systems in the market today are also very limited in comparison to our design project. In the market, the most resemblance to our device comes in the form of wi-fi enabled watering systems. Our product aims to not only solve the problem of monitoring the plant while undergoing transport but also to shelter and provide correction that potentially cause harmful environmental stress on the plant.

2. Project Description

In the following chapter we discuss the project and its generalities. We explain the motivation for the idea and the project as well as our general goals and objectives we wish to accomplish with this project. We have fleshed out our requirements specifications on both by the market requirements and user end objectives as well as articulate any constraints that became apparent in the design. These constraints and specifications address both on the hardware side and in addition the software and programming side. We provided a house of quality in order to demonstrate relations between the engineering specification and the customer requirements and additionally display a block diagram and flowchart breakdown regarding the construction of our project from both a hardware and software perspective. A decision matrix is displayed in order to analyze and compare important criteria corresponding to the user requirements and finally a section which deals with the projects measurables and aim goals within the defined constraints values.

2.1. Project Motivation and Goals

The motivation behind this project was a desire to improve current methods of transportation for various flora-based industries such as botanical gardens, agricultural, or in areas of botanical research. The reasons that these fields present the idea use cases is because they all often revolve around plant life that the survival of the plant is intrinsic to the success of the business or area of research. An example highlighting this would be a field botanist who observed an unusual fungus or parasite on a specific plant species and wanted to observe the longstanding impact of the relationship between the host and parasite. However, the stress from uprooting the plant and putting it in a portable soil pot combined with any potential duress caused to the plant by its potentially harmful parasite could lead the untimely death of the plant before any proper research could be conducted. Another case might be a botanical garden received a donation of a rare and exotic plant and wish to ensure its health prior to its arrival and incorporation into the garden are as optimal as possible. In cases like these it would be paramount to have a system which accommodated the plants growth/living cycles which is where our concept of the P-Quad comes in. Looking into the future revisions of the P-Quad it could potentially see usage in agrarian industries for instance, bananas growth cycle is very delicate and demanding in addition they are often highly prone to bacterial infection and diseases. With an upscaled version of the P-Quad fitted to house the entire tree the bananas could grow in an environment that not only matches their specific climate needs (highly tropical conditions), but as well filter out any diseases and infections that could potentially reduce crop. Even in the event of one unit failing to perform properly the diseased tree would be isolated from the other trees reducing the negative impact on the crop yield.

2.2. Objectives

Our objectives for this design are most largely proof of concept in terms of feasibility and the designs theoretical scalability. In order to prove the design is capable of preservation of plant life for an extended duration, we plan to demonstrate with a plant kept in the preservation system for at least a day and to showcase the portability aspect we plan to have the preservation system fully run off battery for at least 1 hour. Additionally, we would want the device to able to plug into a wall outlet to so that the battery may recharge or that during extended stays, say for instance a pod was to be stuck in a building while shipping details for it were worked out, the battery would not have to see continual use and remains charged up and ready for when it is truly needed..

2.3. Requirement Specifications

In the tables 1 and 2 below we itemize both our target market requirements and the objective and constraints. This includes the engineering specifications with what we observed to be good metrics to meet and the constraints which limit it us in our design capability.

Table 1: Market Requirements

The electronics device is able to run for a minimum of 3 hours on battery life.
The device size is within a 2 ft x 2 ft x 3ft rectangle
The device deposit 2 oz. of water in 12-hour intervals.
The device weighs less than 20lbs.
The device provides 250-1000 lumens during the day time cycle.
The cost of the device is under $600.

Table 2: User Objectives and Project Constraints

Low Cost
Easily Portable
Small Size
Low Battery Consumption/Long Battery Life
Plant Environment Monitoring
Soil Moisture Control
Light Level

2.3.1. Software Specifications

The software controlling the system is an event state driven that remains in a low power/ monitoring state with periodic sensor reads states throw in. If a sensor detects it is at a threshold for its given values the system triggers an event to switch the system into a feedback/alert mode to inform the systems users that the plant has passed into unideal preservation conditions.

2.3.2 Software Constraints

There are a number of constraints that must be taken into account before producing the software for our device. The first constraint was dependent on the microcontroller used and that is the language that the source code was written in. Dependent on the processor that we use, we might have the option to use an object-oriented programming language, such as C++. In the event that object-oriented languages are not available to us, the language that we used using was embedded C++. Another constraint that we ran into based on the processor we use is the amount memory available to us. This includes both the program memory and data memory. Living in the embedded world requires us to use a minimal amount of memory at all times. Due to the nature of our product we did not encounter a situation where run out of memory, whether it is program or data memory. However, it is still important to understand the possible restraints from the amount of memory available in the microcontroller. These constraints are reflected in table 3 below.

The software turns on the power LED when there is power supplied to the device.
The software turns off the power LED when power is removed from the device.
The software controls communication with the temperature sensor.
The software controls communication with the moisture sensor.
The software controls communication with the humidity sensor.
The software controls communications with the light sensor.
The software polls all sensors every hour for potential hazards.
The software displays the current status of internal variables to the output display every hour for fifteen seconds.
The software reads in input variables from the user via the display buttons.
The software, based on input variables, controls the amount of water pumped into the device.
The software illuminates the internal lamp in intervals based on input variables.

Table 3: Software objectives and constraints

2.3.3 Hardware Constraints

With every hardware/structural design there were inevitable constraints. Since our projects holds a lot of natural elements, we must make sure the electrical components are protected at all times. Our major concern is the PCB getting in contact with water. For the initial design we created, the base was an open box with majority of the space going towards the soil and a small section barrier for the PCB itself. The fault with that design is that when the plant's soil is watered, the evaporated water over time can create a humid environment with enough vapor to short circuit the components. To prevent that we relocated either the soil chamber or the PCB. The decision would be to relocate the plant on a higher level than the PCB, so that any moisture just reciprocates in the chamber holding the plant and prevent any component failure from the brain of P-Quad, the PCB. With regards to protecting the sensors, we do not have to worry about any moisture building up in the container because their built-in feature offers that protection.

Another major constraint is delivering the wires to all needed components. Since we are going to have a sealed 3-D printed solid structure, we must take that into consideration. Strategic hole placements can allow the wires to be efficiently extended to their respective components. Instead of the structure being built and us manually drill the holes We decided to design the box to include the small round openings at the needed locations using Solidworks.

There is a main issue with the PCB wires reaching to the right component(s). Every time we open the box there is the risk of tension on the wires if they are not long enough. Making the wires longer also hinder any voltage and signal making it harder to obtain the desired results while making it shorter can cause the wire to break. There should be a middle ground to all of this and

To improve upon the lid, we were thinking of creating cylindrical cut at the center of lid. This would allow the plant's pot to be resistant to movement while being in a fixed location that is most efficient when it comes to the PCB and sensor connections. The main problem that can arise from this is the different pot sizes available on the market. If the pot is too small it moves around causing it to be in an unstable condition during transport, nevertheless if the pot is too large it cannot fit at all and eventually tip over or fall off. To solve this problem there are two of many ways, we can either determine one pot size that is compatible or even better create two different cylindrical cuts with one slightly bigger than the other. For the second approach, two cuts might be enough because the more there are the more instability it causes for smaller pots

In order to meet the consumers requirements, the efficiency of P-Quad should be 80 % or higher. And in order to achieve the customers specified requirement many different sensors have to be implemented. While the plant is inside the pod, the temperature of the internal environment must be monitored in order to make corrections to make sure that the plant is in its natural temperature. Depending on the customer desires there is a manual thermometer installed inside the pod so if the customer want to read the temperature they are able to; however, the temperature is displayed digitally in the display that was implemented in the outside of the pod if the customer does not desire to open the face of the pod. In addition to a thermometer in the inside of the pod, there is a thermometer on the outside of P-Quad as well. This is to ensure that the surrounding temperature is also measured. The thermometers collect the data from the inside of the pod and from the outside environment and compare them adjusting the temperature for the best possible environment for the plant.

P-Quad, having numerous features, required power to function; hence, a crucial consideration was power consumption of the system. When there is a specimen placed inside the pod and P-Quad is turned on, the system is provided with power with a local AC outlet. This scenario is for when the customer only like to monitor and control the specimen while not being transported. At a smaller scale we showed this by plugging in a wall outlet. During transportation P-Quad is powered by an internal rechargeable battery that can be removed. For the wall outlet an AC to DC converter is needed. The adapter is plugged in the wall and the male part of the cord is directly plugged into the PCB powering the microcontroller. To regulate the power and make sure there are no overcurrent, a switch relay was used. The value of the input voltage is around 3 to 5 volts after stepdown which in turn be enough to power all the sensor that are implemented inside the Pod. The external sensors required for the P-Quad to be efficient are ambient light sensor, temperature sensor, moisture sensor and a humidity sensor. If the microcontroller in unable to support all these sensors, then a different microcontroller was selected, or a combination of multiple microcontrollers were used, however this did not prove to be the case. Most of the hardware constraints are focused on implementing the proper sensors and the right relay switch. For the relay switch we are using a solid-state relay due to their advantage over a mechanical relay. Unlike a mechanical relay, which is limited to DC power a solid-state relay is compatible with AC power or DC power which opens many additional avenues for design. Another constraint for the pod is the heating element, has to be monitored. However, implementing a heater or a device that can heat the pod was very costly and took up a lot on space inside the Pod which in turn had much less space for much needed sensors and living space for the specimens, so in order to tackle this issue we used LED lights that not only provide the lumens for the plant but also the heat needed for the plant to survive.

For the battery implemented in P-Quad must be a cadmium lead electrode or other type of battery that complies with Title 42, United States Code, Chapter 137, Section 14301-14336, which makes it a requirement that the regulated batteries be easily removable from the rechargeable consumer product. Even though P-Quad has a standby technical service department for the consumers, all the batteries that are used in P-Quad has proper labeling including a three chasing arrows or comparable recycling symbol; furthermore, the battery compartment has the proper battery type, so the consumers know exactly which type of battery is safe to use for proper functionality of the product.

2.3.4 Hardware Specifications

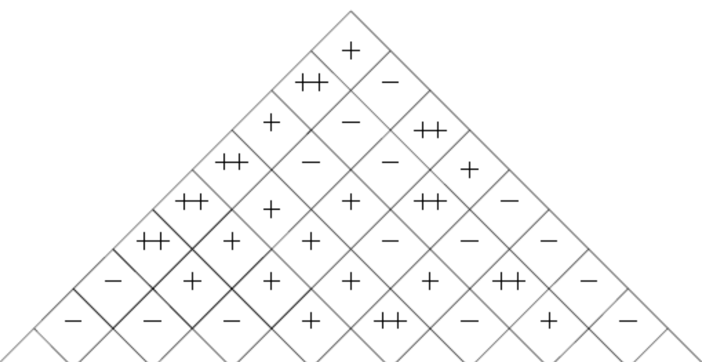
The entire design team for this project constructed the hardware specification plan. The first specification that is going to be apparent is the actual enclosure that houses the entirety of the Plant, Sensors, PCB and the pump. The actual P-Quad that is able to house many different sized plants. The Dimensions are 1.2 by 1.2 by .5 feet for the demo; however, when in actual production P-Quad should be able to hold larger specimen.

The sensors should be able to monitor and control the environment that is specified by the customer. The data that is collected by the sensors should be stored in the internal memory of the system. And the stored data should also be displayed on the screen outside the pod, so the customers can read the data and swift through the past stored data to better manage the life of the specimen. All the sensors should be installed in a manner that increases the accuracy of the reading. The sensor should not be installed solidly inside the pod, so they may not move around when P-Quad is in transit. The wiring should be implemented in accordance to National Electric Code (NEC).

The system is a collection of sensors utilizing a microprocessor unit to as a driver. A printed circuit board contained all the sensors circuitry. Finally, an attached housing unit holds the battery bank and the charging circuit to supply power to the system.

2.4 House of Quality Analysis

The house of Quality for P-Quad illustrated in figure 1 on the following page we visualize this house of quality shows the relatability between engineering specification and customer requirements. In order to begin construction of a feasible idea research is done to find out what customers' requirements are. For P-Quad we determined that cost, portability, lower power consumption, efficient monitoring, and safe data storage are some of the most important requirement the customer asked for. Now has a designer of this idea we came up with numerous engineering specifications to provide the answers to the customer requirements. Battery life, lower cost, display, power usage, memory, these are the most important specification needed to solve the customers major concerns. The house of quality also helps with the determination of how engineering requirements affect each other either a lot, which is represented with a plus sign, or does not affect each other much, which is represented with a minus sign. For example, the battery life and illumination affect each other a lot thus there are two plus signs. This makes sense because if more illumination is needed or used this drain the battery much faster. House of quality also shows the importance of customer requirements. Some of the most important customer requirements include cost, portability, power consumption, monitoring and data storage. To meet these customer requirements certain engineering specifications are implemented in the design; furthermore, a chart of "Importance" is created to set a numerical value that is added up for all the engineering requirements and that gives a number value to determine exactly how important is the certain customer requirement. In the case of P-Quad the most important customer requirement is the battery life, which makes sense because the most distinguish feature of P-Quad is the transportability of the specimen for a long distance. Another very important information house of quality gives is the measurable and testable engineering specifications. In the case of P-Quad making sure the battery life is over 3 hours, providing over 1000 lumens of light, making sure providing enough power of over 600 Watts are some of these measurable and testable engineering specifications.



IMPORTANCE 1-10 STRONG-9 ◎ MEDIUM - 3 ○ WEAK - 1 △			BATTERY LIFE	DIMENSIONS WEIGHT	EFFICIENCY	ILLUMINATION	COST	DISPLAY	MEMORY	Power Usage	Temperature Control	CORRELATION BETWEEN CUSTOMER AND ENGINEERING REQUIREMENTS
			+	−	+	+	−	−	+	−		
COST	−	7	◎	◎		○	◎	◎	△	○	△	238
PORTABILITY	+	10	◎	△	○		○					160
POWER CONSUMPTION	−	8	◎	○		◎	◎	◎	△	◎	○	336
MONITORING	+	6	○		◎	○		○	△		○	114
DATA STORAGE	+	7	△		○		△	△	◎			105
TARGETS FOR ENGINEERING REQUIREMENTS			> 6 HOURS	2x2x3 FEET <20 LBS	> 80%	> 2500 LUMENS	<500 $	3X5 INCHES	>500 MB	<600 W	20-40 C	
IMPORTANCE TOTAL			243	97	84	111	171	160	84	45	55	
Demonstrable Specifications					✓	✓				✓	✓	

Figure 1: House of Quality

2.5 Block Diagram

The block diagram shown in figure 2 on the next page gives a description of the system architecture of PQAUD. The MCU communicated between the various sensors and serialize the data to manage the various output devices such as the display, lamp, and pump. The legend at the bottom shows which group member were assigned to the equivalent portion of the project. Each color in the diagram represents the responsibility for that member or team. This block diagram represents the entire system from the battery providing power to the sensor system power circuit and also the maintenance system. From there the sensor system providing data to moisture sensor, temperature sensor and the light sensor. From these sensors the data and instructions going to the individual components like the pump fan and the lights in P-Quad.

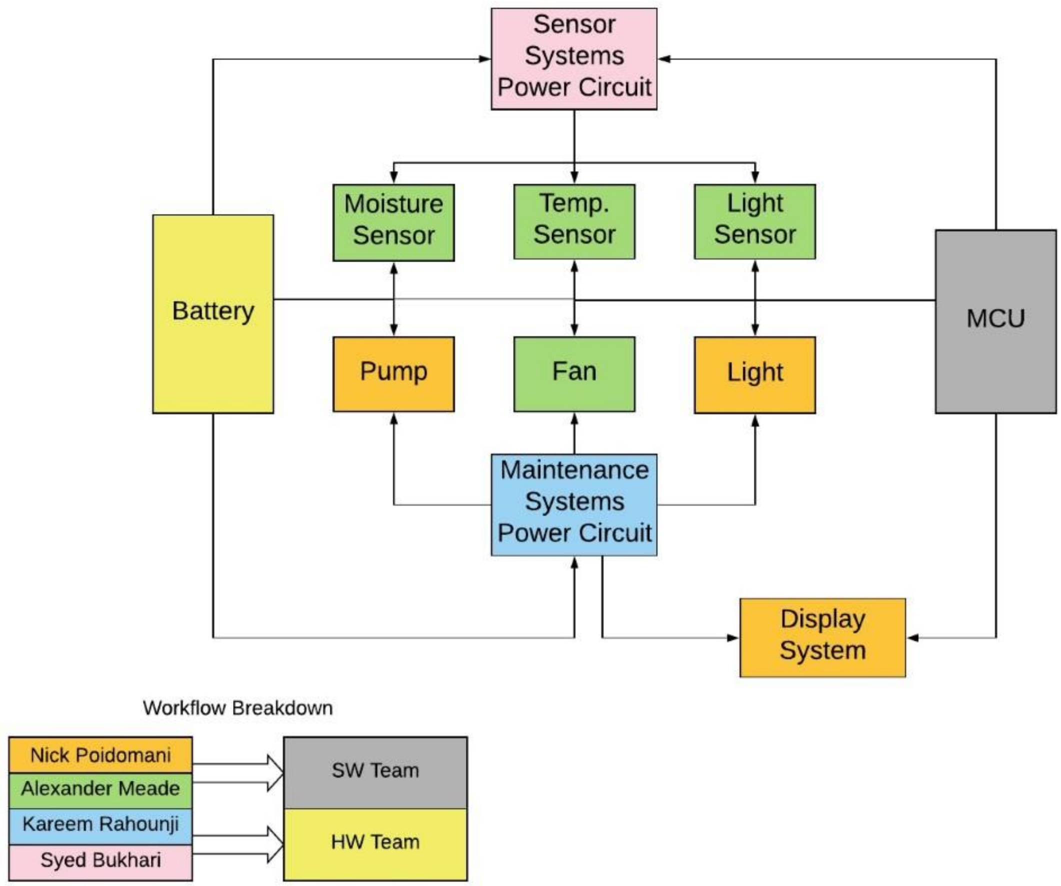


Figure 2: Block Diagram

2.5.1 Software Engineering Design Flowchart

The following image in figure 3, shows the design flowchart for the software used in the embedded electronics for P-QUADs system. The microcontroller software utilized state machine framework based on the user input as well as communication with the various sensors contained in P-QUADs environment. The only interaction the user has with the system is when the environment variables and conditions variables are set.

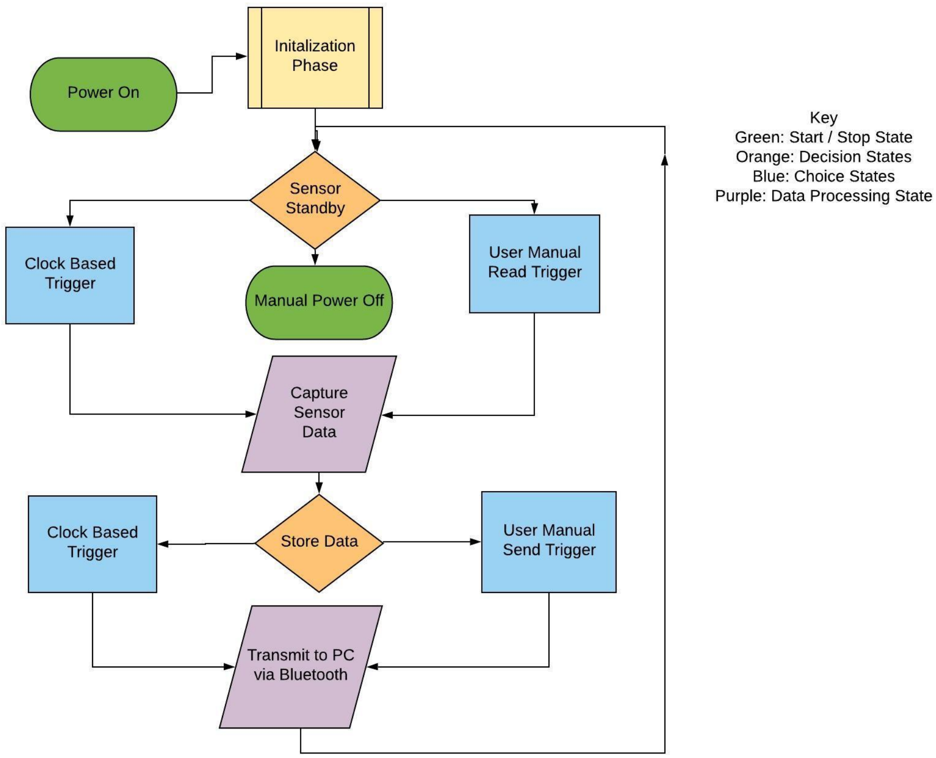


Figure 3: CpE Design System Flow

2.5.2 Electrical Engineering Design Flowchart

The electrical engineering design flowchart depicted in figure 4 shows the over view of the electrical part of the system. The AC power went through the relay switch which is a solid-state relay, which has the ability to switch off AC loads at the point of zero load current which protected from any power surges or any type of arcing, or electrical noise making the system safer and more efficient. From the solid-state relay the microcontroller was then be powered according to its power rating of 3 to 3.6 volts. The replaceable battery was not be connected directly to either the AC power or the relay switch to increase the protection from surges when P-Quad is being transported. The micro controller has received data from moisture sensor, humidity sensor, temperature sensor and the light sensor, and according to this data from the sensor the micro controller then relays instruction to the LED light, water pump, and other hardware to control the environment inside the pod to nourish the specimen; furthermore, all the information that is needed to alter the pod is sent from the micro controller to the LCD display for the consumer. This helped the consumer read and note the status of the specimen inside the pod without the need to open P-Quad. When the user needs to change the setting or see stored data, they need the access to buttons that then relay data to the micro controller which in turn send the data to LCD as described above. To dive in a little more detail, the alternating current provided the main power however, when the main power is not plugged in then the battery takes over and for that to happen we implemented a switch system that seamlessly alternate between AC power from the wall wart to the battery power. This was accomplished by an automatic switch. We had been leaning towards manual switch to give the user more control and decide when they want the battery to take over and provide the power. This is due to the complex nature of the transportation of plants.

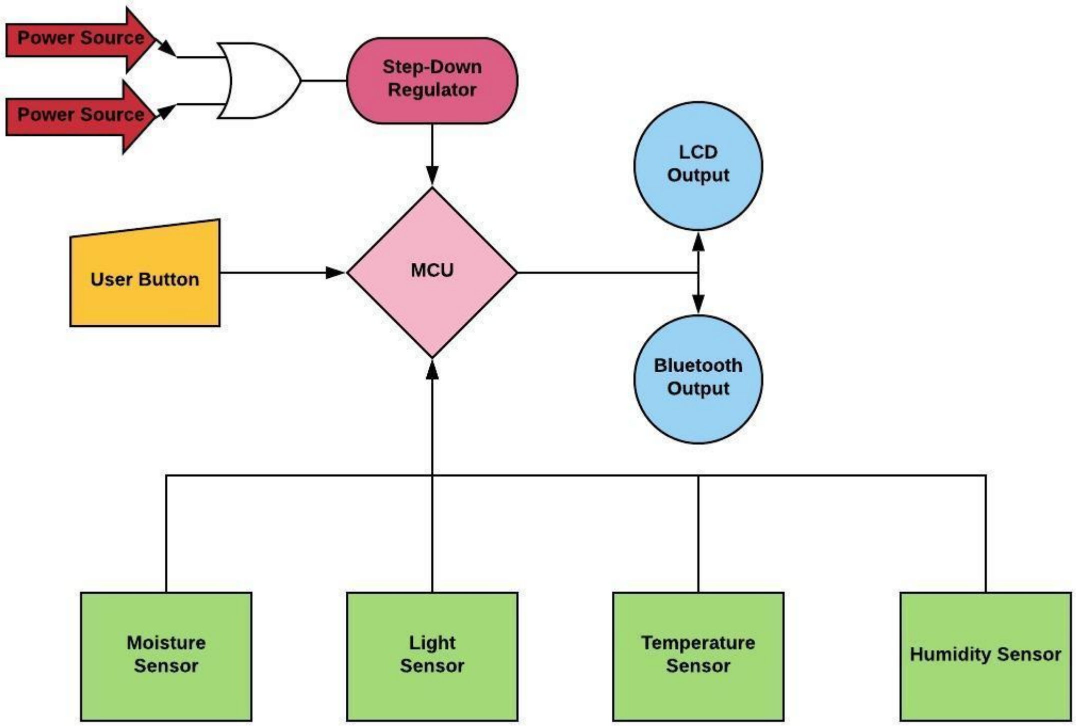


Figure 4: EE Design System Flow

2.6 Decision Matrix

P-Quad is a great peek into an alternative engineering application of transporting not only plant but other plant like specimen that require transport with temperature control, moisture control, and that has certain other needs that require power consumption. The P-Quad has presented many opportunities for users to develop interests in unique and rare plants without the fear of safely transporting them from one place to the other. The following image in figure 5 is the decision matrix which has a list of values that provides us to systematically identify, analyze, and rate the performance of relationship between some important criteria and the choices the consumers have. This matrix is very useful for large decision factors and assessing each factors significance for example the Survivability of a specimen which is weighted the highest value of “5” and then its value is compared to all the options that are available to the consumers. This particular decision matrix represents a best option and to choose from those options on a rating system. This shows the users their options by comparing all the things that the customer demands and how each of those demands are being fulfilled and how well those demands were met.

Weighted Decision Matrix - Which Way is Best?

Date last saved: Oct-14 2018

Decision Factors		P-Quad	Mutiple Specimen	S-Term Protection
Criteria	Wt.	1	2	3
Plant Survivability	5.0	5	2	1
Power Usage	4.0	-3	2	2
Efficency	3.0	5	1	0
Data Storage	2.0	5	1	0
Safety	2.0	5	2	1
Cost	2.0	-4	-2	5
Support	2.0	5	1	-2
Weighted Scores		50.0	25.0	21.0

Which transport option is Best?
Transporting valuable specimens have some very demanding requirements. Some of these like Survivability, power usage, efficincy, safety, support and data storage- can not be short changed and no corners can be cut. To have the most realiable option this matrix shows the weights of all the criteria and compares it to all the options and gives the customer the best options for them.
▶ Winner: P-Quad 50.0

Criteria	Definition
Plant Survivability	How long a specimen will be safe from the elements when being transported
Power Usage	How much power is needed or required to provide the specimen with nutriants and light
Efficency	How well the specimen is cared for and how is that information is relayed to consumer
Data Storage	Amount of data-history and information on different plant types-can be stored
Safety	Not limited to non-toxic materials, lose of data, safe transport
Cost	Up-front costs and ongoing costs (e.g. payment to developer)
Support	How good is the support community at answering tricky questions about using the device, is the P-Quadupgraded regularly to keep up with changes

Figure 5: Decision Matrix

2.7 Project Measurables

The goal of this senior design is to develop a sophisticated, yet easy to use system that makes the transportation and Maintenance of any plant or flower virtually effortless. With Changeable environment with a press of a bottom, readily available sensor data may it be temperature or moister level, and an easy to learn setup process, P-Quad promises following project measurable specifications to the consumers.

A. Battery Life
   a. Transportability is the most crucial aspect of P-Quad and to achieve the highest distances that can be traversed before reaching an AC source, a rechargeable battery source was implemented for the longest possible survivability of the specimen. To show the battery life for this project at a smaller scale we used a 9-volt lithium battery, which demonstrated that all the features that are required to monitor and maintain the P-Quad remains functional at the very minimum of three hours.
   b. Replacing the rechargeable battery is also something that can be achieved by the consumer without the assistance of a trained professional thus reducing the Maintenance cost for the customer. However, P-Quad has several dedicated technical service representatives that assist any customers that call for help with processes like change of battery. In order to show the change of battery we demonstrated to the judging committee in person

during our presentation how to change the battery to simulate a real-world situation.

B. Memory
   a. Storage of data and information are extremely important parts of P-Quad or any system that has supervisory control and data acquisition used for reliability and control of a system. The internal memory holds information on various plants and or flowers that the consumer desires to transport. With a press of a few buttons the user can set the P-Quad's environment to the desired plant.
   b. Furthermore, there is a large library of different environments that can be used in a sort of custom setting if the specimen is very rare, so the user should be able to choose a certain setting for moisture control, and temperature control, and how much lumens of light to provide to the plant. We believe that the right number for internal memory space is no less then 500 MB, this provided ample space for data and information storage, may it be default plant settings for custom settings that are implemented and saved in P-Quad by the consumers themselves.
C. Power Usage
   a. Having an LCD to display information, different number of lumens of lights, temperature control, moisture control P-Quad consumed power at a rate that was challenging to balance with distance traversed and battery life. To remedy this problem, we used LED lights for the plants, and have power saver settings for the display to reduce the power consumption as much as possible. We have the power usage to be around 600 Watts
D. Temperature Control
   a. P-Quad has a feature implemented to monitor and control the temperature for the plant or a rare specimen that is placed inside during transportation and when P-Quad is idle. All most all the plant has a requirement that is unique to their type when it comes to temperature demand. Therefore, to tackle this P-Quad has various function sensor that not only monitors the temperature and display it on the screen for the user but also it provided the plant with the correct temperature. We had the sensor set up so it can monitor and adjust the temperature from 20 to 40-degree C, this covers all the plants.
   b. This leaves out how the plant that are rare and have unique requirements. For that like described before, the user has the option to input custom setting hence, having the proper environment inside P-Quad.
E. Light Control
   a. Providing adequate light is the next essential part of keeping a plant healthy during transportation. We used sensor for this as well. The ambient light sensor relates the amount of light that is present in side P-Quad and LED lights are mounted inside the enclosure to light up or dim down accordingly. The target for the light is estimated to be over 500 lumens.
   b. Different color lights (add later)

F. Humidity Control
   a. The last measurable project engineering specification is the monitor and control of humidity in P-Quad. For that we are using board mount humidity sensor which relays the amount of moisture that is present for the specimen. The measurement bounds that we implemented are one to two oz of water. That is injected in P-Quad with the use of a pump that has water in a reservoir.
   b. Many plants require different amount of water for their nourishments and health if a plant receives to much of water it can burn under the light or if it receives to little water then it is becoming dry and might die. P-Quad guarantees the safety of a specimen when being transported or when idle, and to provide the customer with the best product we included the feature of controlling the water dispensing in the storage, so the customer can input custom value of the amount of water to be dispensed.

G. Water Pump
   a. In P-Quad a feature that is needed to keep the health of the plant at a good level, we implemented a water pump that included rubber pipes pump mechanism and a small spray at the end of the pipe that is attached to the water pump. Initially our group was thinking about having the pump on a time delay so that it can pump in water that was set so a certain predetermined amount of water dispensed. However, we did look at another option which was to include the information in the software and have the user pick how much of the water that was dispensed. Both of these ways are great, and we had to implement some degree of both.
   b. The pump works on a timer that is controlled by which setting the user as chosen. From there after that set time is up the pump dispenses the water inside the pod to water the plant. However, to properly dispense the water on the plant and soil a small spray is installed on the end of the pipe so ensure the water in sprayed uniformly. For the proper moisture control we have moisture sensor that detected the amount of water in the soil, however, with the help of humidity sensor we are able to gauge the humidity outside of the soil hence, proving water directly to the plant not just the soil.

3 Research and Investigation

With this chapter we aim to dive into the various and potential technologies with close or approximate relation with the P-Quad. This section dives into existing product or products and projects that are highly contemporary to our proposed project. In addition to this we wanted to gain a thorough understanding of the core technologies that are integral to the project's final vision so that we could be more educated in the options and selections to be made. Next, we look at each specialized part and break down the various functions of it before selection a list of parts comparing them and then choosing one to use for the P-Quad.

3.1 Existing Similar Projects and Products in Market

In terms of existing marketed products there aren't many that match this one on the same conceptual level. Certain hydroponic systems have similarities such as controlled lighting systems. Additionally, at the home market level there exists automatic self-watering systems for houseplants. However, these systems are fed into an already existing plant enclosure and offer no control or feedback of the environment surrounding the plant as a whole. A highly similar project recently just went to its end of life cycle from the vendor Parrot Electronics was their Parrot Pot smart plant pot system. The parrot pot was designed less with the intent for travel and more with the intent for an automatic growth environment. That difference aside however there were a lot of similarities in concept between our P-Quad and the Parrot Pot. The most noticeable similarity was the fact that both utilized embedded sensors in order to monitor the plants health levels. Another feature both shared is battery power the Parrot Pot however only runs off of battery power compared to the P-Quad system which would offer interchangeability between battery powered and AC outlet power. There also exists differences between the two beyond the difference in power methods, first for instance the Parrot Pot has no direct way of controlling the pot on the device and instead utilizes a phone application in conjunction with a dual LED feedback system on the plant to indicate status. Whereas on the P-Quad we plan to have the controller for the system directly integrated into the container with the control inputs being underneath or attached to the display unit which showed the plant's current environmental readings. The final notable difference is that while the Parrot Pot has the ability to monitor luminosity levels, the pot itself cannot perform any sort of correction or adjustment to them and merely provides a feedback to the user to move the pot to a location with more light. (15)

Aside from the market presence of the Parrot Pot, there is also a patent dealing with the preservation of plant life for the shipping of floral arrangements numbered US8250805B2 (hence force referred to as B2). While this one shared some similar concepts to our P-Quad its main differentiator is the fact that this proposed system exclusively deals with plants who have been severed from their roots meaning that while our system aims to deal with a plants growth cycle and help keep it alive and thriving, this B2's system deals with the plan in state of dying and trying to preserve that plants life in that state for as long as possible. Despite this core difference there are similarities. First B2 proposes enclosing the plants with an antimicrobial preservation media in order to prevent the plants from catching any viruses or diseases which is directly comparable to P-Quad's filtration system and model of isolation for multiple plants. Additionally, the B2 proposed a protective box enclosure which would serve the two-fold purpose of both shielding the plants from harm as well as simplifying the process of shipping the plants over long distances. Similar to the P-Quad's water reservoir and pump system the B2 has miniature water vases that encapsulated around the stem of the plant so that it may drink water in order prolong its life, however unlike the P-Quad this system leverages no mechanical parts or soil and relies on the stem of the plant to do the entirety of the work. Lastly B2 proposes the incorporation of temperature and humidity control for the plants but makes no documentation on how to accomplish this making it another similarity in concept. (16)

3.2 Relevant Technologies

The most pertinent relevant technologies to the P-Quad are its embedded sensor systems which are comprised of light, temperature, humidity, and soil moisture as the main sensors. The temperature and humidity sensors are typically bundled into one combination sensor in today’s market, in both cases these are capacitive meaning they use a metal oxide which stores or dissipates electrical charge based on the surrounding humidity value. For the light sensor we chose to use an ambient light sensor which in short measures the amount of ambient light around it. Additionally, we want to look into pH level sensors in order to try and monitor the amount of nutrients provided to the plant over a given timeslot. More technologies that are included are a light fixture to act as an artificial sunlight supply for the plant, so it can carry out the process of photosynthesis successfully. Next there is a water tank and pump system in order to provide the enclosed plant with a steady supply of water through a small pipe or jet into the soil. The pump mostly utilizes a small motor in order to drive the water throughout the pipe system. Filtration is another present technology in our P-Quad system, the goal of the filters is to assure that the plants only receive fresh clean air cleared out from any potential diseases, bacteria, or fungal infestations. (12)

3.3 Microcontroller Introduction

Perhaps the most crucial component for the entirety of the project is the microcontroller (hence force MCU) as it acts as the brains of our project. The most important thing the MCU does as it acts as the driver for our sensors as well as the way we collect data from them so that we may build a picture of the systems environmental surroundings. Next the MCU is in tasked with the containing the subroutines and/or interrupt service routines responsible for maintaining ideal environmental conditions for the pod and the checks to see if any of these conditions are not met.

3.3.1 Microcontroller Selections

With the importance of the microcontroller in mind, we decided it pertinent to setup guidelines to what is most important for us in terms of achieving functionality. The first and foremost is the amount of GPIO (general purpose input and output) pins the MCU is able to access since having multiple sensors and systems being driven by the MCU proved difficult if we cannot directly interface them with the MCU itself. Operational voltage is another important factor due to the fact that most of these embedded sensors are designed for low power environments (sub 5V) we wanted to make sure that they would be able to link directly to the MCU without having to worry about overloading the sensors this also ties in to power consumption. Memory, both program storage and RAM, play a large role in ensuring the projects functionality. First if our compiled code cannot fit on the program memory space then we would have to reduce functionality arbitrarily and if there isn’t enough RAM the program could experience delays in runtime functionality or loss in data integrity both of which are deemed unacceptable. Finally, there is the matter of the CPU which while we prefer it to be fast enough to keep the processes executing in a timely manner a bigger concern is the level of programmability of the clock and the various sleep modes it supports after all since this is a low power portable system a large portion of the time won't be spent executing the control systems, but ideally keep it in a less power consuming standby state for monitoring.

3.3.1.1 Texas Instruments MSP430

One of the microcontrollers that we considered for P-Quad was the MSP430 from Texas Instruments and the reason we really liked this MCU is because it is an ultra-low-power microcontroller that has features numerous peripherals targeted for various applications. With the help of five low-power modes in portable measurement applications the battery life can be extended which looked to provide P-Quad longer battery life when being transported. The main reason we decided not to use MSP430 is due to its BSP being a UART. Bootstrap loader is basically a built-in application of MSP microcontrollers, and it allows its user to read and write to its memory. Now UART which stands for Universal Asynchronous Receiver-Transmitter, is a computer hardware feature that helps transmission of data while also giving the user the ability to customize the data format; furthermore, Universal Asynchronous Receiver-Transmitter allows the user to configure the speed of the data transmission. Asynchronous communication allows us to send message only after the preceding message is finished and this can cause a latency issues when interfacing and therefore, we did not use this microcontroller. This MCU provided up to 500MHz of clock speed which was deemed far more than would be needed to keep the program execution flow running at expected rate, however the faster clock speed could lead to issues with power consumption even after the mutability of the clock the speed would still be higher than needed in most cases.

3.3.1.2 SMTicroelectronics STM32L451CCU6TR

Another microcontroller we had in consideration was the STM32L45 which is an ultra-low power ARM 4 CPU. One of the main reasons for consideration for this MCU was its variety of communication channels which included one CAN, four I2C, one Low Power UART, one SAI, three SPI, and one regular UART which makes for a total of 16 communication lines. The data bus width in this model was 32-bit lines and in all cases our sensors only transmit at 16-bit width meaning that 32-bit would be unutilized fully. The MCU contained 11 total timers allowing us for more precise sampling control. The chip contained 83 fast I/O's the majority of which were rated for 5V tolerance. One unique feature noted about this family was the nest vectored interrupt controller (NVIC) which allowed support for 16 separate priority levels with 16 separate interrupt lines and allowed up to 67 maskable interrupt channels. This allowed for low latency interrupt processing due to the NVIC tightly coupled design paradigm.

3.3.1.3 Microchip Technology SAMA5D2

The SAMA5D2 chip is another ultra-low-power ARM cortex A5 processor based MPU. It runs up to 500MHz and has ARM NEON SIMD engine. Now this chip provides many security functions to protect the consumer's code. Moreover, the external data transfers are secure. Some additional features of this MPU are low system cost and high integration with 4-layer PCB with less than 200uA retention mode with fast wakeup. A downside of this particular MCU was that it included 2 communication lines for I2C which would prove a problem considering we have three sensors that utilize I2C. Some plus sides to this particular model was it came with support for LCDs, keyboards, and touchscreens built in. In addition to this it came with 7 different types of RAM controllers as well as 10/100Mpbs ethernet support. The bus width is 32-bit wide which again presents some issues of underutilization of the data buses again since our sensors are only 16-bit.

3.3.1.4 Texas Instruments F280049PMSR

Another Texas Instrument controller we looked into was from the TMS320 family aka the Piccolo series. This system in particular features real time control making it perfect for close loop control application. However, like many previous MCUs it once again lacked in I2C lines having only one it does however have two of each of the following CAN, SPI, and SCI as well as one LIN (Local Interconnect Network), one power management bus, and FSI (Fast Serial Interface) transmitter and receiver. This series did have some interesting features however. One such feature is its several on chip math units such as the Trigonometric Math Unit which contains predefined common trigonometric operations and functions optimized for runtime and the Virterbi/Complex Math Unit which allows for calculations involving complex numbers. The board also features a wide array of self-diagnostic tools including a missing clock detection circuit to inform the user it the external clock has been disconnected or there is an issue with the on-board crystal. A brownout reset circuit to help the board recovered from a brownout, which is an extended drop in voltage supplied extended being defined as minutes or hours. Additionally, this is all managed by an embedded real-time analysis sensor in order to rectify problems as soon as they occur.

3.3.1.5 Microchip PIC24FJ1024GA606-I/PT

The final MCU we examined for our project was from the PIC24FJ1024 family. One of the most noticeable features about this particular chip series was the hardware real-time clock/calendar with time stamping. This built in feature would allow us to date operations down to the exact hour/minute/second format in addition to providing the month and day. This could prove highly beneficial for plant life with highly sensitive and restrictive growth cycles and ultimately allows us more control over the preservation features present in the P-Quad. Additionally, the sleep and low power modes are modifiable to selectively shut down peripherals meaning the process of saving power would be made easier since we could selectively shut off the power intensive components such as the pump or the light during standby periods and leave the embedded environmental sensors running in order to monitor the condition of the P-Quad thus allowing for ease of switching back on of the control systems. Additionally, this family model supports 3 I2C lines, 3 SPI lines, and 6 UART lines which would provide us with plenty of communication lines with for our sensors and other various electronics we may come across. This chip is also 16-bit architecture meaning that there would be no issues present with the current sensor selection. Some downsides were that it only has 5 16-bit times (which however can be combined into 32-bit timers) and only allows up to 5 external interrupt sources which may prove restrictive depending on what features we choose to implement. Finally, the chip lacks native ethernet or wi-fi support which may have to be supplemented with a module.

3.3.2 Microcontroller Comparisons

Below in table 4 we have a comparison of our various microcontrollers and the metrics we used to measure and compare them which lead to our decision. In addition to the names of the controllers we also including the manufacturer to show the variety of options considered.

Table 4: Microcontroller Measurements of comparisons

MCU	F280049PMSR	ATSAMA5D21C-CUR	STM32L451CCU6TR	PIC24FJ1024GA606-I/PT	MSP430FR2311IPW16R
Manufacturer	Texas Instruments	Microchip Technology	STMicroelectronics	Microchip Technology	Texas Instruments
Program Memory (kB)	256	160	256	1024	3.75
Data Memory (kB)	100	128	160	32	8
Clock Frequency (MHz)	100	500	80	32	16
GPIO	26	128	38	53	11
Operating Voltage Range	1.2~1.3V	1.2V	1.71V~3.6V	2~3.6V	1.8V~3.6V
Size L x W (cm)	33.6 x 33.6	1.4 x 1.4	1.58 x 1.58	.07 x .07	.05 x .04
Price (USD)	$10.81	$6.71	$6.01	$4.41	$1.62

3.3.3 Microcontroller Selection

Ultimately our group decided to choose the PIC24FJ1024GA606 by Microchip Technology The decision behind this was several folds. First and foremost, one of the biggest selling points of the chip to us was the real-time clock and calendar with timestamping in a project that aims to preserve and monitor a plant’s life over extended periods of time in order to prolong its life to maximum. This technology would allow us to implement strict timing controls on things such as watering cycles or allow us to simulate the time cycle of the plant's natural exposure to light. Another benefit was that this MCU possessed 3 I2C lines which is the minimum number of lines we need in order to properly communicate with the sensors. In addition to several versatile low power modes the MCU allows control of shutting off external peripherals such as the pump or the light which further aid the P-Quad in its design of being a low-power system. Additionally, it has a large amount of program memory meaning that we would be able to store code that possess a larger amount of more complex subroutines or program a multitude of different states in order to cover as wide array of plant life as possible in the P-Quad. Since the chosen MCU was a 16-bit MCU there would be no issue of writing backwards compatible code since all the chosen sensors were 16-bit as well. While it took some compromises in the data memory, clock speed, and timer amount categories these tradeoffs were considered to be within a workable amount for the project scope. The MCU contained 53 GPIO a good majority of which are programmable to different pin functionalities via PPS (Programmable Pin Selection) which is another factor to why we chose this particular MCU.

3.3.4 Microcontroller Selection Addendum

Once the PCB board design process had begun we encountered a particular problem with this and other selections that had not been accounted for and that was that the PCB layout software we decided to use EagleCAD did not have the part listed in its database and even after downloading additional libraries provided by the manufacturer (Microchip) the selected MCU was still not present after some consultation with our Senior Design professor Dr. Wei, we were advised that we should not attempt to create the part ourselves nor should we pay for a professional to make the part footprint and schematic and instead were advised to reselect the MCU with one that has a known footprint and schematic in the CAD program. While this initially proved concerning after some digging into alternative MCUs we found a highly viable alternative from the same Manufacturer MicroChip which was a slightly older model of a chip from the same family and was also present in EagleCAD. At first there was concern that this chip would prove to have stark differences compared to the old chip but diving into its specs we quickly put this to rest. First the data bus width was still 16-bit lines meaning that communication with the sensors would not be an issue and with regards to sensor communication this particular model had slightly more GPIOs containing 84 pins to the previous 53. All the communication modules of I2C, SPI, and UART were still present in the same amounts as the pervious chip furthering the fact that this chip would be able to handle the communication side of the project. The clock on this model was able to reach up to 32MHz double the speed of the old clock however this could be divided down if this proved too fast and power intensive for our purposes. Its program memory is smaller than the previous choice coming in at 128kB of storage as well as the RAM dropping from 32kb to 8kB however due to the compact nature of the embedded processes going into the chip's program both of these were considered acceptable losses. Finally, being from the same family as our previous selection the new MCU is still able to leverage all the features of the PIC24F family such as programmable pin selection and disabling of external peripherals. Ultimately despite some minor tradeoffs this chip was chosen because it mostly closely matched the original chip in terms of both the decision metric's performance, technologies present in the chip, and price range.

3.3.5 Wireless Communication for MCU

One thing we had to take into consideration was the fact that our MCU chip did not have a wi-fi or Bluetooth. This raised a concern for our group as the P-Quad would potentially need to communicate with external systems which under the current design implementation would prove impossible. In order to remedy this situation, we needed to look into a communication module to fit on the PCB so that the data could be communicated to external sources. Due to the nature of communications we had to decide which wireless communication medium we would be transmitting over from the common selections. The mediums we examined are NFC (near field communication), RFID (radio frequency identification), Bluetooth, and Wi-Fi.

NFC: This communication protocol is an extremely short range (no greater than 4cm or 1.6 in) protocol which enables two-way communication between electronic devices. This protocol method is often used between two small electronic devices to setup extremely short-range communication system typically used for electronic identity documents, contactless payment systems, or keycards as opposed to a full blow communication system in itself. The reason for this is several fold starting with the fact that NFC despite being a highly simple setup its speed rating is very low speed compared to most other protocols offering an operational frequency of around 13.56 MHz and a selection of data transfer rates of 106Kpbs, 212Kpbs, or 424Kpbs. In addition, NFC offers no internal cryptographic protection and despite its short range of broadcasting it is susceptible to man-in-the-middle attacks and this vulnerability can be exacerbated by the use of antennas used to magnify the strength of the radio signals making these attacks even more trivial to perform. Despite these flaws NFC often is useful for establishing a bootstrap to a more powerful and secure connection protocol creating a multi-authentication network. (55)

RFID: This communication protocol uses electromagnetic field to identify and track tags, which contain electronically stored information, attached to an object. Consisting of 3 tag types active, semi-passive, and passive. Each of these tag types is able to store up to 2KB and despite the three classes of tag most function the same as all store the data in the RFID tag waits to be read until a RFID reader’s antenna sends an electromagnetic signal to the tag’s antenna then the tag sends radio waves back to the reader either powered by an internal battery or uses the power generate by the reader’s electromagnetic field finishing with the reader picking the tag’s radio waves and interpreting them as data. The main differences between the tag types are fairly simple passive tags rely entirely on the RFID reader for a power source and can be read up to 20ft, whereas active and semi-passive uses internal batteries to supply power to the circuit with active tags using the battery to broadcast radio waves to the reader while semi-passive tags rely on the RFID reader to provide the power for broadcasting similar to passive in this regard also these two types broadcast around 850-950MHz and can reach up to 100ft or in some extreme cases with extra batteries 300ft away. Regarding storage type RFID storage has three categories read-only, read-write, and WORM (write once read many) the first two working as their name indicates and the WORM tags being able to write additional data once in addition to their default setting but after this one time use the data cannot be overwritten. Despite these many options for RFID it ultimately doesn’t work as needed for the project as it is stuck with one-way communication meaning sending anything back to the system as acknowledgement or control would prove impossible with it. (55) (56) (57)

Bluetooth: The next wireless communication technology is Bluetooth which aims to be a two-way protocol able to exchange data over short distances (around 330ft) utilizing ultra-short wavelength UHF radio waves in the 2.4-2.485GHz range. Bluetooth uses a packet-based protocol alongside a master/slave architecture meaning that one master may communicate with up to seven slaves with all the devices sharing the master's clock and basing packet exchange around the master clock, additionally a slave may belong to more than one master and communicate with both of them. Bluetooth devices are classified into 4 distinct classes starting with 1 as the highest and ending with 4 at the lowest the dividing factors for the classes being based around both range and max power. Class 1 is the strongest of the devices drawing 100mW and broadcasting approximately up to 330ft. Class 2 drops significantly from class 1 only being able to broadcast up to around 33ft but taking only 2.5mW of power. Class 3 and 4 are both the low powered implementations with class 3 being able to broadcast up to 3ft while only taking 1mW and 4 taking only .5mW but reaching distances of only 1.5ft. All of these versions possess a fairly fast data transfer rate of 22Mpbs, and Bluetooth devices are a two-way device structure making them a possible candidate for the P-Quad. However, it should be noted that the effectiveness range can be decreased by certain factors such as material coverage, propagation conditions, battery power levels, and antenna configurations. (55)

Wi-Fi: The most commonly used wireless communication technology today it was created as a radio wireless LAN (local area network) allowing devices on said LAN to communicate large amount of data with each other tow-ways in a trivial manner. Access to wi-fi network is permitted to any device in the range of the signal that possess a wireless network interface controller (NIC) which does lead to a concern as Wi-Fi is more susceptible to eavesdropping attacks than its wire counterpart due to this. Different standards of Wi-Fi exist however the two most common are the 2.4GHz and 5.8GHz bands. Wi-Fi is one of the stronger broadcast signals with the max cover range being around the same as Bluetooth class one at 330ft. However, unlike Bluetooth Wi-fi is capable of delivering data rate speeds of 144Mpbs making it infinitely more suited for larger scale wireless communication. Wi-Fi transmission is somewhat susceptible to interference which can be caused by a number of devices in the same broadcast region beyond the load bearing capability of the Wi-Fi network infrastructure which can lead to slowed data transfer speeds or sometimes loss of performance. Wi-Fi ranges and speed are directly correlated with the antenna type and gain as well as the radio power output and receiver sensitivity making it a highly varying but highly configurable communication protocol. One benefit to Wi-Fi over the other methods mentioned here is large breadth of support in both technical documentation and support in libraries and hardware. Despite the flaws these beneficial factors together make it the best candidate for the P-Quad's wireless communication system. (55)

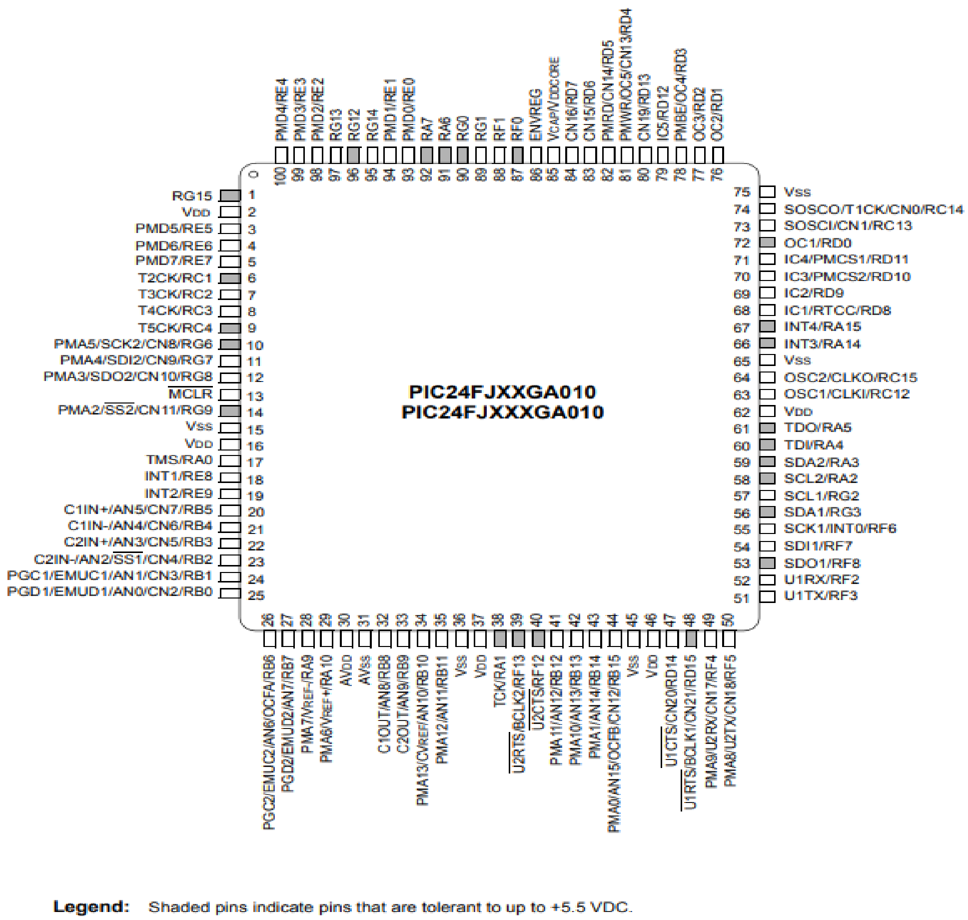


Figure 6: MCU Pin Diagram

Above in figure 6 the pin layout for the MCU we have chosen and below we have a mapping of each pin to its various functionality along with if the pins are programmable. The other benefit of using the Simulator provided by Microchip is the ability to start writing code for our project without having any of the hardware physically. We can write, compile, and debug our application without ever having the microcontroller on hand and programmed.

3.3.6 Compilers

The MPLAB X IDE does not have an incorporated compiler upon download and install to the host machine. The user must download a Microchip microcontroller specific compiler based on the processor used. There are options dependent on the processor size, the 8-bit processors require the MPLAB XC8 compiler, the 16-bit processors require MPLAB XC16, and the 32-bit processors require the MPLAB XC32 compiler. Upon the creation of the project the user selects the specific compiler specific to the processor selected.

The PIC microcontroller that we have elected to use is the PIC24FJ1024GA010. PIC24s fall within the 16-bit processor family, therefore we used the XC16 compiler for our design project.

3.3.7 MPLAB Code Configurator

Another interesting feature of MPLAB X is the MPLAB Code Configurator (MCC). The MCC is a tool that allows a graphical interface with the user to initialize modules and settings. The code configurator then generates the necessary code to initialize the desired modules based on the settings applied in the graphical interface. Using the MCC could prove beneficial to us. Using it would allow us to focus on the application specific code and not on the setting up of different modules.

3.3.8 Application Flashing Process

The process to go from source code to running the application on the flashed chip requires three main components: a computer running the MPLAB X Integrated Development Environment, a Microchip proprietary programmer/debugger, and a target board with the PIC microcontroller.

There are a few options in selecting the programmer/debugger which are quickly summarized in table 5. They all support a common base feature set, but they don’t all support every debug feature available. Our options and their features are shown in the below table. The PICkit 3 is the lowest cost member of the debugger/programmer family and supports the most fundamental debugging features that developers are most likely to use. The ICD3 adds intermediary support of more advanced debugging options. The most expensive and most powerful debugging option is the Real ICE (In-circuit debugger). It offers all the possible features and tools for debugging as well as a sophisticated trace and log functions. The best feature of the Real ICE is the ability to view variable values in real time or very near real time.

Table 5: MPLAB X Features

Feature	dsPIC30	PIC24F	PIC24H dsPIC33	PIC32MX	PICkit 3	ICD 3	Real ICE
Trace Data & Program Flow							
Runtime Watch							
SW Breakpoints							
Stopwatch							
WDT Overflow							
HW Breakpoints	1-4	1-6	1-6	1-6			
Peripheral Freeze on Halt							

3.4 Environment Measurement Metrics

For this project to be successful we need to first understand the various environmental factors of keeping a plant alive and healthy and the proper way to measure them. Without this the sensors might be used improperly or record the wrong readings vital to the stability of the plant's environment.

3.5 Sensors

Sensors are a device or a module or a subsystem whose purpose is to detect events or changes in the environment in which it is placed and then send these changes, information in form of data, to other electronics like LCD display for user or to the microcontroller. A sensor is almost always used in collaboration with other electronics with different degree of complexities to a regular light fixture to a computer with many different functions.

3.5.1 Soil Moisture

The first and most unknown to us is soil moisture which simply stated is the amount of water level contained in the soil the plant is in but let us dive deeper into what that means and how we can utilize it. First there are two ways one can measure soil moisture via moisture tension or via moisture content. (6)

The first method tension tells you how easily water can be extracted from the soil. With proper saturation the tension is low and plenty of water coats the soil particles as well as occupies the surrounding pore space, the lower the tension the less effort the plant must apply to get the water into its roots. However, there is a threshold depending on the strength of the roots and the type of soil in which the plant can no longer be able to trivially extract water from the soil (even if water is present in the soil) and have begun to stress its roots to get the water it needs. This state is referred to as high tension and if not rectified can lead the plant to wilt and eventually die. The unit tension is measured in is kPa (kilopascal) and it is the number of Newtons over a square meter. (5) (6) (7)

The other method to measure moisture levels is by checking the content. This is a traditionally volumetric approach in which you measure how much water is in the soil as compositional percentage. The unit of measurement for this method is referred to as VWC (volumetric water content) and signifies a ratio of inches of water to inches of soil. It should be noted that despite the assumption that 0% would be the threshold for danger depending on how the type of soil compacts there is a certain VWC that was present but inaccessible to the plant, therefore the VWC at which the plant begins to become stressed is not at 0% but slightly above that. (7) (8)

For the actual sensors themselves there are three types volumetric meters, tensiometers, and solid-state sensors. Volumetric meters tend to be the most expensive class of meters and require calibration to the specific environment they inhabit; however, they offer the highest accuracy and instantaneous data. Tensiometers are typically less expensive than their volumetric counterparts while still offering good accuracy on data. The drawback is these sensors often have regular maintenance of needing to clean out and refill the water in the sealed tubes which provide the baseline for all the measurements. Additionally, because they utilize water in the sensor itself, they may perform inadequately in colder temperature environments. The last and cheapest class of sensors solid-states measures the electrical resistance in the soil, the lower the resistance value the more water is present in the soil and vice versa. The main drawback of this type is they require specific installation, periodic maintenance of replacing the gypsum plating to combat dissipation to keep the readings accurate, and slower throughput on reading environmental conditions. Additionally, these sensors often do not perform well in soils with high salinity concentration. However, they require no calibration and are undoubtedly the cheapest option. (7) (9) (17) These comparative metrics are displayed side by side along with their potentially components in table 6.

Table 6: Moisture Sensor Comparisons

Sensor	VH400	SMS-BTA	WaterScout SM100	SM150T	SEN-13322
Manufacturer	Vegetronix	Vernier	Spectrum Tech	Delta-T	SparkFun
Voltage-Supply	3.5V-20VDC <13mA	5VDC 3mA	3V-5V 6-10mA	5V-14V ~18mA	3.3-5V
Operational Temperature	-40°C-85°C	-40°C- 60°C	0.5°C-80°C	0°C-40°C	0°C-35°C
Dimensions (Length)	9.376cm	8.9cm	6cm	5.1cm	5cm
Standard Accuracy	±2%	±4%	±3%	±3%	±5%
Unit Price (USD)	$40	$99	$89	$250	$6

Note Standard Accuracy is defined as accuracy measured from 20°C-25°

3.5.2 Soil Moisture Part Selection

For our selection of a soil moisture sensor we went with the SparkFun SEN-13322 with the reason for this choice was several reasons. First and most notably was its much cheaper price relative to the other choices on the list with its mere $6 price tag. This initially gave us concerns into functionality and performance of the sensor. However, after consulting with our senior design professor Dr. Wei the discussion led us to the conclusion that most of the sensors would assuredly perform around the same and offer the same level of functionality despite the price tag which was a big step in assuring our selection. The next major reason was the way the sensor is controlled and outputs. All the other sensors had their own methods for communication and providing power to the system that were typically designed for prefabricated devices made for data logging. These could prove troublesome as most of the methods of communication to the data logger were not explained in the datasheets of the products meaning that intergrading them into the PCB and the P-Quad’s MCU could prove extremely difficult. Whereas the SEN-13322 had a very simple communication process of ground, VCC, and signal in which we could tie signal to a GPIO pin in order to take in readings making it the most straightforward sensor in terms of communication. Another large issue we had to consider is that aside from the SEN-13322 none of the other soil moisture sensors had any listings in EagleCAD or any downloadable libraries for it that our group was able to find. This meant that we would have to create both a footprint and a schematic for an already semi-obscured design which would prove disastrous for the PCB fabrication as any mistake in interpretation from the datasheet information could invalidate the entire connection and render the whole board worthless for connection of that sensor and could possibly lead to other issues with the board. In conclusion the hurdles presented by the other soil moisture sensors proved too daunting to overcome in addition to an already large price tag while SparkFun's SEN-13322 proved to be the most accessible in terms of both price and operational concerns. (22) (23)

3.5.3 Soil Type

Another thing we must consider in addition to the moisture levels of the soil is the type of soil in which the plant is kept in. This is important to remember because regardless of the various sensors and control factors present in the P-Quad, if the plant cannot properly access nutrients and water from the soil housing its roots. The type of soil is determined by the present levels of three types of weathered rock particles (sand, clay, and silt) in its contents. The first common type is sandy soil, which often has trouble retaining water, but it is very lightweight and warms up much more rapidly than the other types. Next is clay soil, this type has the most compact particle type making it the most effective at retaining moisture content. This however comes at a cost of highly restricted airflow and increased difficulty of plants with non-hardy roots to penetrate deeply enough to absorb the nutrient rich soil. Following is silt soil which while inferior to clay in terms of water and nutrient retention it is superior to sand-based soils in that it has adequate moisture retention, again one must keep in mind that this means the soil drains very poorly. Saline soil is used typically in the cultivation of plants from brackish regions however for the average plant the high salt content typically leads to plant growth being stalled or in worse cases damaged as it can easily lead to wilting of leaves. Moving on the next type of soil to consider is peat soil which is a swampy soil rich in nutrients due to volumes of ancient compressed plants providing rich pockets of nutrients. Peat has a unique property in that its composition holds water very well in dry conditions but shields plants from drowning in overly wet periods. The tradeoff for this is that peat soils moisture contents are often much more acidic than its counterparts. Finally, the last type of soil we consider is loamy soil, which is a soil that contains an approximated equalized balance of sand, clay, and silt, while still aiming to gain the benefits of peat by utilizing previous organic matter to increase content of nutrients such as calcium. It has a good balance of water retention and air flow for the plants and it is compact enough for plants to easily penetrate deep enough to absorb water. Due to the nature of this project being a universal proof of concept of the idea the type of soil we plan to utilize for this prototype is loamy soil as it is the most ideal fit to the general plant population. In practice the P-Quad is able to accommodate any type of soil with its housing enclosure, but we chose this for the sake of demonstration. (12)

3.6 Temperature & Humidity

Closely correlating with moisture is the temperature of the environment this is divided into two major classifications air temperature and plant temperature both still measured by typical metrics of Celsius or Fahrenheit. Air temperature is the traditional measurement of the air particles surrounding the plant. While plant temperature is dictated by the air temperature, the relative humidity, and light level; These factors often lead to the plant temperature varying from the air temperature. Plants can try to regulate this temperature by various internal methods such as raising transpiration rate. The temperature of the plant regulates the two processes essential to the plant's life photosynthesis and respiration if the temperature reaches above certain points (generally this temperature is around 25C/77F). For this project regulating temperature is going to be a passive task, since we don't plan to include an air conditioning or heating unit the main thing, we aim to control is to prevent the plant from overheating by utilizing a fan cooling system in combination with controlling the lights in the enclosure. While transpiration and perspiration typically occur at higher temperatures air flow is still an important part to the ecosystem of the plant. These two methods in conjunction ensured a naturalized system similar to what a plant would experience during the day. The fan provided a simulated breeze should the enclosure become too hot and helps push airflow through the filtration system which is discussed at length later. Additionally, disabling the light not only lowers the temperature, but can help the plants regulatory systems maintain a natural growth cycle. (3) (4)

The next factor which needs to be monitored for the overall health of the plant is the humidity levels. A similar concept as soil moisture humidity refers to the amount of water present in the surrounding air expressed as a volumetric percentage of either grams or kilograms per cubic meter. For a plant's growth cycle humidity is important in keeping the pores on the plant lubricated enough so that the plant may breathe properly, should the surrounding air become too dry the plant has a hard time breathing naturally and begin to enter the wilting stages if not addressed soon. This is further complicated by the fact that water absorbed via the roots does not help replenish the water content of the leaves. In the P-Quad we aim to maintain the optimal humidity level for stable plant growth (optimal humidity is defined generally as between 50%-60% humidity level of surrounding air may vary depending on the thickness or coating of the plant in question leaves). The proposed system for maintaining humidity are a combination of monitoring via the sensor and using a misting unit to increase periodically when deemed necessary by the sensor. Since lowering humidity is an energy consuming system that would negatively impact our planned battery life and the option of moving the plant from one enclosure to another is self-defeating the way planned to combat high humidity for the proposed system are entirely preventative by aiming to keep the humidity of the enclosure near the lower levels of the recommended threshold. (4)

3.6.1 Part Decision

For the selection of the temperature and humidity we decided to go with the Honeywell HIH8120-021-001 (or HIH8120). Despite being a more expensive option compared to some of the other sensors the HIH8120 offered several benefits that some of the other sensors didn't have. First it had a great standard accuracy rating with only a 2% drift in accuracy which is the most accurate levels of the sensors we tested. Additionally, the response time for the HIH8120 was 6 seconds making it the second fasters response time second only to the Bosch SensorTech BME280 which would help mitigate internal delays in data readings. It had a very lax voltage rating compared with some of the other sensors allowing for voltages from 2.3V all the way up to 5.5V meaning that this sensor would be able to operate as expected under any voltage conditions that would be present on the board without the need for voltage rectifiers or other voltage regulation-based technology. Additionally, unlike some of the other sensors we were able to find and confirm that the HIH8120 had both a schematic and a footprint present in EagleCAD making its integration into the PCB much easier than those sensors that lacked these two characteristics. Once again similar to the previous sensors comparison we collect all this information into table 7 for an easier comparison visualization.

Table 7: Humidity and Temperature Sensor Comparisons

Sensor	SHT30-DIS-F2.5kS	HDC2010YPAR	BME280	HIH6030-021-001	HTS221TR	HIH8120-021-001
Manufacturer	Sensirion	Texas Instruments	Bosch Sensortec	Honeywell	STMicroelectronics	Honeywell
Voltage-Supply	2.15V ~5.5V	1.6V~3.6V	1.7V-3.6V	2.3V~5.5V	1.7V-3.6V	2.3 V ~ 5.5 V
Operational Temperature	-40°C ~ 125°C	-40°C ~ 85°C	-40°C ~ 85°C	-40°C ~ 100°C	-40°C ~ 120°C	-40°C ~ 125°C
Humidity Range	0 ~ 100% RH	0 ~ 100% RH	0 ~ 100% RH	0 ~ 100% RH	0 ~ 100% RH	0 ~ 100% RH
Standard Accuracy	±3%	±2%	±3%	±4.5%	±4.5%	±2%
Response Time (Seconds)	8	8	1	6	10	6
Price (USD)	$3.93	$3.53	$7.56	$10.00	$3.57	$11.04

3.7 Luminosity

Light levels in plants are another critical factor in a plant's life as without a primary light source the plant cannot properly carry out photosynthesis. Light on a plant is typically measured via either lumens, a derived SI unit for visible light emitted by a source it can also be represented as a unit per area via the lux unit where 1 lux equals 1 lumen/foot^2, or foot candles, a typically antiquated measurement describing the light of a candle from a distance of 1 foot it is equivalent 10.764 lux .Since the light fixture for the P-Quad is artificial and is only be activated by our programs instruction a large consideration to be made is regarding the specific lumen level of a the plant which varies with species. For common plants (crops, herbs, and house plants) the minimum acceptable level of light is 2000 lux, the optimal range is from 7000-7500 lux, however if the plant has a flowering phase it might be beneficial to provide up to 10000 lux at certain periods of the day. Duration of light exposure from plants varies, but typical flowering plants require 14-18 hours of light exposure per day. In addition to the duration and intensity of the light provided, light color is another factor which can impact the plant growth. Plant growth responds best to the full spectrum lights since natural sunlight is a full spectrum light source. (1) (10) (11)

As far as detection of light goes sensors tend to fall into two major categories photocell and ambient light sensors. The former is a type of sensor that changes its resistance value based on the amount of light received (more light provided means higher resistance and inversely intense light means higher resistance). The other, ambient light sensors use an integrated circuit component called photodetector to measure the amount of light (in lux) by using a p-n junction to convert light to current. While photocell type sensors are much cheaper for the sake of this project, we are interested in the intensity level of light specifically measured in lux, so for this reason all parts looked at and compared are of the ambient light sensor type. (13) (14) Again in table 8 we collect all potential candidates into a table for quick comparison.

Table 8: Light Sensor Comparisons

Sensor	Manufacturer	Voltage-Supply	Operational Temperature	Wavelength (nm)	Unit Price (USD)
BH1680FVC-TR	Rohm Semiconductor	2.4V~5.5V	-40°C ~ 85°C	530	$1.51
APDS-9306-065	Broadcom Limited	1.7V~3.6V	-40°C ~ 85°C	560	$1.61
VEML7700-TR	Vishay Semiconductor	2.5V~3.6V	-40°C ~ 85°C	545	$1.94
OPT3001DNPR	Texas Instruments	1.6V~3.6V	-40°C ~ 85°C	550	$3.35
VCNL4040M3OE	Vishay Semiconductor	2.5V~3.6V	-40°C ~ 85°C	550	$2.65
EAALST05RDMA0	Everlight	5V	-40°C ~ 85°C	630	$0.65

3.8 Part Decision

Our final part decision for the light sensor for our project was the VEML7700-TR made by Vishay Semiconductors. There were several reasons for this selection first the part had an operational voltage range that was highly standardized (2.5V~3.6V) and this fit perfectly with the selected MCU, in addition it has very minimal power dissipation of 50mW which is great for saving battery supply of the unit while running. Its operational temperature range provided the device enough protection to be used in this closed environment that may potentially get hotter or colder based off our plant's feedback. While it's one weakness was had the shortest level of wavelength detection it still was within an acceptable range and for the relative economic pricing of the part this was overlooked in favor of it. For the temperature and humidity sensor the group decided on the SHT30-DIS-F2.5kS by Sensirion. The decision to go with this one was several folds. First its operational temperature range was well within any values we expect to see for the environment of the P-Quad. Next it had a good operational voltage and supply current rating (2.15V~5.5V and 800µA respectively) fitting it alongside the MCU and any other embedded electronics. It has the range to sense relative humidity from 0% all the way up to 100% and has an accuracy with a 3% tolerance range. While we sacrificed some in the response time, we still managed to keep it under 10 seconds response time while keeping the highest possible accuracy levels for an economical price compared to some of its competitors therefore ultimately making it the best choice.

3.8.1 Part Decision Addendum

Unfortunately, after our group began designing the PCB, we came upon the discovery that EagleCAD did not have a footprint or schematic design listed for the part meaning that our PCB designers would have to build the part from scratch. Since we had been advised against this on several occasions we decided to go back and pick a different sensor from the list. We decided to go with the EAALST05RDMA0 (EAAL) for short from Everlight for several reasons. First and foremost, we checked in EagleCAD and the EAAL was present in its entirety in the library making it easier to work with. Next it had the same operating temperature ranges as most of the other selections meaning it would work fine the P-Quad environmental conditions. The part came with some minor compromises such as a slightly high peak wavelength detection of around 630nm instead of the average range of the other parts 530-560nm, but it came at it much more economically efficient unit price of $0.65 per sensor compared to the rest of the sensors which were around $2-$3.

3.9 Agricultural Sensors

Another sensor type considered was agricultural sensors which are a class of sensor used for precision agriculture in order to help farmers monitor and optimize growth of plant life. Agricultural sensors come in a wide variety covered a multitude of factors the most common types are locational, optical, electrochemical, mechanical resistance, soil moisture, and airflow. Due to the mature of P-Quad's project scope out of these we decided to look into electrochemical, mechanical resistance, and soil moisture. Of these three, soil moisture was deemed so imperative that it received its own section and analysis, so we won't be discussing it again here. The next which was looked at briefly was mechanical resistance which in this specific context aims to quantify soil compaction. With this technology the sensor uses a probe that penetrates the soil in order to record the resistive force it took to achieve the depth of penetration. The idea by monitoring this would be to accurate gauge how much stress the plants roots had to endure in order to reach its optimal rooted state. However, after several considerations such as the limited depth of the P-Quad's soil enclosure, the fact that loamy soil, which is the planned growing medium of the P-Quad plant, is traditionally very easy for the standard plant root to penetrate, and the fact that by using the soil moisture sensor we can gauge a very similar metric how easily the plant’s roots can extract water from the soil it was decided that a sensor measuring soil compaction was deemed redundant and would not help to satisfy the project scope of the P-Quad and therefore further investigation into this type of sensor was halted. The final type of agricultural sensor we decided to investigate was the electrochemical sensor. The justification behind this type of sensor was it allowed a very tight analysis of the various nutrient levels of the P-Quad's enclosure and other things such as the pH levels of the water present to make sure that the acidity of the water feeding the plant wasn’t too great for it. There are two major classes of electrochemical sensors those that detect via gasses and those that detect via ion content of the soil, for the purpose of the P-Quad we only looked into those that detect soil ion content. A quick rundown on the way it works is an ion-selective electrode usually made of glass or polymer membrane is coated in order to detect the activity of specific ions such as nitrates, potassium, or hydrogen. Once the content of the current environment has been measured it can be compared to an ideal standardized sample (measured in a controlled lab environment) that contains the optimal levels of each chemical and then the user can take the difference to see how far off the nutrient levels of their current environment is. While we felt these sensor technologies would be very beneficial to the P-Quad's project scope upon deeper research several problems presented themselves. The most pressing of these issues is that the technological implementation of these soil ion reading sensors is still fairly early in the market production and the sensors are special order and not widely available yet and those that are ready for purchase command a fairly hefty price. Due to these constraints combined with the fact that the P-Quad’s environment is largely installed and maintained by the user and therefore the amount of nutrients for a plant’s transportation period can be determined beforehand it was decided that the electrochemical sensors be cut from the scope of the final product. (18) (20) (21)

3.10 Inter Integrated Circuit (I^2C)

Inter-Integrated Circuit is a synchronous serial communication bus invented in 1982 by Philips Semiconductor (now NXP Semiconductors). It utilizes a multi-master and multi-slave bus that allows multiple peripherals to communicate with multiple masters. It is a synchronous communication protocol that uses a clock provided by the master to send and receive packets of data. (65)

3.10.1 Design

I^2C uses only two bidirectional open drain or open collector lines. The lines are referred to as Serial Data Line (SDA) and Serial Clock Line (SCL), pulled up with resistors. The I^2C reference design allocates a 7-bit address space with a possible 3-bit addition for a 10-bit total extension address. Common bus speeds for I^2C modules are 100kbit/s standard mode and a 400kbit/s high speed mode. The restrictions on the I^2C bus come from the how many nodes per bus (limited by address space) and the total bus capacitance of 400pF. This restricts reliable communication distances to a few meters making embedded applications a perfect contender to utilize an I^{2}C bus.

3.10.2 Reference Design

When using I^{2}C bus, the nodes have two roles:

- Master node – node that generates the serial clock line and initializes data transmission and reception.
- Slave node – node that receives the serial clock line and returns data when called by the master with matching address.

The I^{2}C bus allows for multi-master environments, meaning that any number of masters may be present on the same serial bus line. The roles of master and slave can be changed in between packets being sent and received. (64)

There is a total of 4 modes of operation: master transmit, master receive, slave transmit, and slave receive. Typically, each of these modes of operation are initiated and terminated with a special START and STOP signals. The START and STOP signals act as message delimiters.

The START, depicted in figure 7, is indicated on the I^{2}C bus by a transition from HIGH to LOW on the SDA line while the SCL line is held HIGH. This notifies the nodes on the bus that transmission is about to occur. After the data has been transferred to the desired node the message initiator then issues a STOP condition in which the SCL line transition from LOW to HIGH while the SCL line is held HIGH. In cases where there are multiple bytes per transaction a repeated START condition which has the same HIGH to LOW transition while SCL is held HIGH as a regular START condition. (68)

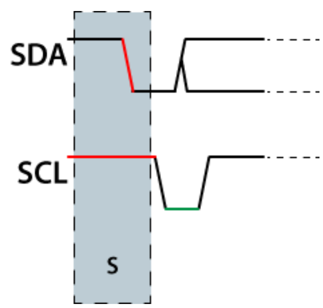


Figure 7: Example of I2C START

The STOP condition shown in figure 8 on the following page, is indicated by a transition from LOW to HIGH on the SDA line while the SCL line is held HIGH. This is a notification to all nodes on the bus that the transmission has stopped and that the bus is now free to be used again. (67)

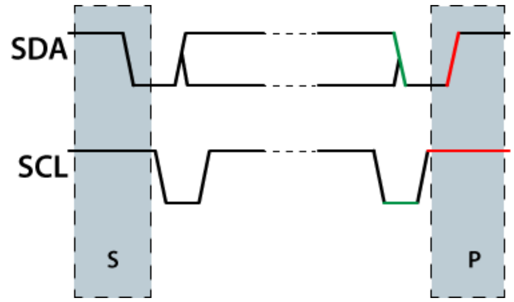


Figure 8: Example of I2C STOP

When multiple bytes of data need to be sent across the I2C bus, after the first packet of data has been transmitted, a REPEATED START condition needs to be initiated. The following figure shows an example of a REPEATED START. This is identified as transition from HIGH to LOW on SDA while SCL is held HIGH and is visualized in figure 9 below. (66)

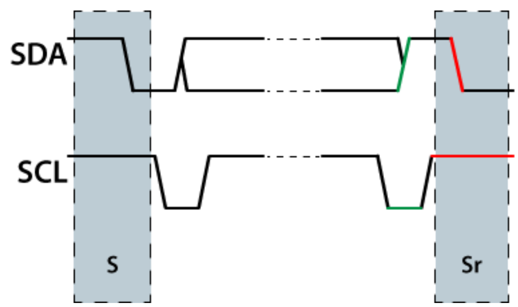


Figure 9: Example of REPEATED START.

3.10.3 Physical layer

Both lines used in I^2C (SCL and SDA) are open-drain design, so pull-up resistors are needed to drive the lines high until they are pulled low by the master or slave. A logic 0 is output by pulling the line to ground and a logic 1 is output by letting the line remain high. A line is never driven high by the master or slave, it can only ever be driven low this allows multiple nodes to access the bus without signal contention from short circuits.

A major consequence of this configuration is that any number of nodes could be driving the line low. Any node that drives the line low are low for all nodes connected to that line. If a logical 1 is trying to be transmitted all other nodes can detect this and reason that another node is controlling the line. This creates a method of arbitration for SDA lines and a form of clock stretching for slaves on the SCL lines.

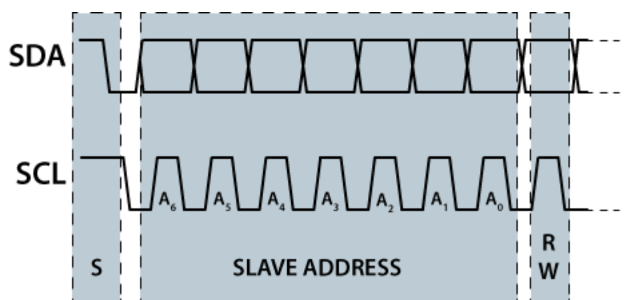
The SDA line only changes while the SCL line is low. The transmission of bits comes when the SDA line is at the desired output and the SCL line pulses HIGH. While the SCL line is LOW, the transmitter sets SDA to the desired output level and after a small propagation delay, allows the SCL line to float HIGH. The SCL line won't be logically HIGH until a finite delay due to the RC time constant of the pull up resistor and the parasitic capacitance of the bus. This delay might be stretched longer if the slave has enabled clock stretching. Before the next bit is sent and the SCL line is HIGH, the transmitter waits a minimum time of 4us (for standard-speed I^2C) to verify that the transmission has propagated to the desired node.

3.10.4 Slave Addressing and Packet Format

When a master wishes to READ from or WRITE to a slave node. The master initiates communication by applying a START condition followed by an 8-bit packet. The 8-bit packet is comprised of a 7-bit address followed by a READ/WRITE bit. The master writes a 1 to the READ/WRITE bit to inform the slave that master is requesting data from the peripheral node. To write data to the peripheral slave the master writes a 0 to the eighth bit to inform the slave that the master writes data to the slave in the next message.

After the slave address has been written to, the selected slave returns an ACK or NACK depending on whether the slave has recognized. The acknowledgement travels in the opposite direction that the 8-bit data was transmitted this process is represented in figure 10. This means that the receiver and transmitter switches roles for a single bit and returns 0 if the data was acknowledged and a 1 if the data was not acknowledged. If a NACK signal is returned the sender learns two things:

- When master is transmitting to slave, the slave was unable to accept the data because there was no such slave, the command was not understood, or the slave could not accept any more data.
- When the slave is transmitting to the master, this means the master wants to stop receiving data from the slave.



- Figure 10: Slave Address and Packet Formatting Example (65)

3.10.5 Clock Stretching using SCL

A prominent feature of the I^2C protocol is allowing slaves to stretch the length of SCL pulses to arbitrate data. If a slave is using clock stretching the device can hold the SCL line low after sending or receiving a byte of data, notifying the sender that it is not ready to process more data. The transmitter can’t complete the transaction until the SCL returns to the HIGH state. The same is true if a second slower master tries to drive the clock at the same, therefore if there is a multi-master bus all but one of them typically loses arbitration. Clock stretching is the only time that the slave can drive SCL line.

3.11 Display Research

In P-Quad all the in the information that the user inputs or all the collected information about the plant or the stored data have to be displayed for the user to read and understand. For this we are implementing a display screen on the outside of P-Quad. To determine which display to implement in P-Quad our group looked at various different types of display.

3.11.1 LCD

The term LCD is an acronym for "Liquid Crystal Display" and is a very common type of screen used for displaying information. Liquid crystal technology works with two pieces of polarized glass, also called substrates. Contained within the substrates are liquid crystals that allows or stops light from passing through them. The first layer of substrate allows polarized light waves to pass into the liquid crystals. To change the amount of light and frequency of light emitting from the liquid crystals an electrical current is applied to each area of crystals. This causes the liquid crystals to align and emit different colors after passing through the second substrate allowing us to see the different pictures and colors.

3.11.2 Active and Passive Matrix Displays

The terms active and passive refer to the way that the voltage for each pixel holds the charge required to display the correct pixel. The pixels are 'active' when the pixels are maintaining their current state while other pixels are being addressed compared to 'passive' in which each pixel must maintain its state passively without being driven by a circuit.

3.11.3 Passive Matrix Displays

In passive-matrix addressing schemes the two layers of substrates act as a row substrate and column substrate. The pixel is activated by applying a voltage down the correct column of one substrate and a ground activated on the correct row of the pixel. The intersection is where the voltage was applied. This system is very simplistic however, there are a few drawbacks. The first is the slow response time due to the time it takes both substrates to change the voltage of each pixel that changes. The second drawback refers to the imprecise voltage control. When a voltage is applied to one pixel the pixels surrounding it are also slightly varied, making images look unclear and fuzzy due to lack of contrast. The lack of clarity and response time led to a new way of addressing pixels using active-matrix displays.

3.11.4 Active Matrix Displays

Active-matrix addressing scheme utilizes capacitors and thin-film transistors (TFT) to capture the state of a pixel. TFTs are made by depositing thin films of an active semiconductor layer as well as the dielectric layer and metallic contacts over a supporting but not conducting substrate. TFTs are important to limit the number of connections needed to illuminate a pixel. Instead of using a transistor for each color of the pixel (one for each red, blue, and green), the pixels are addressed in rows and columns. Therefore, for an M x N matrix the number of connections goes from M*N to M+N. This vastly reduces the number of connections. Using the transistors one-way current passing characteristic, the transistor prevents the charge that is being applied to each pixel from being drained between refreshes to a display's image.

3.11.5 LED

LED screens utilize the same technologies that LCD screens have been using. The difference however, is that LEDs are used to provide the backlight rather than the fluorescent bulbs used in LCD displays. This makes LED screens far more efficient in terms of energy and smaller in size (the screen is much narrower). LED screens can be further grouped into two categories: Direct (back-lit) and Edge-lit.

Direct (Back-lit) LED screens use an array of LEDs directly behind the layer of substrates. This enables areas to have more focus on light – meaning each specific cell has more contrast between bright and dark. Edge-lit LEDs use exactly what it sounds like, the edges are lit and are reflected into the center of the screen. This lets the screen become even thinner and lighter than the direct lit LED screens.

3.11.6 OLED

OLED displays, contrary to what the name suggests, don't derive in any direct way from LED displays. OLED is an acronym that stands for organic light-emitting diodes. Instead of using a liquid crystal and voltage combination to determine color and brightness, the OLED screens use an emissive electroluminescent layer in place of liquid crystals. The layer is made up of a film of organic compounds that emit light in response to electrical currents. In between this film are two electrodes, one of which is typically transparent, that supply the current to emit light. This makes OLED screens very powerful in that they don't need an external light source to produce a picture. Now that each pixel is controlled individually, rather than with the same backlight, the contrast and colors are much greater on OLEDs. Without being restricted by an external light source, OLED screens are extremely thin and most notably, flexible.

3.11.7 Plasma

Like LCD and LED screens, plasma screens have two panels that contain photoelectronic materials that emit light when charged with electricity. During the manufacturing process electrically charged ionized gases are sealed in plasma form and injected in between the two sheets of glass. Plasma screens are superior to LCD and LED screens in terms of their contrast and color accuracy. The format however is an issue in our specific setting. Plasma screens only become cost effective in the larger screen sizes, usually the price threshold becomes feasible in sizes greater than 40 inches. This poses a problem for our project due to size constraints.

3.11.8 Seven Segment Display

The last option to display necessary functionality and control of our device is a standard seven segment display or SSD. In standard SSDs pins that are connected to each individual segment are selected based on the alpha-numeric data that is to be displayed. Each combination of segments can represent a different symbol or number.

3.11.9 Display Selection

After collecting the research on the different possible display types, there were several factors that led us to our final selection. The first criteria we looked at was the size screen required for our design. The first option that can be crossed off our list of possibilities was the Plasma screen option. Plasma screens were only available in very large screen sizes, typically over 40 inches, which is not a feasible size for a portable, battery operated device. The next criterium was the amount of information that we can display using the different display technologies. LCD, LED, and OLED are all capable of displaying any image or text that we might choose to display our data. The only display type that truly limits our capabilities, in terms of amount of data displayed, are the SSDs or seven segment displays as the range of display capabilities for SSDs is highly restrictive for the amounts of data or text we wish to display at a given point. For our display we decided to go with a QPASS 1602A LCD screen. From Arducam Our reasons for selecting this part were primarily two-fold the first being one of our group members hard the part leftover from a previous project meaning that this would help supplement some of the cost of buying parts and even if we did have to buy another couple of modules for the prototyping the cost of the 1602A is only around $4-$6 a unit. The other primary reason is because we decided that an LCD screen would be the most fitting choice for the P-Quad. We decided this due to reasons like price range of screens, ease of integration into the PCB, durability of the display coupled with certain aspects like the limited size space and the amount of content we would need to display at any given time we decided that LCD fit all of these constraints better than the other categories of displays.

3.11.10 Arducam LCD Module ST7066U-0A

This is the LCD module that we have decided to use for P-QUAD's display portion of the device. There are several features that fit the needs of the specifications. The display can show sixteen characters and two lines for a total of 32 characters total. It also allows us to scroll through data text, making it easy for us to string multiple words together into sentences which is what is needed by P-QUAD's display section of the device. Each of the character displays are composed of a five by 8 dot matrix to display characters. This gives us a wide variety of symbols to choose from. The display also features its own DDRAM that stores the characters that are used in the text to be written onto the display. This is very useful to us because this means the display can function on its own without any software needed in the microcontroller. That also means that we as programmers only need to initialize the LCD display a single time.

3.12 On-Device Storage

An issue that came up a little later in the design cycle of the P-Quad was the issue of storing all the data readings from the sensor to some sort of storage device that was directly onboard with the PCB or immediately attached to it. This issue came about because despite the simplistic nature of that data being stored, more than likely a Unicode encoded raw text-based file format like .txt or formatted text-based file like .csv or. json. While the formatting and communication of these file types is easily worked around where we would encounter difficulty is the size of the files, we would need to store. While text files are traditionally very small in space consumed there would be potential for large scale growth of our files due to two major reasons, the first would be how often we sample the plant's environment via the sensors and store it. Suppose for instance that our slowest sensor samples every 10 seconds we would then set all sensors to sample around the same rate of 10 seconds during periods of reading then suppose that these read periods last a minimum of 1 minute or 60 seconds generating 6 different reads during the period then multiple those 6 read cycles by 4 (one for each sensor reading) would generate 24 lines of text in the file which roughly equates to 1.7KB (under the assumption that 1 line of text storage consumed is around 70 Bytes). While this sounds like a paltry file size in today's big data world the only storage space on our MCU would be its internal RAM which totals at 8KB meaning that we could fit around 4 read periods at most, assuming that we would be willing to sacrifice the performance by eating up all the RAM for just storage. Now let's assume the P-Quad enters these read periods around 5 times every hour which we can express as 120 times per day this total file size would be around 202KB which exceeds the entire MCU's memory program, ROM, and RAM combined making it whole unrealistic to store this data directly on the MCU. The other issue comes in with persistence assuming that we could fit the file entirely on the MCU (which isn't possible since the ROM and program memory are inaccessible to use during runtime) RAM is a form of volatile storage and the data would be lost upon power down or fluctuation. Since one goal for the P-Quad is to be able to view the data over time in order to gauge how table the plant's environment over a time period is this constraint presents a big problem that we can only alleviate by adding an additional memory component. (60)

Having established that we need an additional memory storage element we can now proceed to look into the 3 major types of storage devices for the scale of project we are making. The first is internal flash memory known as EEPROM (electronically erasable programable read only memory) which despite its name is able to have its data written over however only for a very limited number of cycles (product average tends to indicate around 100 write cycles). Another note to be made about internal flash memory is that is typically expressed storage in bits not bytes meaning that the larger sized products of ranging around 4Mb store in actuality store only 500KB averaging a little over 2 days' worth of readings based off the estimated file sizes established prior. This solution proves viable, but a very poor one so we continue to explore other options. The next type of technology examined is external NAND flash. This type of IC would sit close to the MCU and have all the data relayed over to it very quickly. (59)

The next type of technology examined is external NAND flash. This type of IC would sit close to the MCU and have all the data relayed over to it very quickly due to the 8-bit parallel interface used to drive this type of memory. Additionally, the storage volume of these devices is much greater than their internal counterparts while still being measured in bits usually averaging around 2-4Gb making them capable of holding files around 250MB-500MB. While still susceptible to wear they often have a lot more write cycles in their life cycle (typically around 100K or greater) before they burn out and can be fitted with wear-leaving algorithms to combat this. However, two major drawbacks are present with this storage type the first being that in the case of failure the entire memory unit is shot, and not only is all data is irreplaceably lost but the board would have to undergo extensive repairs in order to be usable as intended again which in the case of the average end user renders the entire product useless. The other and more pragmatic rather than design issue is these types of memory are extremely small for the amount of storage volume one receives with a 4Gb device costing around \$15-\$20 compared to hard disk passive storage which is around \$50 for 1TB (which is 8000 times larger than a 4Gb device). While a much better choice than our first two options and entirely viable we decided to review one more type of storage that we felt would prove a more worthwhile solution to the previous types.

Our last storage type we examined was using an onboard micro-SD card reader. What this would entail is attaching a card reader onto the board similar to the external flash, but now the memory would be tied to the micro-SD that would be inserted into the reader itself as opposed to a component locked on the board. The first unique trait that this possesses that our other types of storage didn't is that it is modular by design, to elaborate once we put the card reader into the system the actual micro-SD cards themselves could be removable and exchanged. This ability leads to a few useful things, the first and most obvious being that in the case of one micro-SD card failing the end user would merely be able to swap in another micro-SD card only potentially suffering from some data loss as opposed to catastrophic device failure. Another benefit to using this method allows users to remove the readings at any given time and make backups of them or to manually wipe out the data if desired. This could prove useful for a variety of reasons like if the P-Quad had reached its destination with one plant and was then being sent off to another with a different plant in tow, while new readings would come in fine for the new plant the old ones might not necessarily be at the period where they are removed by the software and could potentially lead to confusion and concern if the environmental conditions for the other plant are different enough from the first which might lead to false positive control conditions being activated (such as pumping more water or providing less light) which then might actually hurt the plant. The other situation could be if someone was using the P-Quad as a controlled environment to conduct research they would want to make periodic hard backups of any data that might prove vital to the research and this could be easily achieved with the removable micro-SD card model. Price wise compared to the other solutions the removable card slot is in-between the internal and external flash memories in total cost with the slot itself ranging from $2-$3 and the micro-SD card ranging from $5 for a 32GB card to $20 for a 128GB card which would provide enough space the 32GB providing around 158415 day of continuous writing or around 434 years before filling up the available space at the estimated rates from before. A final added bonus is that much like the external flash these micro-SD cards last for hundreds of thousands of write cycles before malfunction and come with a memory controller built in which handles the wear-leveling of the devices to ensure equal usage of sectors. It is for these combined reasons that we decided to use an onboard card reader with an external micro-SD card as the memory storage solution of the P-Quad.

3.13 Electrical Relay

A relay can be defined as a device that provides an electrical connection between two or multiple points basically relay is an electrically operated switch that open or close electrical circuit electronically, they can be used to control one electrical circuit by either closing or opening contacts in another circuit. Relays are used when it is necessary to control a circuit or when several circuits must be controlled by one signal. A relay has part that allow it to operate at a very fast speed. The basic parts and their functions include.

- Frame, which is the outer most part of the relay and it contains all the moving parts that make up the relay. Frame itself is made of heavy-duty metal.
- Inside the frame, to create a magnetic field a long wire is wound around a metal core, which is called a coil, this coil is what makes the electromagnetic field possible.
- Armature is the moving part of a relay. This armature is what opens and closes the contact when the coil is energized. After a contact is either opened or closed a spring that is attached to the armature returns it to its original position.

- Contacts are the electrical conducting parts of the relay that closes or opens a circuit.

3.13.1 Relevant Electrical Relay Technologies

An electromagnetic relay is classified in two different types, attraction relay and induction relay. Attraction relay has an armature that is pulled in to the magnetic pole. The electromagnetic field is applied on the mobile component corresponding to the square of the current flow. But like previously mentioned, alternating current produces a frequency that is almost doubled thus, giving of damaging noise. The other type of electromagnetic relay is called the induction relay. The main principle behind this relay is that the split phase motor. The power driving the component is produced by cooperation between electromagnetic fluxed in addition to a current. Each of these flux initiates a voltage around its rotor. Therefore, the current flowing in one flux reacts with the other flux. This translate to P-Quad in a way that makes it almost a no brainer when it comes to which relay would be best and it is the solid-state relay which is described in the following chapter.

3.13.2 Electrical Relay Selection

To protect P-Quad we decided to use a solid-state rely mainly because they are much faster than electromechanical relays. An electromechanical relay relies upon the time that is needed to switch an LED on or off. Because solid-state relay has no moving parts, it has a much longer life expectance compared to other options available. Another big advantage solid-state relay has is that the output resistance always remains constant even if the relay is used a lot. The solid-state relay does not have any sparking which makes it ideal to use in an environment like the pod where there is a water pump to maintain moisture in the pod; furthermore, due to lack of sparking in the rely it can be used in environment that are explosive or combustible. When using an electromagnetic relay, whenever the contact is opened or closed there is a sound that is generated; however, with the solid-state relay there is no such issue, it offers a completely silent operation which is always preferable specifically in P-Quad. Even though the solid-state relay and the electromagnetic relay have the similar specifications, the solid-state relay is much smaller than the mechanical relay. keeping in mind that the primary function of P-Quad is to transport plant life and other specimen and knowing during transit there can be much turbulence and bumps in order to make sure the contact in a relay do not open or close due to this disturbance solid state relay has to be implemented because a SSR is not nearly as sensitive to the environmental factors that can be present during transit which means that a solid-state relay is not susceptible to any chock or humidity which is exactly what was used in P-Quad. The solid-state relay that we decided to use is G5LE-1A4 DC24. The G5 relay has a large drive current and a 24-volt input voltage which is higher than most mechanical relay. The G5LE relay also has a big feature, it can operate with either DC or AC power input which is what the P-Quad needs because when P-Quad is not in transit it is plugged in a wall to run on AC power however, during transit it used DC from a battery source. To power the relay there is a solder pin on the relay which has a 1440 ohms coil resistance which keeps the relay from over-heating. The contact rating is 10A which is high enough for the relay to function with P-Quad perfectly. The turn on voltage for the relay is 18-Volt DC. These comparisons are displayed in table 9 below and 10 after the following page.

Table 9: Electric Relay Comparisons

Relay	G5LE-1A4 DC24	G5LE-1A4 DC12	G5Q-1A4 DC12
Coil Res	1.44 kOhms	360 Ohms	720 Ohms
Cost	1.3 USD	1.3 USD	1.33 USD
Coil Current	16.7 mA	33.3 mA	16.7 mA
Coil Voltage	24 VDC	12 VDC	12 VDC
Contact Rating	10A	10A	10A
Turn on Voltage	18 VDC	9 VDC	9 VDC
Coil Power	400 mW	400 mW	200 mW
Release Time	5 ms	5 ms	5 ms
Mount Type	Through Hole	Through Hole	Through Hole

What is the primary reason to have a dependable relay implemented in any system? Like discussed previously there is always a chance of a power surge which can cause an over current condition which in turn can either completely or partially damage a system; furthermore, over current can lead to a possibility of arc flash or arc blast which can harm any users in the vicinity. Some industrial relay that are used to protect a system which is operating under a lot of voltage or high current needed to have its circuit breakers open almost instantaneously and in order to do that fast the relays have to send the signal of a fault condition to the breakers fast. Some of the fastest relays operates in milliseconds. The G5LE operates in 10ms and has a release time of 5ms, which makes us feel that this is one of the fastest relays that we can implement in P-Quad so in case there is a power surge or an over-current condition or any other fault the relay switches, and P-Quad is safe from damage. Below is a table that shows all the features that is offered in this relay plus some of the other relays that we considered.

3.14 Light Fixture/Lamp Research

3.14.1 Incandescent Lamps

Incandescent bulbs are essentially controlled fires on display. When an electrical current is applied to the base of the bulb the current is passed through a filament, typically tungsten, the electricity heats the filament until the light is emitted. This is the same light that we see when a fire burn. As the filament burns, small particles are removed and when there are no particles left to be burned the bulb burns out. This means that the light being produced by the filament also produces high amounts of infrared radiation, or heat. In our biological environment, heat when not desired could be detrimental to the plant specimen's life. The amount of light produced in relation to the amount of heat of an incandescent bulb is approximately a ten to ninety percent ratio, which is horribly inefficient. The best part of incandescent lamps however, is their resemblance to regular sunlight. Incandescent bulbs are the closest light source we can get to the sun without the use of multiple different bulbs and lamps.

3.14.2 LED Bulbs

LED bulbs work in a similar fashion to the incandescent bulbs. Applying an electrical current to the base of the bulb passes the current through a filament. This filament has special characteristics that produce light when applied with an electrical current. In the case of the light emitting diodes, the filament is a semiconductor that fires photons when the electrons pass from a negatively charged semiconductor to the positively charged semiconductor. This makes LED bulbs much more efficient in comparison to the vacuum/filament combination of incandescent lamps and bulbs. They consume much less energy over time and also emit more light per lifetime of the bulb. The trade-off comes in terms of cost. The cost of single LED bulbs could be upwards of 40-50 times more expensive in comparison to fluorescent and incandescent bulbs. However due to the efficiency of the semiconductors in the long-run LEDs always prove a cheaper option to the standard filament bulbs.

Table 10: Electric Relay Comparisons 2

Relay	Coil Res	Cost	Coil Current	Coil Voltage	Contact Rating	Turn on Voltage	Coil Power	Release Time	Mount Type
G5LE-1A4 DC24	1.44 kOhms	1.3 USD	16.7 mA	24 VDC	10A	18 VDC	400 mW	5 ms	Through Hole
G5LE-1A4 DC12	360 Ohms	1.3 USD	33.3 mA	12 VDC	10A	9 VDC	400 mW	5 ms	Through Hole
G5Q-1A4 DC12	720 Ohms	1.33 USD	16.7 mA	12 VDC	10A	9 VDC	200 mW	5 ms	Through Hole

3.14.3 Fluorescent Lamps

Fluorescent lamps are based on the excitation of low-pressure Hg gas by an electric current. When the excited electrons return to their ground state, they emit photons at 254nm (UV-C radiation). These photons are absorbed by a phosphor coating on the inside of the tube. This phosphor emits, or fluoresces, at longer wavelengths. Most of these phosphors emit in the visible light spectrum.

The major benefits of fluorescent lamps are the availability of inexpensive fixtures and bulbs, low IR emission (low heat), long lifetime, and stable spectral output. However, the problem is that the irradiance level of fluorescent lamps is relatively low; this makes it difficult to build a lighting system that is the same as sunlight. Fluorescent lamps only provide about 20% radiation of full sunlight.

3.14.4 Compact Fluorescent Lamps

The descendent of regular fluorescent bulbs, compact fluorescent bulbs are just the more energy and space efficient to their predecessor. The newer compact fluorescent bulbs have a gas that is easier to excite than in their previous versions. Because of their similarities I will not digress into the same explanation of the science behind fluorescent lighting fixtures. I do however want to explain some drawbacks of using these as a lamp.

The main fallback of compact fluorescent bulbs is the time required for them to produce their best results. The gas molecules contained in the bulb usually take around fifteen minutes to become completely excited and give off the maximum amount of light. Another potential drawback lies in the amount of heat produced by compact fluorescent bulbs in recessed areas. This is due to the amount of electricity needed by the gas to produce light. If there isn’t enough space for the heat from electrical current to dissipate, the surrounding areas might be exposed to excess amounts of heat.

3.14.5 Lamp Selection

After discussing the different types of lamps fixtures, we could use for P-QUAD we came to a team conclusion that we need to pick a very energy efficient bulb that is both compact in size and produces a high number of lumens.

In this section we discuss the selection of our lighting fixture to be embedded onto P-Quad. This fixture served as a substitute for sunlight, providing the necessary nutrients for the plants’ survival. The selected part must efficiently run under our voltage constraints and be small enough to fit in our dome enclosure that is encapsulating the plant. With all that said, most importantly there should be enough light to be emitted from the fixtures for the sustenance of the plant.

The first characteristic the LED fixture should have is to have a maximum voltage rating of maximum 5 volts. The reason for that is our PCB’s maximum pin output voltage is 5 volts. Our second characteristic is for the LED to emit, at the bear minimum red and blue of the light spectrum. Green light is not really required because it is the least efficient wavelength when it comes to the betterment of the plant. Generally, chlorophyll reflects green light making its absorption close to obsolete. Ideally, we would want to use white light, so it can mimic actual sunlight, to an extent, with all the light spectrums included within it. Another characteristic of the LED fixture is to find the delicate balance between size and light emission to maximize its efficiency. The last characteristic is for the fixture to have a method of control, either by a button or a switch.

The light fixture selected is a multiple array of LEDs equally spaced out and emit a significant amount of light. Permission letter to integrate this product to P-Quad has been received by the creator, but it is in pending status as we have not yet received a reply. It is a hand-made and soldered LED device that was built by an electronics seller on eBay. To describe the selection, it is rectangular with two wires sticking out of the end of the circuit. We do have back-up almost identical LED device in the case that the seller does not grant us permission. It follows United States of America electrical conventions, as the red wire signifies the positive terminal and the black wire signifies the ground terminal.

This was a perfect selection as a lighting fixture due to multiple reasons. Our first reason is that the fixture emits light at our required efficiency, and it delivers concentrated, high lumens at low power consumption, which is ideal for our automatic back-up switching system. The second, a significant reason, is that it runs at approximately 5 Volts which fits perfectly under our power supply constraints of a maximum dissipated voltage of 5 Volts. Third reason is size, it is a very small yet high brightness-emitting device, so it would fit well inside our dome enclosure. Fourth and final reason is the LEDs' emission of white light. Having white light supplied to the plant assures it can absorb all color ranges of light of the visible spectrum, mimicking a more natural plant environment.

Since the LED light fixtures are going to be attached onto the inner side of the P-Quad dome, above the plant, there is one potential problem we might experience, moisture build-up. The watering process of the plant can cause significant moisture build-up in the dome causing any non-protected electronics to possibly short-circuit. In addition, the high light intensity can generate more heat and cause a significant increase in the evaporation process. Condensation can, and if not properly protected, cause short circuiting of certain components that are not built to handle. There are two ways we can approach this, either replace the current fixture to a waterproof one or create a pocket that protects the fixture from most of the condensed water.

Assuming we are to stick with the current light fixture selection, we have to shield it in some way. There would be no possible way to install it and not get moisture build-up over time. If we enclose it in a soft rubber bubble with only two holes small enough for both wires to pass through, water vapor would still find a way of going inside, dampening the bubble. Placing it inside a rubber shell can also subject to fire hazard when running for a long period of time due to the heat not having the potential to dissipate elsewhere. Another protective approach would be placing the fixture inside a hard-plastic case and be powered on its independent circuit using a battery. This can allow the light to travel through the transparent case while protecting the fixture from short-circuiting. The second approach seems to be the optimal approach to the problem, but if we were to go that route, the preference over that would be to purchase a vapor-tight sealed LED fixture build for those kinds of applications.

The lighting fixture marketplace are filled with vapor-tight LEDs, as at my current internship as an electrical designer, there are some place that require that trait by the National Electrical Code. These electrical related laws are set-up to protect individuals, equipment and the seller's "waterproof" guarantees. Purchasing an LED array that is already set-up and protected against condensation might seem like the inevitable choice. Whichever we decide as our final selection, reducing moisture build-up on our device is a requirement and must be addressed with the highest of priority to prevent any safety concerns or malfunctioning components.

3.15 Water Storage and Pump Mechanism

In the P-Quad we propose to have a mechanism in which the plants water levels are not only monitored but adjustments to water content can be made by an integrated pumping system when it senses that the water threshold has dropped below a suboptimal point. The proposed system in which this is accomplished is planned to include both an on-device water storage unit as well as a pumping and delivery system that provides water to the plant once a signal has been asserted that it drops below this preset water threshold level. The first thing we needed to consider is where the water would be stored because if we wished to provide water to the P-Quad's plant we would need to incorporate that water supply into the system somehow. This portion went through several revisions due over the course of development. At first it was proposed that inside the housing base would be a walled off portion of the housing enclosure that would hold a premeasured volume of water that the pump would be placed adjacent to along with a piping system to deliver the water to the plant. This method however presents a plethora of problems, first there was the concern that the housing enclosures internal sealing would not be sufficient to stop the water from leaking slightly into the rest of the chamber since our plan was to 3D print the enclosure. Since none of us have a large knowledge of mechanical design it felt this method was too risky as if we couldn't be sure of the sealing rating of the material used to print the housing was up to standard then we couldn't have confidence that water would not leak out and potentially damage the hardware of the P-Quad since this housing base was also planned to hold the main PCB and power supply to the system and regardless of the amount any contact of water with the PCB could lead to a number of problems ranging from undefined behavior to outright catastrophic failure of the specific peripherals or the entirety of the PCB. Alongside the concern for potential leakage this method raised another issue sweating.

3.15.1 Water Storage First Design Draft

Elaborating on this we began under the assumption that we could say with absolute certainty that the separation between enclosures would ensure complete isolation of sections and we could guarantee that no water would leak from once section into the other (of course this assumption was already considered to be practically impossible for our budget and level of experience) but even after those assumption we were left with questions what about sweating and humidity levels of the internal enclosure, by sweating we refer to the common occurrence of moisture collecting on the outside typically in droplet shape formed by surrounding condensation caused by the difference of the temperatures of the container holding the liquid and the surrounding temperature of the container. While the amount of liquid typically produced via this condensation is small in scale and size, we still run the risk of having our exposed hardware in the same enclosure as a liquid which could potentially lead to problems. Similar to the previous issue having the water inside the enclosure risks in running up the humidity levels inside of the enclosure and once again these increased humidity levels could lead to potential problems just as the previous.

Finally another concern was with regards to extra water remaining in the enclosure leading to troubles such as mold, mildew or the eventual weakening and erosion of the internal housing (which admittedly while it wouldn't wear down entirely outright it could in theory wear down enough to allow additional moisture or microscopic organisms inside the inner housing compartments) which would lead to long term use case issues .In order to begin to rectify this situation we started to propose solutions that would ensure complete isolation of water supply while addressing the potential of sweating and humidity to mitigate these issues as much as possible.

parallel interface used to drive this type of memory. Additionally, the storage volume of these devices is much greater than their internal counterparts while still being measured in bits usually averaging around 2-4Gb making them capable of holding files around 250MB-500MB. While still susceptible to wear they often have a lot more write cycles in their life cycle (typically around 100K or greater) before they burn out and can be fitted with wear-leaving algorithms to combat this. However, two major drawbacks are present with this storage type the first being that in the case of failure the entire memory unit is shot, and not only is all data is irreplaceably lost but the board would have to undergo extensive repairs in order to be usable as intended again which in the case of the average end user renders the entire product useless. The other and more pragmatic rather than design issue is these types of memory are extremely small for the amount of storage volume one receives with a 4Gb device costing around $15-$20 compared to hard disk passive storage which is around $50 for 1TB (which is 8000 times larger than a 4Gb device). While a much better choice than our first two options and entirely viable we decided to review one more type of storage that we felt would prove a more worthwhile solution to the previous types.

The last storage type we examined was using an onboard micro-SD card reader. What this would entail is attaching a card reader onto the board similar to the external flash, but now the memory would be tied to the micro-SD that would be inserted into the reader itself as opposed to a component locked on the board. The first unique trait that this possesses that our other types of storage didn't is that it is modular by design, to elaborate once we put the card reader into the system the actual micro-SD cards themselves could be removable and exchanged. This ability leads to a few useful things, the first and most obvious being that in the case of one micro-SD card failing the end user would merely be able to swap in another micro-SD card only potentially suffering from some data loss as opposed to catastrophic device failure. Another benefit to using this method allows users to remove the readings at any given time and make backups of them or to manually wipe out the data if desired. This could prove useful for a variety of reasons like if the P-Quad had reached its destination with one plant and was then being sent off to another with a different plant in tow, while new readings would come in fine for the new plant the old ones might not necessarily be at the period where they are removed by the software and could potentially lead to confusion and concern if the environmental conditions for the other plant are different enough from the first which might lead to false positive control conditions being activated (such as pumping more water or providing less light) which then might actually hurt the plant. The other situation could be if someone was using the P-Quad as a controlled environment to conduct research they would want to make periodic hard backups of any data that might prove vital to the research and this could be easily achieved with the removable micro-SD card model. Price wise compared to the other solutions the removable card slot is in-between the internal and external flash memories in total cost with the slot itself ranging from $2-$3 and the micro-SD card ranging from $5 for a 32GB card to $20 for a 128GB card which would provide enough space the 32GB providing around 158415 day of continuous writing or around 434 years before filling up the available space at the estimated rates from before. A final added bonus is that much like the external flash these micro-SD cards last for hundreds of thousands of write cycles before malfunction and come with a memory controller built in which handles the wear-leveling of the devices to ensure equal usage of sectors. It is for these combined reasons that we decided to use an onboard card reader with an external micro-SD card as the memory storage solution of the P-Quad.

3.15.2 Water Storage Second Design

The next major solution we came up with was a removable storage container that would sit inside of a crevice in the housing enclosure this crevice would be a separate slot grafted into the enclosure and then the water would be held inside a small removable casing that would be held inside the crevice similar to how a test tube is held inside of a rack. The pump would connect via a tube that would provide water to the plant with a similar mechanism as the first iteration. Ultimately this design was also too flawed for us to implement with it bringing several problems to the table as well. The first and foremost would be the amount of work and the integrity of the enclosure design in implementing this one of the main being we couldn't ensure that the container once inside the crevice would be able to consistently remain at rest by which we mean if during the movement of the P-Quad it might be possible for a sudden jolt to dislodge or cause a big spillage of the water inside of the container which could then potentially seep through and get inside of the PCB housing section of the enclosure and lead to all the potential problems discussed previously. Another issue was we couldn't be sure that condensation would form on the underside of the container similar to a stalactite and again lead to issues regarding condensation getting inside of the housing enclosure which could again lead to issues with unwanted moisture or higher humidity. Finally, one issue that this one brought up was an issue of structural integrity since this solution would take a large chunk out of the enclosure there was concern that the material wouldn't be strong enough to support addition weight placed on the housing such as the dome or in the depression for the plant pot and might crack after an extended period of time or during transportation.

3.15.3 Water Storage Final Design

The next design which was the one we settled on aimed to again rectify the issues present with the previous solutions while still aiming to ensure complete isolation of PCB from any form of moisture from humidity, liquid condensation, spills, or otherwise. In this version we decided that the simplest yet most efficient solution would be to create the water storage tank in an external box with the pump system connected to that and then strap it to the side of the main housing enclosure using Velcro strips placed on the sides of each connecting segments. This method ensures complete isolation of the water from the areas which could possibly contain the PCB ensuring that we would not run into the issue of moisture coming into contact with the hardware. Additionally, with the two walls of separation between the PCB enclosure and the water storage tank we ideally have removed the potential of sweat penetrating the inside of the enclosure where the PCB is located. Furthermore, with the tank being easily removable the users have the ability to refill the water on demand and clean the water storage tank which can help us combat the issues of mildew or mold. For this design the delivery and pumping mechanism remained largely the same as the first two iterations with one acceptation being made to the cover for the enclosure. Since the water and pump system were now external the pipe would have to go through a guided bore hole the external cover that would then be placed in the soil. The unfortunate compromise is one should remove the pipe from the inside before taking the plant cover off so that the removal of the plant cover didn't move the entire pump system. While this left a small opening on the side of the cover that could raise issues during travel this issue is readily mitigated by creating a small cover cap to place in the open hole during times when full sealing is required.

3.15.4 Pump Overview

Next is the discussion of the pump system and how it all is planned to work in order to deliver water to the P-Quad. First a brief description on what a pump is and how it generally operates. Pumps are designed to displace fluid from one location to another location and are driven by one of three mechanical actions (direct lifts, displacement, and gravity). Pumps are operated with a mechanism typically reciprocating or rotary and consume energy provided by a source, such as wind, combustion, or electricity in order to accomplish the displacement of the fluid. With these facts in mind for the scope of this project design we decided to look exclusively at electrically power pumps as alternative energy sources such as wind or solar could prove difficult to provide continuously during transportation and potential harm from emissions from a combustion engine driven pump might prove harmful to the plants continued preservation and therefore was dropped from consideration. When it came to commercial solutions for electric water pumps we had several factors to consider two of the most pressing matters being the size of the pump which ties in with its feasibility to fit in or around the water storage tank and the lift height the pump can push the fluid to as if the pump is unable to drive the water from the tank to the plant then it would prove a poor solution. Other factors to consider was how long was the cord to supply power to the pump or could we find a pump that was battery operated and how much wattage would the pump draw as this could potentially compromise the user goal of being a low powered system if the pump drew too much power.

Table 11: Comparison of Electric Water Pumps

Pump Model	VicTsing 80	VicTsing 120	Homsay 53
Dimensions LxWxH (in)	1.87x1.68x1.24	2.56x1.65x1.95	1.26x1.54x1.65
Cord Length (ft)	5.9	4.1	6.1
Power Draw (W)	4	6.5	2.4
Lift Height (ft)	2.6	2.6	1.64
Fitting Nozzle (in)	.33 and .51 interchangeable	.33 and .51 interchangeable	.30
Gallons Per Hour	80	120	53
Price	7.99	9.99	8.99

3.15.5 Pump Selection

After digging into options for a pump that fit the majority of these parameters we came to the conclusion that a small aquarium water pump would best fit the type of workload we wanted to put on the pump in addition to meeting our other requirements such as being electrically powered, submersible, being small in the amount of space it consumed relative to the design. For a cost-effective market solution, we found the vendors VicTsing and Homsay to provide that product line that is the most accurate fit to our use case. These lines of aquarium and fountain hydroponic water pumps fit all the paradigms we were looking for in a pump including the submergible, small scale, powerful enough to drive the water the distances we need it to reach and is a low powered electric water pump. Even with this narrow selection of vendors we needed to decide which model they offered fit us the best. We discounted anything starting at the 320GPH (gallons per hour) pump and beyond it as they are required a wattage of 20W or greater making them poorly suited to this project. From their remaining models that left 3 main options which are compared in table 11. Looking at the comparison of the pumps and their benefits and drawbacks as they related to the P-Quad project we decided to go with the VicTsing80 as it provided a good middle ground of both low power and a large enough lift height to be able to push the water from the external storage tank into the P-Quad's plant preservation enclosure. These are reflected in table 11 on the previous page.

3.15.6 Piping for the Pump

Lastly to close the choices needing to be made on the water storage and pump system we had to select the type of piping that would best fit for being a mode of transportation of the water from the storage tank into the plant's soil. One of the first types of piping considered was glass for its lightweight and relative cheapness compared to some other materials, since however this was aimed to be a portable system there was concerns that the constant travel and luggage of the P-Quad would strain the glass systems to the point of shattering which in addition to causing a spillage of water would leave glass shards that would potentially damage the system and possible the user so this was quickly discarded. A promising type of pipe material was copper for a multitude of reasons. Copper is highly durable and not prone to leaks or drips. Additionally, copper doesn't leave any traces of pollution in the water it transports unlike some types of metal pipes (galvanized steel would often taint water slowly during its transportation) ensuring that the water from the storage tank to the plant would remain clean. Copper is a very sturdy metal in multiple regards being able to assuredly withstand any force that would be placed on the P-Quad system externally baring a catastrophically destructive force on the whole system as well as being highly heat tolerant both of the liquid being transported and external heat applied. However, two major issues arose from with the copper piping the first being copper piping is very expensive relative to the amount of material used. A 1ft segment of copper piping could cost anywhere from the $4-$8 depending on the thickness of the wall size of the pipe making it a fairly costly solution. An additional but massive issue is that copper being a metal is not easily malleable without the proper tools to prevent damage to the material and since the P-Quad would require an obtuse path to get from the water storage tank to where the water needs to be delivered forming this path out of copper tubing could prove problematic. This brought us to our next and final choice for piping plastic-based pipes divided into two kinds cross-linked polyethylene known as PEX pipes and polyvinyl chloride pipes known as PVC pipes. Both are very comparable in terms of what they off which is an immunity to rust, degradation, and corrosion over long periods of time. They both also are able to carry high water pressures while being a highly affordable for the amount of material ordered and being easily malleable to fit through a wide array of different piping configurations. Of the two's downside PVC is both unable to transport liquids at hot temperatures and splits if the liquid being transported inside freezes at any point. While PEX's greatest concern is the fact it is still unknown for certain if it pollutes the liquid, it holds during transportation as well as being slightly more expensive than PVC. After looking over these factors we concluded the standard PVC pipe would be the most suited for the transportation of the water from tank to the plant as the use case for this since the water would be kept between freezing and boiling points and if these conditions were put on the P-Quad otherwise the system would fail in a multitude of ways. Finally, the flexibility of the PVC piping would allow us to readily deliver the water in an efficient way without compromising the integrity of the pipe system. (54)

3.16 Wi-Fi Module

One of the secondary goals of P-QUAD's design is the mobile application that allows for users to save their templates for P-QUAD's environment variables and monitoring settings. In order to control the P-QUAD device, the device must be compatible with some sort of wireless communication. There are essentially two options, we choose a microcontroller that has a Wi-Fi module incorporated or we find an external wi-fi module that has embedded communications with a separate microcontroller. In this section, the Wi-Fi modules that are possible candidates for P-QUAD's system are discussed.

3.16.1 ESP8266 Wi-Fi Module

One of the best options, in terms of capabilities and cost, is the ESP8266. This specific module is highly popular amongst hobbyists and professionals so there are plenty of resources for us to reference during the implementation portion of our design project. For this Wi-Fi module, there are three operating modes: station mode, soft access mode, and station and soft access mode. During station mode, the module connects directly to the wireless access point and then uses the connection from the access point to connect directly to the internet. Soft Access mode still connects to the wireless access point just like in station mode, but it allows other devices to connect to it so that now devices that are connected to the module can also access the internet. For P-QUAD's system, we used solely station mode. We only need the P-QUAD device to be connected to the internet and not any other devices. The ESP8266 has a frequency range of 2.4-2.5 GHz, which is standard amongst Wi-Fi devices. The communication interfaces supported by the ESP8266 Wi-Fi module include: serial peripheral interface (SPI), inter-integrated circuit(I2C), and universal asynchronous receiver-transmitter (UART). Since the ESP8266 uses these three different communication interfaces it allows us to be more flexible in which what ports we have available to connect the Wi-Fi module to our microcontroller.

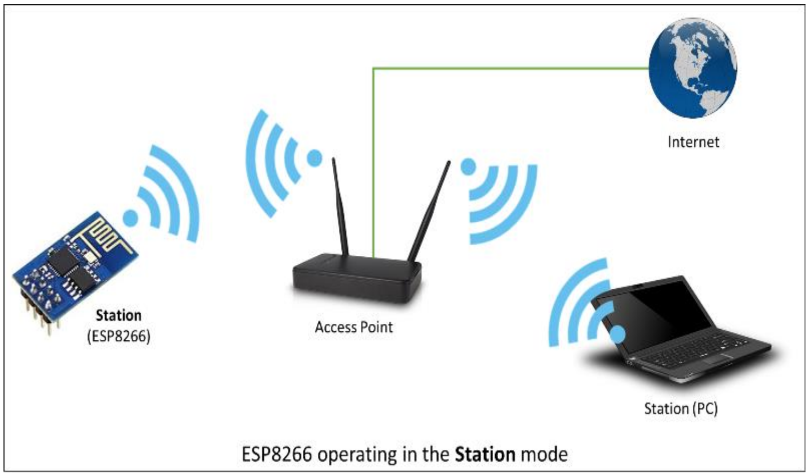


Figure 11: ESP8266 Operation Example in Station Mode

For P-QUAD's system, we used solely station mode shown above in figure 11. We only need the P-QUAD device to be connected to the internet and not any other devices. The ESP8266 has a frequency range of 2.4-2.5 GHz, which is standard amongst Wi-Fi devices. The communication interfaces supported by the ESP8266 Wi-Fi module include: serial peripheral interface (SPI), inter-integrated circuit(I2C), and universal asynchronous receiver-transmitter (UART). Since the ESP8266 uses these three different communication interfaces it allows us to be more flexible in which what ports we have available to connect the Wi-Fi module to our microcontroller.

3.16.2 AXM23001

The AXM23001 is a full SoC (System on a Chip), with number of different peripherals and modules that come integrated onto the system. The AXM23001 was an expansion of its original AXM22001 modules portfolio for IoT (Internet of Things) applications. The AXM23001 has very high-performance components, including ARM Cortex-M3 MCU with 2MB Flash memory, 2.5MB Data memory, SSL hardware accelerator and a full suite of peripheral interfaces including UART, SPI, I2C, PWMs, GPIOs and ADCs. In addition to standalone application, the AXM23001 has full TCP/IP protocol suite running on-chip and various serial host protocols such as high-speed UART and high-speed SPI to offload TCP/IP and WLAN protocol processing loading for system CPU in an embedded system. One of the benefits of AXM23001 is the built-in Google Firebase API support for cloud connectivity, so in the even that we choose Firebase as our database for P-QUAD we have libraries that are already supported by the database. That way we only have to learn how to use the API calls rather than writing them ourselves. (73)

3.16.3 PAN9310/9320

The PAN9310/9320 is Panasonics fully embedded Wi-Fi module that implements a fully integrated stack and API that minimizes firmware development. The module also includes a security suite. It has a standalone 2.4GHz Wi-Fi module that supports the IEEE 802.11 b/g/n standards that are designed for applications with high throughput data. The module is a power-efficient solution for WLAN applications. It combines high performance CPU, high sensitivity wireless radio, baseband processor, medium access controller, encryption unit, boot ROM with patching capability, internal SRAM, and in-system programmable flash memory. (72)

Table 12: Comparison of Wi-Fi Modules

Embedded Wi-Fi Module	ESP8266	AXM23001	PAN9310/9320
Full SoC	Yes	Yes	Yes
Communication Interfaces	UART, SPI, I2C	UART, SPI, I2C	UART, SPI, I2C
Multiple Access Modes	Yes	No	Yes
Wi-Fi Security	WPA/WPA2	WPA/WPA2	WPA2/TLS/SSL
Network Protocols	TCP/UDP/HTTP	TCP/IP	TCP/IP

3.16.4 Wi-Fi Module Selection

P-QUAD's system only needs to interact with the internet when the user is prompted to select the template for the environment variables for transportation. The P-QUAD system then fetches data from the database to be displayed on the system display. When choosing the module for P-QUAD's system, we really want to strive for simplicity due to the nature of the system. Therefore, we don't plan on using a microcontroller that has a built in Wi-fi module. We are choosing to use a Wi-fi module that is an extended peripheral of a different MCU and the options we considered are tabularized in the previous page table 12. For that reason, we are going to use the ESP8266 as our access point to the available Wi-Fi network.

3.17 Filtration System

An addition we considered to the P-QUAD system was a form of air filtration in order to make sure that the air the plant was getting oxygen from was clean and free of micro-organisms. Before we could dive into our plans to implement a filtration system into our project, we needed to understand how they work. Essentially an air filter is a device that allows air to pass in through some sort of fan system or natural ventilation and during this process it goes through the filter which is designed to trap small particles such as hair or dust (or in some case microscopic like bacteria and airborne viruses) and have the remained of the air come out with the large majority of these things removed. The level of what is filtered from the air is determined by the type of material that the filter is constructed from.

The first and most common type of filter is made from fiberglass. Most commonly found due to its economically efficient production cost and relative sturdiness to price range fiberglass is the most common solution for trapping larger contaminants with ease and reliability. However, smaller microscopic contaminants are often be able to slip past fiberglass making it slightly less reliable than some alternatives. Additionally, fiberglass requires more filter media space than some of its counterparts because it requires the material to be meshed across itself similar to a basket weave, unfortunately this also can negatively impact airflow to the desired location as it leads to a higher pressure drop. (61)

The next most common type of filter is activated carbon which is created from specially processed charcoal. By its nature carbon is a highly porous material which would make it desirable for a filter after it has been treated by various different cleansing processes (different levels can be obtained by different methods such as heat, chemical, and other various methods). Once this process has been applied the resulting carbon makes an extremely tiny filter able to easily filter out microscopic particles that could slip through other types of filter. These qualities and other natural ones about the element make carbon filters one of the most reliable and durable filters in the market today. This however comes at the tradeoff that they are costlier than some of the other alternative filter material per unit area. (61)

The final type of filter are artificial plastic filters constructed typically from polypropylene. These types of filters are generally referred to as synthetic filters possess several unique benefits. The first is by their design they are environmentally friendly as they can be recycled once they have reached their maximum usage and can in turn be used to create other synthetic filters. They also have the benefit of temporary reusability as synthetic filters are washable cutting down on both cost and waste generated by them due to this feature. While they do have these strengths and are also priced within a range slightly higher than fiberglass making them highly affordable, they do have the disadvantage of having less filter quality than the other two commonplace materials. While they are typically secure enough to be used to building air filters, they struggle to filter things down to the lower micro levels efficiently. This ultimately led us to the consideration of fiberglass HEPA filters as our primary choice for a filtration medium providing a good tradeoff between price and performance for the project. (61)

However, upon research the methods to provide airflow into the system we began to run into some problems bigger than the filter choice. The most noticeable of these problems was the scale at which the average air filtration system was intended for. Our research of the market showed that there were primarily three classifications of air filtration those intended for entire buildings, those intended for a single room, and those intended for green houses. The first class were obviously outside of our desired range as they encompass the filtration of the entire building making them wholly unsuited to the scope of this project. The last class and most promising sounding were the green house type which were specifically designed for the filtration of air for plant life. However much like the first class these filters were designed with the filtration of the entire green house in mind and not just single elements of it so despite its specialized nature geared towards plants this path ultimately had to be scrapped as well due to its design being aimed at larger scale targets.

The smallest commonly available market solutions and most likely to fit our paradigm are filters that target single rooms in a building sometimes referred to as standalone filters.

While this at first sounds unrelated it was decided that for the sake of the project, we decided that through abstraction we could classify the P-Quad holding chamber as a type of room and decided to see if there were any filters in this class that could potentially satisfy our project needs. While this one had more promising leads at first problems still presented itself with trying to incorporate this into the project. The first major issue was that even the smallest air filtration system we could find commercially was around the size of the proposed P-Quad’s meaning that containing the system inside the P-Quad would be an impossible to meet design constraint. Next the control of these small-scale systems was done typically via a phone application or an on-device interface menu. We felt that either of these control methods would prove too tedious and complex trying to integrate in the P-Quad’s existing control scheme and could lead to unnecessary complications. Lastly all the systems shown required their own 120V/60Hz power supply which would mean that both the filter and the P-Quad (when not in portable mode) would have to be plugged in and during portable mode the filter would not even be usable. Lastly the projected cost of the filtration system was around $80-$100 making it an extremely expensive component of the P-Quad’s fabrication. After an analysis of the facts we decided that the filtration system would ultimately have to be cut from the final scope of the project for these reasons. The proposed inclusion of the filtration would violate too many of the design objectives and constraints of the P-Quad such as its portability as lugging the P-Quad and the filter around would be very cumbersome and its low energy consumption due to the fact that there was no way to power the filtration systems without a 120V source. We briefly considered constructing our own air filtration system, but due to the lack of mechanical experience and that this project is aimed to show our design abilities on the hardware and software aspect this idea was dropped, and it was decided that the P-Quad would not include a filtration system.

4 Standards and Design Constraints

A key factor to the success of any project in its goals and eventually in the industry and market is to view and analyze and implement solutions based around the standards relevant to the project. Standards are an important part of any product as they provide a set of guidelines and deliverables that the project should aim. By observing and implementing around standards we can ensure that the product we create had a strongly defined background for its usage that users can look to be informed about the potential limits or uses cases of the project.

4.1 Standards

To protect implemented electronic devices, specifically the PCB, it is of best interest to consult the expertise of professional engineers and scientists who have performed countless experiments to create/achieve an accepted standard. The United Kingdom based & internationally recognizable International Electrotechnical Commission (IEC) creates and publishes standards for all electricity related technologies to help synchronize, protect, and globalize current or future electrotechnology.

4.1.1 Standards of Electricity

Intro to and standards of enclosure contacting electrical equipment:

Electricity and water do not mix well, and both are used in P-Quad. Above all, safety is the number one priority of this product. Among many reasons to develop a safe product one of those reasons is if the consumer can't operate the product safely then it failed in the market and leaves us open to numerous legal actions. The instructions to maintain the product are provided to the consumers. When cleaning P-Quad from the outside or when cleaning the inside of the pod that house the specimen, make sure that the power cord is unplugged from the wall. To protect the electrical wires from accidentally creating a harmful environment all the wires are installed inside the wooden body of the pod and are separated from the inside of the pod. To further protect the consumer and the product a relay system is implemented to prevent power surges. A power surge is a high amount of voltage that goes through the wiring for a short period of time basically a surge is a transient wave of voltage, current, or power through a system it is providing electricity to. The relay is implemented to protect the rest of the system from this overheating or short circuiting whenever there is a surge.

To protect potential hazard and the electronic devices, specifically the PCB, it is of best interest to consult the expertise of professional engineers and scientists who have performed countless experiments to create/achieve an accepted standard. The United Kingdom based & internationally recognizable International Electrotechnical Commission (IEC) creates and publishes standards for all electricity related technologies to help synchronize, protect, and globalize current or future electrotechnology.

The P-Quad model is designed with multiple enclosure layers containing electrical equipment. IEC standard publications help in preventing safety hazards, equipment mis-use and international compatibility. According to IEC 60529. By definition, an enclosure is "a part providing protection of equipment against certain external influences and in any direction protection against direct contact." (63)

Table 13: IEC Degrees of Access to Hazardous Parts

1st Characteristic Numeral	With Respect to Persons	With Respect to Solid Foreign Object
0	Non-Protected	Non-Protected
1	Protected against access to hazardous parts with the back of a hand	Protected against solid foreign objects ≥ 50 mm diameter
2	Protected against access to hazardous parts with a finger	Protected against solid foreign objects ≥ 12.5 mm diameter
3	Protected against access to hazardous parts with a tool	Protected against solid foreign objects ≥ 2.5 mm diameter
4	Protected against access to hazardous parts with a wire	Protected against solid foreign objects ≥ 1.0 mm diameter
5	Protected against access to hazardous parts with a wire	Dust Protected (Dust shall not penetrate in quantity to interfere with satisfactory operation of the apparatus or to impair safety)
6	Protected against access to hazardous parts with a wire	Dust-tight (No ingress of dust)

International Protection Codes shown above, in table 13, (of the form IPxx) are degrees of protection to help reduce or prevent electrical interferences. IP codes indicate various levels of protection depending on numeral selections, the larger the numeral is the more protection the device potentially provides against relatively foreign matter. First and second numerals are determined from tables 2-1 & 2-2, respectively.

IEC 60529 Table 2-1 offers degrees of access to hazardous parts with respect to persons and solid foreign objects. The First Characteristic Numeral in the first column of Table 2-1 is the first number to be selected for the International Protection Code (IPxx). Selecting numeral “2” as the minimum attainable goal implies a human finger does not have access to a hazardous electrical part and is protected against relatively foreign solid objects of diameter no less or equal to 12mm.

The Second Characteristic Numeral of the IP code lies within IEC 60529 Table 2-2 shown here in our table 14 on the following page. The table provides eight degrees of protection against water ingress. Since one of the project’s main functions is providing the plant water and maintain moisture levels, it adheres to at least a level “2” and may increase to a level “4” depending on time, functionality and funding. Concluding from both tables, the IEC 60529 IP code for P-Quad is to bear a minimum protection level of IP22.

Table 14: IEC Degrees of Protection against water ingress

Second Characteristic Numeral	With Respect to the Harmful Ingress of Water
0	Non-Protected
1	Protected against vertically falling water drops
2	Protected against vertically falling water drops when enclosure tilted up to 15°
3	Protected against spraying water
4	Protected against splashing water
5	Protected against water jets
6	Protected against powerful water jets
7	Protected against the effects of temporary immersion in water
8	Protected against the effects of continuous immersion in water

4.2 PCB Standards

Founded in 1957 the Association Connecting Electronics industries a trade association that have the responsibility to standardize the assembly and production requirements of all the electronics equipment assemblies. There is an IPC standard associated with almost every step in designing a PCB board. Some of the standards that the company has created includes but are not limited to:

- Design Standards
- Printed Electronics
- Printed Boards
- Electronic Enclosures
- Embedded Technologies
- Assembly

All the PCBs made commercially are required to follow a set of standards, IPC-221B which lists the general requirements for component mounting and PCB design, to ensure reliability of the product. Considering that P-Quad is planned to be used globally, IPC standards are also recognized internationally and translated well by following IPC's set of rules. The PCB for this system can help achieve desired results and makes this product reputation and reliability an asset in the market. There are different classes of standards in the industry for the design and fabrication of a PCB, and they are divided in class 1,2 and class 3. Class 1 is mostly for the testing and verification of design for general electric products; whereas, class 2 is for electronic products that are used for communication and are dedicated parts. The last type of PCB is classed as 3, and these products are used for military or medical applications.

4.2.1 Class One

A little detail about each classes of PCB are in order to better understand when they are used for specific applications. A class one boards are considered of a limited life category or that they have a use that might be considered lower reliability or a low life expectancy. An example of this might be a flash light that might need to function when needed; however, there is no expectation for the user and even the market where the flash light might be functional even after years. This is not to say that there are no flashlights that won't last for a long time, but these products are much for expensive and in low demand so the regular flashlights or let's say a toaster still uses a class one PCB to keep the cost of mass production low and still deliver to the users.

4.2.2 Class Two

Class two are a standard defined by IPC A-610 for building electronics. For class two the expectancy of a reliable PCB is higher, these PCBs are to perform continuously and have extended life without interrupted service. In other words, reliability is expected but not critical, so they are dedicated service electronic end products like a laptop's motherboard.

4.2.3 Class Three

Class three are to be considered as highest performance class of printed circuit boards family, they are expected to endure high level of inspection and testing and must meet all the stringent standards set by IPC. As explained before class three PCBs are used in military, medical, and aerospace applications. Class three standards are called in to be followed when the failure of a printed circuit board might be the difference between life or death. In order to meet such stringent requirements, there is no doubt the cost of the PCB would be very high.

For P-Quad at the prototype level we had to decide between class one and class two. What we have discussed is to order our first prototype PCB as a class one type and after making sure that it is working properly with the rest of the system, we might get a class two PCB ordered before the final presentation and see the difference in the reliability of P-Quad improves. We understand that for when P-Quad is implemented in the real-world scale and when important and rare specimen are being transported the reliability and longevity of P-Quad is paramount and class three PCB type was used. However, for this project we saw how reliable we can make it with class two standards.

4.3 Power Supply Standards

In the technological world of the modern era, any new product that is develop deeps power supply for it to function; however, in order to use any power supply, it must meet the relevant safety standards that are defined and administered by a national or international agency with lab test. The main objective of having such harsh safety standards for power supplies used in electrical equipment is to guard against the possibility of fire or electric shock. Using a power supply that meets all the safety standards is paramount in P-Quad not only because it is ethical to keep the end users safe but also because this project has a higher chance of fire due to a close proximity to water. If the power supply that does not have the proper conductor or a insulator then there is a high chance of electric shock and injury to the user. In addition to the dangers of electricity and water, this is compounded by the fact the P-Quad is in transit and undergoes turbulence and disturbances, so it is imperative that we use a power supply that meets all the standards. One of the power supplies that was implemented in P-Quad is a battery power supply. And batteries could be a hazard if they are not carefully regulated when being transported or when under certain amount of pressure. For rechargeable batteries the dangers of overcharging and discharge can explode the batteries which can cause fire or even electric shock. Power supply that is being used in this protect with meet the North American standards as well as European standards. We make sure that the power supply cords and the AC/DC adapter also meet all the electrical power supply standards so not to cause fire when P-Quad is not in transit and when it is idle in someone's home or office or in a laboratory. We expect P-Quad to be used not only in The United States rather all over the world and below are some standards that are described in table 15 below few of the many standards implemented for the safe use of batteries and other power supplies.

Table 15: Power Supply Standards

Standard	Description	Details
IEC 60065	Safety of Audio, Video electronic apparatus	Intended to protect against electric shock, fire with respect to A/V equipment
IEC 62368-1	Information and communication technology equipment	Standards for computer or communication equipment in North America
IEC 60601-1	Safety of Medical electrical equipment	Covers surgical, monitoring and hospital devices. This standard has been revised multiple times.
IEC 61010-1	Control and laboratory equipment	Covers the power supply used in control and laboratory equipment.
UL 1310	Standards for Class 2 power units	

Switching from alternating current to direct current was a considerable challenge back in the day, but with today's booming technology in electronics it is easier to do so with low cost equipment. The introduction of AC was very important as it can be very efficient when transporting power over long distances without the need of having the inconveniency and inflated cost of transmission wires. For DC, its efficiency comes from providing 'direct' current which implies that the current is being supplied to a load or component at a consistent, almost constant, rate relative to time. Using DC to transmit power over a long distance would require tremendous amount of voltage. By introducing high voltage to the system, the transmission lines must be protected. To protect the transmission lines' wires, the wire has to be very thick and withstand the high voltage rating and heat generated. The disadvantages of creating such a heavy durable wire, can put up a hefty cost, loss of power relative to distance, and may even break due to the high gravitational pull coming from its weight. With all that said we would have to find a way to convert the AC power provided to us into DC without much loss in power, unless intended. Coming back to the subject of powering P-Quad, we would want to utilize both to get the desired end results.

The typical wall receptacle at home provides approximately 120 Volts (±10 Volts) at 60 Hertz, which indicates that the current alternates 60 times back and forth from the nearest power/sub-station. Having a convenient power supply is great but we are going to run into problems when needing direct current instead.

The fundamental problem to be encountered is that almost all the equipment utilized in P-Quad was powered on DC power and run at a maximum of 5 Volts, yet the power that we get from wall receptacles is 120V AC. How can we come about to solving this problem? There are three solutions to this widespread problem, the first is to use a stepdown transformer to reduce the voltage rating coming from the wall. The second solution is to use a rectifier after stepping down the voltage. The third solution is to use a voltage regulator to keep our components running at a consistent voltage rating.

The first step is to stepdown the wall input voltage using a stepdown transformer. Using a transformer to step down the voltage from approximately 110 volts to around 5 volts is what we need to power the components across the P-Quad project. Reducing the voltage by that much can allow a lot of our low power equipment to run, but the input power is not always ideal. To solve that issue, we would need a capacitor in parallel with the load, as well as a voltage regulator. More is discussed upon further below when talking about the voltage across the load. However, for now, the upcoming stage after stepping down the voltage is passing through the pairs of diodes that performs the AC -> DC transformation.

A rectifier is a circuit that runs on AC power and is built in a certain way to transform the input alternating current (AC) into direct current (DC), prior to reaching the load. There are two types of rectifiers, a half wave rectifier and a full wave rectifier. While they both output direct current, the full wave rectifier is considered to be more efficient due to its utilization of the negative sinusoidal input as well, creating a consistently uninterrupted power supply. On the other hand, the half wave rectifier ignores the negative sinusoidal input, causing a larger gap between each positive peak, hence decreasing the efficiency. To achieve a full wave rectifier, we build the circuit with an AC power supply, a DC power supply, diodes, a resistor, a capacitor and the desired load.

For us to achieve the transformation from alternating to direct current, the critical component to achieve that is a diode. One of the diode's main functionality is helping re-direct current in a certain desired direction. In our case, we would need four diodes to achieve the current transformation. Having two pairs of series diodes, each conduct the current for every half cycle provided by the AC input. Since the AC input's charges are alternating back and forth, the diodes pairs allow the charges to go in one direction only, whether it be in the positive half or negative half of the cycle. Doing so allows constant positive cycle charges to flow through the load as direct current (DC).

After the full wave rectifier is built, we are going to run into the problem of our load receiving unstable or a rippled DC signal. To regulate the direct current going to the load, we would need a smoothing capacitor. The capacitor being placed parallel to the load allows the smoothing of the signal and help output a constant DC voltage value. The capacitor does that by charging and discharging whenever the AC input is transitioning from one peak to the other. It is a small gap between each of the peaks, but it is important for the load to be supplied with consistent voltage and current. So, whenever the transition is happening, the stored charge makes up for the needed voltage across the load, preventing any voltage inconsistencies. Ideally for each component, we would want to keep a consistent output voltage of whatever the component needs.

For the shared node between the load and capacitor, the input voltage across them, after passing through the diodes, must be at a consistent level. For the voltage not do deviate from our desired value we would need to add an extra component to the circuit. A voltage regulator is a built system to maintain a certain voltage rating at the output. This helps to prevent over or under-current from being supplied the load potentially causing problems such as deficiency at the output or overloading.

With the physics of the problem laid out for us to understand exactly what is needed to be achieved, we would want to integrate all of the above solutions to the problem previously discussed: How can we power low voltage equipment efficiently, at a consistent voltage rating, while using the convenient wall AC adapter? There are many answers to the that problem but only one of them would be our main approach.

The rectifier these days are overly saturated on the market, but for our purpose, that is not the only thing we need. For the selected Printed Circuit Board (PCB) that was integrated into P-Quad, it would require to be running at approximately 3.6 Volts, with the rest of the equipment running at a maximum of 5 Volts. To get the 3.6 Volts required to power the PCB, we would need a step-down transformer and rectifier combination that would supply at the very least 3.6 Volts with a maximum of 5 Volts. The device to do so should be able to transfer 110 Volts AC to 3.6-5 Volts DC. If we do get 5 Volts as our rectifier output, we can build a simple circuit to reduce it 3.6 Volts while using the 5 Volts in other need applications such as the LED lighting. Whether the components that require power but not necessarily through the PCB, or from the PCB itself, for us get the right power to provide to all the voltage varying components, we need to manually install voltage regulators at each load to assure that each component is getting the right amount of power at a stable rate.

Once all of this is achieved, we would require a lot of testing on the breadboard was made to make sure the voltage is steady and efficient. To do so, we would require both a digital multimeter and an oscilloscope to determine how much variations or noise there is and try to eliminate it as much as possible. Working with low cost equipment can be a barrier when trying to obtain close-to-ideal results, but that doesn't mean it cannot be done and integrated onto P-Quad. A lot of data from these experiments was used to determine factors like: How close to the ideal value can we get? Is this the best method to approach this problem? How did the system react when running on 9V battery?

One of the promises in our house of quality from figure 1 was making P-Quad a portable entity. To achieve that we need some form of back-up power to keep our system running in case of a black-out or simply for transport. This is extremely important as this determined the survivability of the plant, or plants, across a potentially significant period of time. Our determined optimal method for a back-up power would be a 9 Volt lithium battery. We specifically selected our battery at a rating of 9 Volts because first it: (1) Lasts longer, (2) can be easily replaced, (3) have enough voltage to be able to supply our components efficiently, and (4) it is portable and can be protected. In the next part, we discuss how the battery back-up was implemented.

Implementing a battery as a back-up system is a challenging but direct one. We have to make sure the battery is always supplying the correct amount of voltage to the respective components. Since our battery is at 9 Volts and our system runs at voltages between 3.7 Volts and 5 Volts, we would first need to reduce the power dissipated by the battery. To do that we would need to do reduce voltage one of two ways: (1) a voltage divider circuit or (2) using resistors in series. Whichever of the two methods we choose to approach it with, we would end up selecting the one that offers us the perfect balance between size and efficiency.

When we first decided to make P-Quad portable, we decided we need some form of power to provide to the circuit when disconnected from the wall receptacle. Implementing a battery as our back-up power reserve was thought to be, relatively, not difficult, especially at this point in our academic career. Even when some might argue it is, however, there are a lot of intricate details that have to be addressed or it would not work efficiently.

Primarily, we discuss how the battery was inserted to the circuit in order to work as back-up for power. A wire attached to our 9 Volts battery extend from the main circuit, sharing the same node with the load at the positive end, and going to the common ground at the negative end. Between the positive end of the battery and the node connecting to the rectifier circuit, there is a diode. The anode of the diode is attached to the positive end of the battery, while the cathode side would continue up towards the end of the node. The purpose of this diode is implicit is the brain behind the battery coming into play when the load is not receiving any power. Since we are using a rectifier as our input to the main load, our ideal diode pairs we discussed earlier can act as a barrier/open circuit against reverse current. We added an extra diode needed after the rectifying diode pairs for extra protection, but that was decided upon after our testing phase.

Before going into the topic of how the above setup came into functionality, one more concern to address is the type of diode used for one, potentially two, extra diode additives. I would like to start by comparing two widely used diodes, ideal (or rectifier or PN) diodes and Schottky diodes. They both have different but similar circuitry symbols, but they are diodes after all, so they should be performing the same essential task, right? The answer to that is yes, but not quite.

The ideal diode and Schottky diode are similar in the sense that, they conduct current in one direction (positive terminal in, negative terminal out), they result in voltage drops when current is passing through them, and they only allow the positive part of an input signal to pass through. The next part of the comparison has arbitrary, but realistic, values to better communicate the differences. These values are updated accordingly after testing has been made to determine the best required ratings. There are minor but important variations between the diodes, the first being the amount of voltage drop for each. The ideal diode tends to have a higher voltage drop of about 0.7 Volts while the Schottky diode has a voltage drop of about 0.3 Volts. That is a small but relatively significant difference, especially when our highest voltage operating voltage, so far, is at 5 Volts. Another major difference directly related to what we want to achieve, is the response time. Schottky diodes are dominant for the sake of having fast switching response time, which allows them to be the top diode selection when working with back-up power. All these attributes of Schottky diodes seem to be the perfect component for our battery back-up, is there no downside?

Getting back to using ideal in its dictionary-defined meaning, a diode does not really act as an open circuit as that is in its ideal form. In applied physics, there is no such thing as ideal because there are a lot of un-controlled variables that go into any application. In engineering, we tend to go with estimates, either for analyzation or simply for simplifications. Diodes are taught to us as being able to block the current from back flow completely, as in current $i = 0$, but that is not the generally the case. The main reason this is being mentioned is that pn (ideal) diodes are better in performing the open-circuit task because of its high voltage drop, however, Schottky diodes can allow a relatively significant amount of current back to the source. This may add more voltage to the circuit and cause serious issues in terms of functionality and components' protection. Even though the amperage was very low, it can compromise the ability of the battery to fully take over as the input to the circuit during automatic switching.

When it comes to switching the source to our 9 Volts battery, the Schottky diode proves to be king as the advantages do outweigh the disadvantages, compared to ideal diodes, However, for us to implement them, we have to be extremely cautious of the factors mentioned above, primarily the backflow of current.

After selecting Schottky diodes as our preferred diode for the task of input power switching, we can now discuss how the P-Quad automatic switching system operates under all conditions discussed above. Current flows at its normal state from the alternating current source input through the rectifier circuit to its final destination at the load. The owner then decides he/she wants to transport the plant enclosed inside of P-Quad, so the owner disconnects the power from the wall and goes on his/her way. When the wall plug is pulled, the first Schottky diode, located before the battery wire attachment and after the full-wave rectifying diodes, is not receiving current into its positive side therefore it opens disconnecting the back side of the circuit leaving only the load ahead. Since the higher voltage of the input is disconnected and creates a bias with the battery, the battery then conducts direct current through another Schottky diode that is in series, resulting in a voltage drop, and does its task of supplying power to the load. The first Schottky diode prevents any current from dropping back to the temporarily un-used part of the circuit. Once power is re-supplied by the AC source, the Schottky diode that is in series with the battery prevents the battery from being active, restoring normal circuitry conditions.

4.4 Legal Standards

For any new system that is developed by engineers, the legal standards are always enforced. It is to be noted that so one that is under a certain age should not be allowed to operate P-Quad when it is being used for transit.

4.5 Data Format Standards

When it comes to the transmission of data objects there are several common formats for the serialization of data in order to solve the issue of how to transmit data from one platform to the other while still maintaining data integrity. Below I compared the most popular standards for human legible data serialization languages.

4.5.1 JSON

The most popular choice for a vast majority of applications JSON (JavaScript Object Notation) is a lightweight format designed for data interchanging specifically in the case of asynchronous browser-server communication. It was originally designed for JavaScript, but today many languages and platforms include support for both the generation and parsing of JSON object. The design of JSON is based around transmitting data objects consisting of attribute-value pairs and/or array data types consisting of the atomic types of numbers (it should be noted JSON doesn't differentiate between integer or floating point), string, Boolean, array (an ordered list of values), object (an unordered collection of name-value pairs typically represented via strings), and null. One of the largest draws of JSON is the code used is considered highly readable by humans regardless of technical background and so it is easily taught and thus making it highly modular. In addition to this fact due to the popularity of JSON many IDEs and web service API's offer preset configuration for working with JSON format. (34) (35)

4.5.2 BSON

Very similar to JSON is BSON which is an abbreviation for binary JSON. BSON was made to address the issue of file size storage with JSON. Values stored in a BSON document are kept in their binary formatting which leads to not only smaller resulting file sizes but faster memory manipulation of data for operations such as traversal or insertion. The binary storage format also proves useful for exchanging of more complex data types such as images or attachments. BSON supports more data types than JSON as well with the core JSON types and the following additional types double (defined by the IEE754), date (integer count of milliseconds a la the Unix era), byte arrays, BSON objects and arrays, JavaScript code, MD5 binary data, and regular expressions (also known as regexp these are a sequence of characters that defines a search pattern which is typically used by string searching algorithms to locate or find and replace operations on strings. Finally, it should be noted that BSON is the primary form of data representation for MongoDB which is claimed to provide MongoDB's speed and ease of traversal. (34) (36) (37)

4.5.3 YAML

The next language YAML (YAML ain't markup language undoubtedly a homage to GNU's recursive naming convention) was designed to be a language with truly human readable code creating a convention that requires little to no technical familiarity in order to understand how to read it. Its structure is organized via white spacing removing the need for both quotes and brackets in addition to this YMAL contains a syntax for relational data with & representing internal references with anchors and * representing aliases. YAML is especially suited for files such as configuration, log, document headers, or debug dumps files due to the ease of viewing and editing of the files data structures.

YAML's data types include a very comprehensive list of independent types with two major classifications scalars which include nulls, decimal integers, hexadecimal integers, octal integers, fixed and exponential floating points, infinity, NaN (not a number), boolean, timestamps, and binary base64 encoding. For its other main classification of collections these include unordered set of key (!!map), ordered sequence of key (!!omap), ordered sequence of key (!!pairs), unordered set of non-equal values (!!set), and sequence of arbitrary values (!!seq). Much like JSON, YAML has a large breadth of support in both languages and environments such as emacs. Language supports ranges from older imperative paradigms like C to object oriented paradigms like java and is still being integrated and supported by new languages such as Crystal, Dart, and Rust. (34) (38) (39)

4.5.4 MessagePack

MessagePack is another binary format serialization designed with the intent of replacing JSON while still being highly influenced by it. Compared to BSON MessagePack offers more compatibility with natural JSON. Continuing with these comparisons MessagePack actually has a smaller overhead than BSON with the ability to serialize smaller objects. MessagePack as a wide range of supported types such as nil, boolean, integer (up to 64 bit signed or unsigned), floating point (IEE754 single and double precision supported), UTF-8 strings, binary data (up to 2^{32}-1 bytes), arrays, associative arrays (maps), and ext which is an arbitrary data segment of an application defined format with the limitation of 2^{32}-1 bytes. MessagePack's limitations of the size of arrays and integers comes at the benefit that it becomes more compact and therefore more concise in access times and times for operations such as traversals or insertions. As with the previous ones for language support MessagePack offers a wide range of implementations including C, C#, C++, D, Erlang, Golang, Haskell, Java, JavaScript, Lua, Ocaml, Perl, PHP, Python, Ruby, Scala, Smalltalk, and Swift making it possibly almost as versatile as JSON. (34) (40)

4.5.5 XML

We would certainly be remiss if we didn't discuss the markup language that originally inspired all of these data serialization languages which is arguably the oldest still used markup language today XML (extensible markup language). XML is a markup language which defines a set of encoding rules and restrictions for documents that aims to be legible to both humans and machines with the main design goal to emphasize simplicity, generality, and usability across the entire internet. XML is a textual data format with Unicode support for multiple human language interpretations this along with its other features allow for it to represent arbitrary data structures typically found in web and application services. Before diving deeper, it should be explained what a markup language is in comparison to the previous language that dealt with data serialization a markup is a system for annotating documents in a way that is syntactically distinguishable from text typically via some notation like tags. Markups lack traditional atomic data types present in the other standards viewed above.

The following sections discuss XML's key terminologies and how they are used to construct the XML document. First everything in XML is expressed as a string of Unicode characters with very limited restrictions on what characters may appear. Some XML documents may include a declaration which describes some information about the XML document pertaining to version or encoding. In XML the keyword processor, often referred to as the XML parser, is used to analyze the markup and passes on the structured information onto another keyword, the application. Requirements are placed on the what an XML processor must and must not do, but since the application is external it is outside of the scope. Strings are divided into markup and content which use syntactic rules to differentiate between the two. Markups are encapsulated with < > or they begin with the character & and terminate with a ; , additionally whitespace before and after outermost elements is classed as markup. Anything inside of these markup notations is considered content. Tags are a markup construct that denote when an element starts and stops or if the element in question is empty. Elements are logical document components that are encapsulated with start and end tags with the characters in-between comprising the elements content. Finally, the attribute is a markup construct consisting of name-value pairs that exist within start or empty element tag that describe the content of the data itself. Due to the long life of XML in the field there exists support for XML in almost any API or IDE along with it being able to integrate with nearly any language due to the fact it is a way of annotating and processing a raw text file. (41)

4.6 Application Program Interfaces

Often inexplicably entwined with the issue of data serialization languages is the calls needed to access and utilize them these routines, tools, and protocols used to construct, and access software are packaged together in what is referred to as application program interfaces also known as APIs. These APIs help offer a standardized platform of communication for web and mobile services that might be in different environments, but still need to be classed as cross platform communicable. The most common of these APIs for general purpose and data transmission via web or mobile application is the Restful API colloquially known as REST API. The general overview of the REST API is that it uses existing HTTP protocols of GET, PUT, POST, and DELETE in order to manipulate data. In REST API a request was made to the resource's URI (uniform resource identifier) which then elicits a response payload in a format such as HTML, XML, JSON, or other various formats. The response can then confirm that alteration to the data has been made in the stored resource and then the response can provide hypertext links to other related resources or collections of resources.

REST makes leverage of both stateless protocols and standard operations in order to aim for fast performance, steady reliability, and the ability to scale by re-using components that can be managed as well as updated without affecting the system as whole even during operation. REST API follows the following architectural design properties: performance is component interactions can be a dominant factor in user-perceived performance and network efficiency, scalability allowing to support large component numbers and interactions amongst said components, simplicity via a uniform interface, visibility of communications via service agents of the components, modifiability of components to change and adapt as needs change and arise even during the applications execution, reliability in the resistance to failure at system level at the exchange of failures within components, connectors, or data, and finally portability of components allowing movement of program code along with the data. (52) (53)

4.7 Economic & Time Constraints

Part of being an engineer is to keep innovating upon a problem by offering more solutions, for example, P-Quad can hypothetically have cutting edge technology including structure, mechanisms, display and efficiency. Even though we can keep expanding our ideas or improving certain functionalities, there are real life constraints that are out of one’s control. In this segment, we specifically talk about economic and time constraints regarding how it affected P-Quad, figuring out the perfect balance and making the best out of what is provided to us.

Economically speaking as undergraduate students funding our own project, we have little to work with, whether it be the actual out of pocket budget or finding the right resources. With our preset budget estimation at $525, it is relatively significant for our P-Quad team, yet not very significant when building a project especially in somewhat uncharted territories. P-Quad must be designed as near to perfect as possible and any flaws or miscalculations could potentially be devastating to our budget, progress and/or success. It is vital to pay attention and re-review every detail once we finalize our research document, this narrowed down our scope and build the P-Quad with the limited time we have.

Speaking of time constraints, we are given a limited time window to operate, therefore we must maximize the efficiency and doing so means we have to search for the delicate balance. This project is a passion of ours as it is the collective knowledge of everything, we have learned over the years whether it be in academics or independent research. One succeeds the other, the more time there is, the more brainstorming and data mining that can be achieved. Working and communicating together helps maximize our time while decreasing inevitable clutter & confusion, hence improving efficiency.

To maximize efficiency and find the perfect balance between money and time, which we can agree are intertwining domains to some extent, we must be prepared and not take any of our time for granted. This minimized the pressure to produce, resulting in lower errors and miscalculations thus increasing our troubleshooting window.

4.8 Safety Constraints

Whenever there is an implementation of new technology, environmental safety and more importantly safety of human life must be paramount. P-Quad has a substantial amount of power and energy that is running through its system, include water pumps and high power 120 V AC there are many safety concerns that arise. This system was designed to be compressed as much as possible so there is more room for the specimen to be placed inside the pod; however, it was also designed in a manner to reduce the amount of time user has to open and check the condition of the plant thus reducing the possibility of any harmful shock. Another safety constraint is related to the weather. Even though the plant is protected by the pod inside P-Quad, the P-Quad itself might be subject to harsh weather if it is being place or transported. In the event of rain, the outside of P-Quad must be of a material that does not allow any sort of slippage, so a user does not experience any moisture outside of P-Quad leading to accidental slip and fall in such as event the company is liable and might face lawsuits and other complications that could be avoid by doing proper amount of research and design.

4.9 Presentation Constraints

P-Quad when implemented in its true size and magnitude should be big enough to fit in the back of an eighteen-wheeler. The true scope of P-Quad is to transport trees, plants, flowerbeds in a massive quantity from either one city to the other or even from one country to another. However, to take full advantage of P-Quad the real size that should be used it when implemented in a full-sized airliner. Transporting numerous plants and trees and protecting them from the elements during the transit is the main goal of P-Quad. However, we don’t have the funds to create P-Quad in its true form, hence, we are making a Mini P-Quad that is about 2 by 2 by 3 feet in dimensions. To compensate for the smaller size, we have appropriately sized sensors and off course we do not need a be relay but rather a smaller relay that is enough to protect the system from any over-currents or fault. In full scale the relays are of appropriate size that is fit to protect the much bigger system. The battery that we used for the presentation is a smaller 9V or 3.6-volt battery that is enough for 3 hours of use while in P-Quad is in transit mode. We have a small LCD screen that has some push buttons to give commands to P-Quad. Also, the LCD screen showed all the information and the data that has be saved in the memory for the user. However, when implemented at the full scale there were programable logic controls that are used injunction with supervisory control and data acquisition. The programable logic controls or PLC are programmed using ladder logic and performs the same functions that are done by our smaller scaled system. Our system is basically be exact representation of a much larger system.

4.10 Energy Constraints

While designing P-Quad one of the main goals was to make the process of monitoring and controlling the environment of the pod as efficient and accurate as possible. Powering the sensors, LCD display, the lights and many other things to achieve this goal requires energy. Although this framework demands energy to work, we as a group decided to use the most energy efficient parts to power P-Quad. We used 120V AC to provide P-Quad with power while it is not in transit. In addition to that we also implemented a rechargeable battery to provide P-Quad with power when it is in transit. For the full-scale P-Quad has to deal with P-Quad being used for transit from one country to another in that case we considered the voltage and frequency differences from 110Volts at 60 Hz to 220 volts at 50 Hz. We needed to have an adapter that can easily be plugged in that are compatible with the replay and provide balanced power to the system.

4.11 Ethics

Technological advances in the field of engineering to the field cyberwarfare has been accelerating at a fast pace. The ethical deliberation; however, over these advancements have been lagging. For this project the goal is to provide a safe environment not only for the plants but also not to cause any harm to animals or wildlife and this is something that the team as considered very thoroughly. In order achieve this we not spare any attention or scrutiny to every part of the project to insure this project is as safe and as ethical as possible.

P-Quad does not have hazardous materials like toxic products in the project or any other related parts. Many companies cut corners to reduce the cost of development however we insured the P-Quad holds up to the highest standards of safety and to function is the most extreme circumstances because the entire purpose of this project is, so customers can transport their unique and rare plant without the fear of failure of the device. To strengthen the customers, trust in this product backup feature are implemented to the project which adds further protection of the transported items. The Toxic Substances Control Act of 1976 code 15 USC 2601-2692 gives the United States Environmental Protection Agency the right to require from any company that made a product the reporting, record keeping and the testing requirements. This includes but is not limited to chemical substances or mixtures that are used in the design and production of the product.

The rechargeable and replaceable lithium battery that was used and recommended for the consumers conformed to The Mercury Containing and Rechargeable Battery Management Act. The reason this act was instated is to phase out the use of mercury in batteries and to move towards more regulated batteries that include more cadmium lead electrodes. Another important part of ethics in a project such as this is patent protection. To insure this project is unique and does not infringe on any other companies' idea, we have performed as extensive research to avoid infringement on any other existing patents that uses portable plant technology. And in case any part of a protected concepts or design that be used by our group are given applicable credit to the owners.

4.12 Environmental Standards

Our project is all about providing an environment that simulates the stimuli that a plant might need to survive for an extended period. This includes the light and atmospheric conditions that are simulated using our peripherals. The purpose of the next section is to identify any standards that might influence our design plan or possible constraints that arise from these standards.

Standard Guide for Use of Lighting in Laboratory Testing

Active Standard ASTM E1733

Designation: E1733—95 (Reapproved 2014)

This standard, ASTM E1733, covers several different topics all relating to how artificial lighting affects living organisms under test regardless of whether these organisms under test require light. This standard also discusses the attributes of light that are in compartments and the affect this light has on a growth of the organism or toxicity of a contaminant, or both. An example of this has been shown that toxicity of organic pollutants is dramatically enhanced when ultraviolet (UV) radiation from sunlight is present. This standard also explains how to simulate sunlight with respect to the visible light to UV radiation ratio using relatively inexpensive equipment. (71) According to this standard there are several different light sources that that provide a variety of different spectral regions. Mimicking sunlight through artificial light sources which is represented in table 16 on the next page. Radiation from the sun with wavelengths that are greater than 290nm reach the surface of the earth, but radiation below 290nm is absorbed by the numerous gases in the atmosphere and are therefore not a part of environmental concern. At the surface of the Earth there is a loose molar ratio of different light wavelengths and radiations. Although it is variable in different places and different times, the standard is visible: UV-A:UV-B is approximately 100:10:1; however, the ratio for UV-B radiation is extremely varying. UV-B radiation varies with the time of day, peaking at solar noon, and varies with the seasons, with highest levels near the summer solstice and lowest levels near the winter solstice. This ratio of different radiation levels is critical in creating a lighting fixture that imitates sunlight.

Table 16: Artificial Lighting Standard Comparison

Lamp Type	Spectral Regions	Fluence Rate (μmol m−2 s−1)	Lamp Cost (USD)	Fixture Cost (USD)
Fluorescent	Visible	20-400	5-20	10-30
	UV-A	1-50	10-40	10-30
	UV-B	1-50	10-40	10-30
	UV-C	1-50	10-40	10-30
	Visible+ UV-A	20-400	20-50	10-30
Short Arc				
Hg	UV-B, UV-A, visible	500-2000	150-1000	2000-6000
Xe	UV-B, UV-A, visible	500-2000	150-1000	2000-6000
Metal halide	UV-A, visible	300-1000	100	1000
Sodium vapor	visible	300-1000	100	1000
Microwave	UV-B, UV-A, visible	500-2000	2000	10000
Incandescent	visible	100-1000	5-100	10-1000

Simulated Solar Radiation (SSR)

In order to provide an environment that has the Simulated Solar Radiation necessary for photosynthesis to occur, we need to simulate a ratio of 100:10:1 of visible:UV-A:UV-B radiation. There are several ways to produce UV-A and UV-B radiation using fluorescent lamps. The most general case is through the combination of two cool-white fluorescent lamp (for the visible light), a 350-nm fluorescent lamp, and a 300-nm fluorescent lamp filtered by multiple layers of cheese cloth (to the UV-B level down). The light is then filtered through a film of cellulose diacetate (0.08mm) to remove extraneous UV-C radiation. The removal of UV-C is necessary as UV-C radiation is much more damaging to biological molecules. Using this combination of fluorescent lamps mimics the radiation and sunlight from the sun. If we wanted to be even more precise in this simulation of the use of incandescent bulbs can also be added to obtain a more accurate spectral balance in the blue and red regions of light. Although this provides a more precise imitation of the sun the UV-A and UV-B levels also need to be adjusted to due to the dilution of light from the addition of lamps. Incandescent lamps also produce much higher levels of infrared radiation so the use of a cooling system or infrared filter on the sample are needed in temperature prone environments. (71)

Standard Practice for Planning the Sampling of the Ambient Atmosphere

Active Standard ASTM D1357

Designation: D1357—95 (Reapproved 2011)

Since P-QUAD is able to hold any variety of plant that needs to be transported, there might be a need to test the toxicity of the air contained within the P-QUAD plant housing. The following standard discusses the standard practice used to sample the atmosphere. The analysis of the atmosphere is influenced by all the phenomena in which all factors except the sampling and analytical procedure are in not within the control of the investigator. The adequacy of the sample depends on the number of samples obtained, the number of sites sampled and the time that the sampling is carried out. The purpose of this large spread of sampling is to determine the amount of contaminant and its characteristics based on the sampling times and frequencies.

It is important to note that statistical methods are allowed to interpret the data that was collected during sampling. The trends that arise do to the relationships across variables of statistical significance. The validity of the results depended on the comprehensiveness of the analysis and the location of the contaminant measured. For example, if daily samples of a particulate matter are obtained only periodically then the geometric mean of the measured concentrations are representative of the median value (assuming data is normally distributed). The geometric mean level can be used to compare air quality of different sampling methods in which regular but intermittent observations of concentrations of particulate matter are conducted. Below is a list of basic principles to apply when conducting sampling methods. (69)

- The choice of sampling techniques and measurement methodology, the characteristics of observation sites, the number of sampling stations, the amount of data collected all depend on the objectives of the monitoring program.
- To cover all variable meteorological conditions that may greatly affect the air quality in an area, monitoring for lengthy periods of time may be necessary to meet the objective of the sampling.
- The topography, demography, and micrometeorology of the area as well as the contaminant measured must be brought into consideration of how many monitoring stations are a required in an area. A photograph and location of the sampling stations is desirable in describing the sampling station.
- The sites of the sampling program should be selected based on the influence of any local source that may cause local elevated concentrations of contaminant that are not representative of the region characterized by the data. Unless the sampling site is of special circumstances, this practice should be used, in general.
- Sites that are being used to determine the impact on air quality by a single, identifiable, source should be selected, if possible, so as to isolate the effect of the source being considered. When multiple sources of the contaminant in the area are being sampled, the sits should be strategically located such that the wind direction data obtained simultaneously near the sites. This provided evidence of the contributions of the individual sources. Multiple sampling sites operating simultaneously upwind and downwind from the source are also desirable.

Sampling Procedure and Siting Concepts

The choice of sampling procedure depends entirely on the contaminant to be measured. ASTM has published several methods for sampling the most common contaminants. Automatic instruments providing a continuous record of concentrations of the contaminant should be utilized whenever possible to save man power and increase efficiency.

The monitors must be in a position such that the air to be sampled accurately represents the air under investigation in complete area. Careful consideration should be shown to the sampling system. The standard dictates that a duct system is used to transport the air under sample to the monitor sampling the air. The duct should have as few abrupt enlargements as possible and minimal number of elbows. Any obstructions or bends may interact with the contaminants under investigation if there is not minimal contact. The material used for the duct should be chosen such that the type of material has little or no interaction with the contaminant under investigation. If there are potential involuntary variables that arise such as condensation or weather hazards, proper precautions should be taken such that the sampling procedure remains unaffected.

Plan of Sampling Procedure

The standard dictates that the procedure for sampling should be undertaken such that:

- A general informative survey of the area, including topography, all sources for contaminants, the height of emissions, traffic, land use area, and potential hazards.
- Preliminary meteorological analysis to identify wind direction frequencies, velocity of the wind, and height-temperature profiles.
- If there is a need for short-term exploratory sampling, then there should be an adequate number of temporary sampling stations to determine the sites in which the best results can be obtained. Using the data collected in previous stages, an estimate of levels of air quality over the area can be calculated. This model should serve as a guide in determining the number of monitors and in which sites they should be placed that maximize the potential amounts of contaminants to better grasp the overall quality of air. (69)

4.13 Quality Assurance

This standard is all about the quality of air in an area and must be held to the highest quality in terms of sampling and procedures. There are numerous quality assurance programs that include all the necessary activities to provide measurement data at the required precision and accuracy. The guidelines for quality assurance programs are given in Practice D3614. According to this standard, shown in table 17, the following elements must be included in the quality assurance program, if applicable:

Table 17: Quality Assurance Standards

Standards for Quality Assurance of Sampling Procedures
Sampling and analytical procedures should be explicitly stated, using a standardized method such as the ASTM where they are applicable and fitting to the objectives at hand.
Calibration must be done using standard methods. For long lasting programs, the number of calibration re-checks should be explicitly specified.
Data collection and recording procedures must be specified, to identify responsibility of reporting records and analysis of chart record recording. There should also be specifications for the upkeep of equipment, samples, and records to uphold the account and credibility of the program.
In the case where samples must be shipped from one location to another to be analyzed, the method in which the samples are transported must be recorded. Especially if the methods involve any sort of preservation prior to analysis, such as refrigeration or other storage conditions.
Methods of computation for the recorded data should be specified, including a portion indicating the validation of data procedures so to minimize the errors in computation or record keeping.
The use of independent, third-party audits of the entire procedural system are helpful in meeting a level of quality.
Interlaboratory tests are also a useful way of verifying the precision and accuracy of the methods used to acquire data. They also make it possible for the respective organizations to check the internal performance against other laboratories engaged in a similar activity or research.

Active Standard ASTM E2877

Designation: E2877—12

P-QUAD's monitoring involves the use of a digital contact thermometer. This standard describes general-purpose, digital contact thermometers that give the degrees in units of Celsius or Fahrenheit. The standard also describes the different classes of accuracy for such thermometers described. If our product were to go to market it would be important to know what our hardware's accuracy was in relation to a standardized benchmark. There are a variety of manufacturers and different selections of digital thermometers provided in the documentation of this standard. The derived SI units (Celsius) are considered standard. Fahrenheit may be substituted so long as all other stipulations are compliant with the standard.

Temperature Sensor	PRT w/ wire filament	PRT w/ Thin film filament	Thermistor	Thermocouple Type K
Temperature-Dependent Quality	Resistance of a platinum wire	Resistance of a platinum film	Resistance of a semiconductor	Emf difference between two dissimilar wires
Nominal Temperature Range	−200 to 650 °C (−328 to 1202 °F)	−70 to 300 °C (−94 to 572 °F)	−50 to 100 °C (−58 to 212 °F), typical	−200 to 1260 °C (−328 to 2300 °F)
Sensor Protection	Glass coating, glass or stainless steel sheath	Glass coating, glass or stainless steel sheath	Epoxy or glass bead, stainless steel sheath	2—bore ceramic tubing, fiberglass
Size (Diameter)	2 to 4 mm glass coated, 1.6 to 6.4 mm	2 to 4 mm glass coated, 1.6 to 6.4 mm	0.5 to 3 mm bead, 0.9 to 6.4 mm	0.06 to 2 mm measuring junction, 0.2 to 6.4 mm
Stability (against drift)	Good/Excellent depending on temperature range and design	Good/Excellent depending on temperature range and design	Excellent	Good, increases as temperature decreases
Vibration Tolerability	Limited to Very Good depending on design	Limited to Very Good depending on design	Very good	Excellent
Accuracy	2 leads: Fair 3 leads: Very Good 4 leads: Excellent	2 leads: Fair 3 leads: Very Good 4 leads: Excellent	Excellent	Good, increases as temperature approaches 0 °C

Table 18: Standard Guide for Digital Contact Thermometers

Types of Digital Thermometer Sensors

The previous page shows table 18 which is a tabular comparator for the thermometer sensors coming up in more details.

Platinum Resistance Thermometers (PRT) – The electrical resistance of a PRT's filament has a positive nearly linear relationship with temperature making it a good contestant for temperature reading. A PRT is composed of a platinum filament of fine wire or film help by some sort of insulating material. The sensor is connected to an instrument capable of supplying a current through conductive wires to the filament. By measuring the voltage across the PRT to determine the resistance using a relevant resistance/temperature equation. PRT calibration can be done with a nominal resistance-temperature relationship with a variable tolerance or a single sensor calibration with an expected level of uncertainty or error. Using a nominal relationship provides the reading device to be programmed with a single resistance-temperature relationship for a PRT family. The interchangeability tolerances are usually greater than 0.1 degrees Celsius and increase as we move away from freezing point. Alternatively, a sensor specific calibration may be used when there is not a nominal curve that exists or if the interchangeability tolerances do not support the accuracy needs. The calibration of a PRT with uncertainties of 0.01 degrees Celsius are possible depending on the range of temperatures and the PRT stability as well as the test measurement capability. Temperature range, vibration tolerability, and stability are key characteristics to consider when selecting a PRT for a specific accuracy class. General guidelines are summarized in the following table. (70)

Thermistor – A thermistor consists of a blend of metal oxides that create a semiconductor material. The electrical resistance of the semiconductor varies with temperature making it a viable competitor for use as a temperature sensor. There are two types of thermistors: one that increases resistance as temperature increases (known as Positive Temperature Coefficient or PTC) or one that decreases as temperature increases (known as Negative Temperature Coefficient or PTC). A large portion of thermistors used are NTC type. Thermistors have the advantages of a fast response time, high resolution, and low uncertainty for their specified ranges. Thermistors also have very good stability and vibration tolerance. To protect the semiconductor, thermistors are usually encapsulated in epoxy or sealed with a coating made of glass. In the case where the thermistor is external from the device doing the measurement, electrical leads that are electrically insulated from the environment and one another should be used to connect the thermistor to the measurement device. The calculation of resistance seen in the semiconductor is very similar to the calculation used for PRTs. A known current is applied to the leads of the semiconductor, and then measuring the voltage across the leads. Then using a specific resistance/temperature equation, the temperature can be calculated based on resistance.

Thermocouple – A thermocouple is two parallel dissimilar homogeneous metal wires, called thermoelements, typically of equal length joined physically and electrically at one end called the measuring junction. The opposite end is called the reference junction. When a change in temperature occurs between the measuring junction and reference junction, an electromotive force (emf) is created across both thermoelements where the temperature gradient originated. (70)

5 Hardware and Software Design Details

This section of the paper we address the specific details relating to the technical design details of the project in both the hardware and software capacity. For any project it is imperative to consider the design aspects from all major views before one begins any stage of a project's development cycle be it pre-production, prototyping, fabrication, or deployment which is why we reviewed step by step overview the design details of the P-Quad. This deal on the hardware side not only with the PCB and electrical components, but as well the mechanical and structural components that compromised the physical systems of the P-Quad. In addition, we addressed the entirety of the software portion ranging from code design plan to communication protocols used to relay the information as well the various methods of communication both internal with the hardware and external with the proposed application stack.

5.1 Hardware Design Details

The hardware design for P-Quad is very simple; however, when moving towards the inside of the systems design it gets arbitrarily more complex. The outside of P-Quad that houses the pod and all the electronic, was designed using SolidWorks. The outer most part of P-Quad we denoted as the “Shell”. The shell is 1.2 feet in length and width and three feet in height and was made in three different sections that comes together to make the entirety of the shell. The bottom most part is about inch and a half in length but still two by two in length and width. The bottom most part is hollow, and it houses all the electrical components that power the rest of the system. This section or the “Housing Base” holds the PCB, replaceable battery and the relay; furthermore, the Housing Base has a waterproofed hole that has power supply cord running through it to power the PCB thus, providing continuity of power to the rest of P-Quad. The best section that goes over the Housing base is a flat piece of “Cover” that has a matching latch like one on the House Base. This cover fits on top on the House Base and is able to open vertically with the help of the latch to reveal all the electrical components. The Cover is secured in place and id waterproofed when closed so no water or other materials fall inside the Housing Base causing any chances of fire or electrical shock and human injury.

The third section of the outer most part of P-Quad is what we denoted as the “Dome”. Dome is two by two by three feet tall (minus one and a half inch for the Housing Base) and it is going on top of the Housing Base and the Cover. There is a mechanism that fixes the Dome on top of the Cover, but the user is able to remove the Dome whenever that a need to access the electronic components. We are considering designing the Dome in a true dome shape with the top being oval; however, we discuss this in detail later. The door was implemented in the Dome section for an easy access to the Pod that house the plant. The door is either a solid part of with the potential of having a mesh door. The idea behind using a mesh door is to improve the airflow to the plant; however, this idea might be scrapped because when considering P-Quad is transported we have to protect the Pod which houses the specimen from the outside environment hence, we went with the solid door option; however, we kept the mesh idea in the back of our minds.

The second layer of the hardware design includes the inside of P-Quad, the “Pod” which houses the actual plant or specimen. The Pod has LED lights that provide the specimen with light that it requires for its proper growth and health. This is the layer that also houses the water pump that provided the specimen with moisture and with water to the soil.

Majority of sensors that send information about the status of the specimen is implemented in the Pod layer of P-Quad. One of the things that is very important in the Pod layer is the tray that is used as the place holder for the soil and the specimen. We understand that Pod needed to house not just one large plant but also the customer might want to monitor and control multiple of one kind of plant at the same time. To tackle this customer requirement, we have the trays that are in the Pod layer of P-Quad to be replaceable. Meaning we have multiple types of trays, from just one large hole in a tray that have enough space for one large pot with soil and plant inside, to a tray that have many smaller holes that house many smaller pots with soil and plants inside them. The trays themselves are completely replaceable to that the user can easily interchange the trays depending on what sized and what number of plants they are wanting to monitor and control during transit.

5.1.1 Initial Design Architectures

For the P-Quad structural design, it was accepted by all to use a 3-D Software for sketching and eventually creating a 3-D printed result. The top of the line software for generating 3-D assembly models and provided access by the University of Central Florida is Solidworks. The justifications behind choosing Solidworks are that one of the group members are familiar with it and it is the standard when it comes to engineering designs.

The structure itself consisted of four separated parts that are assembled together to create P-Quad's cumulative constitution. Starting from the base and up, an open box, a lid to cover the open box, a hollow cylinder on the center of the lid, and a dome cover on top of the cylinder. Each of these components is 3-D printed or laser cut and assembled once all our devices have been integrated to their respective designated locations.

Starting with the base, the structure is a 1.5' x 2' x 10' open box. This serves as P-Quad's foundation as well as a holster for the PCB and the soil. Looking more into this after discussing as a group, our priority is always safety, so having the soil and the PCB in the same location can cause significant moisture levels from the soil's evaporated water. This condition can short circuit components causing hardware failure. This issue is discussed further in detail under hardware constraints.

For the next assembly part, we needed something to close the top-open box, so we added a lid with dimensions of 1' in diameter and 2.5' tall to seal the box. We inserted a door hinge on one of the sides so that the base can be easily accessible by opening and closing at a minimum of 180^o. Running the wires through the lid up can be deemed a challenge so there is admittedly a design flaw here. The reason this design might completely change was expanded upon under hardware constraints.

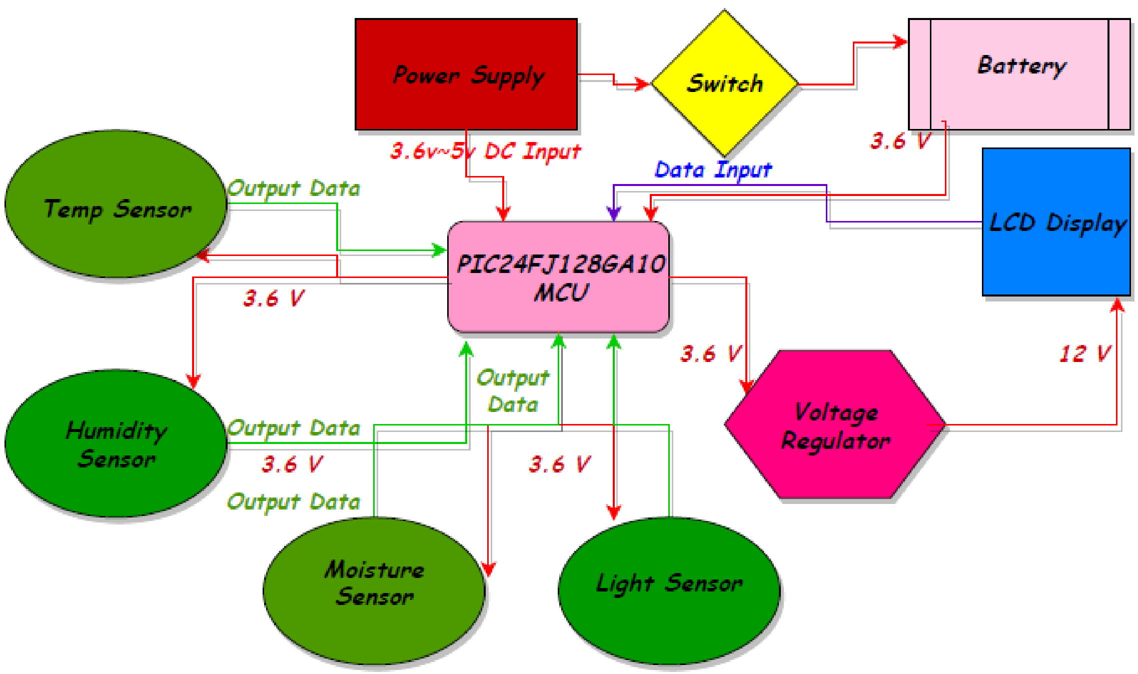
Our third part addition is the hollow cylinder located on top of the lid. The cylinder contains and provide a protective outer shell for the plant. In the initial design this seemed like a great idea, but we do want the plant to get some breathability from the atmosphere outside. To improve upon this, we thought to create a similar cylinder, but it was meshed halfway from the center and up, succeeded by the dome cover.

The dome shaped cover is used to seal the structure from the top. The reason it is detached from the cylinder, is to provide accessibility at any time to a component. To allow the dome detachment and provide flexibility when working, we needed to improve upon this design. For starters, we promised in our figure 1 that P-Quad is a locked structure. We obviously cannot have any of the components potentially getting loose, to achieve that, we thought about adding upon our design of the dome and cylinder to have matching extended lobes on opposite sides (four total) with a small enough hole so that they both can be pad locked together.

5.2 Block Diagram

Every part of a system can be represented in a set of components that can show with blocks. And these blocks can be rearranged in a manner to should the overview of a system. These are called block diagrams and they are a pivotal part of the engineering world. Block diagrams are very important in hardware design, software design and the electronics design. All of the blocks in a block diagram in figure 12 below are connected to each other via arrows that provide a relationship between these blocks. For the inception of our team's idea and making the first flowchart representing hardware, software flow we have constructed an overall system diagram that shows the power in red, the sensors in yellow, and all the software in blue.

Figure 12: Design Block Diagram



5.3 Bread Board Testing

To determine whether the breadboard is non-defective and giving expected results. Multiple circuits with components such as power generators, resistors, capacitors, inductors, diodes and op-amps shall be used in different breadboard slots, to determine there is no fault in the device. To do so we designed, solved, built and tested both simulated (SPICE) and practical (breadboard) circuit problems in order to determine functionality. Once that is achieved, the breadboard can then be used to hold the mock circuit before any soldering is involved below here in figure 13.

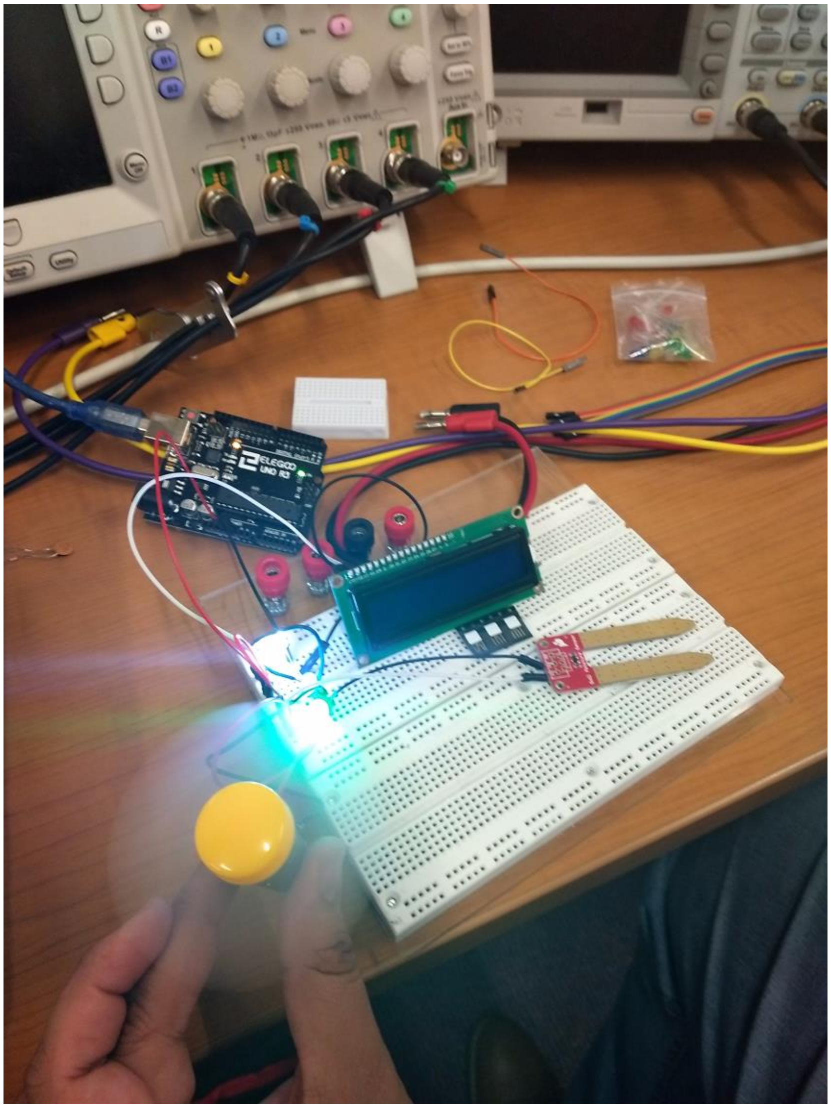


Figure 13: Breadboard Testing

5.4 Sensor Testing

To begin testing our P-Quad selected integrated sensors, it is of best practice to prepare several variables different in nature to compare and determine whether the calibration and readings are correct or correct to some extent. To amplify what is meant by different in nature, an example would be for the moisture sensors we would read moisture levels of both water in its natural state and a house or wild plant's soil. The sensors that are being used to record data for P-Quad are relatively economically friendly sensors, therefore it is expected for our output data to have deviated results. Depending on each sensor's ±%error, we can verify each sensors' functionality level and adjust.

5.4.1 Moisture Sensors

To test the functionality of our moisture sensor (SEN-13322) it was ran through several tests. The error rating on the selected sensor is ± 5%. The first element to be tested was tap/drinking water. This is an obvious first choice due to the reasoning that, since the definition of moisture is "water or other liquid diffused in a small quantity as vapor, within a solid, or condensed on a surface" hence, we should expect a capped rating from our sensor. Since it is to read the highest possible moisture level, the value should ideally be 100%, but our predicted output rating of the sensor shall read a minimum of 95% to be deemed functional. The second tested object would be something on the other end of a spectrum such as dry towel and a slightly damped towel. For the dry towel it is expected to read a value between 0-5% and the slightly damped towel at about 10-15%. Our final variable would be to check Orlando's outdoor humidity. The approach is, through a weather app, to figure out the humidity at our location. The sensor rating was within a ±5% deviation from the real value for it to be valid. The justification for such a large ± percentage is due to multiple affecting factors. The factors that were taken into consideration when outdoors is the height from the ground, whether any body of water exists nearby and whether it is under the exposure of direct sunlight. All these tests might seem excessive to test whether the sensor is functional, but they are necessary data to be gathered because the soil can be affected by some or all three factors simultaneously. After all the data has been gathered, we used everything we learned from our data analysis that affects moisture ratings and apply it to the testing of P-Quad.

5.4.2 Temperature and Humidity Sensor

For temperature and humidity sensor testing of the model SHT30-DIS-F2.5kS by Sensirion, we needed to run functionality tests to determine if it's working properly. For a start we can did same experiment previously mentioned with the moisture sensor. This allows more accurate data gathering regarding P-Quad being sometimes exposed to outdoor scenarios, which is not the environment it is intended to be in. Since the goal is substituting natural elements, especially sunlight, evaporation can be drastically affected and need to be accounted for. To know exactly how much light, we are to use in order to determine generated heat of the enclosure, we needed the light sensor's results.

5.4.3 Light Sensors

Light sensors serve one purpose and that is to detect the intensity of light at a certain period. To test this, we ran it through some tests first to see how it reacts under different light settings. The three tests included those three variables, a dark room, a "tactical" flashlight and the sun. For the dark room we expected 0% which was correct and for the flashlight at about 98% ± 2% which is around the range. With those results hold true, we can then measure, on a sunny day, how much light is emitted from the sun on a regular basis. We recorded each hour from sunrise to sunset across 2-3 days to get a realistic feel of how much light is emitted, at what strengths, and for how long. All our data was averaged on the hour of each day and determine the amount of light that is fed at the top of the hour. With that done, we implemented our analytical results onto P-Quad's light system.

5.5 Potential Hardware Issues

There are some potential hardware issues that we had to tackle in P-Quad. One of these issues is how to get the sensors in the Pod, which is the place holding the specimen. Sensors are an electrical part and the Pod has water and moisture which are pumped by water sprays for the plants and this can lead to water damage to the sensors. To resolve this issue, we have decided to make a small compartment in the Pod that houses some of the sensors to protect them from water damage. However, there is another issue which involves some of the sensors being surface mounted and requiring them to be in close proximity to the area where the sensors can function properly and relay data accurately. To resolve this, we have decided to move the PCB from the House Base to the small compartment in the Pod area so get the best data for the users. Another problem that we are facing is how to get the wiring from the sensors to the area of importance. In this case we have decided to drill holes in the Dome of P-Quad and run the wires from the PCB to the area of interest.

5.6 Software Design Details

5.6.1 Design

The following section discusses the high-level needs for P-QUAD’s software architecture for both the embedded software and the mobile application. This section does not discuss the implementation of the software systems.

5.6.2 Use Case

The following use case in figure 14 on the next page UML diagram highlights a high-level interaction between the user and P-QUAD’s software. Initially, the first user of P-QUAD is required to register an account on the P-QUAD database. This is it store information about different environment variables in the system. After registration is completed, the user logs into the system via the mobile application and be prompted to select whether they want to use a saved template or create a new one. If a saved option is selected the user is prompted if they want to edit the settings for P-QUAD’s environment variables.

5.6.3 High Level User Interaction Flow

The following diagram figure 15 also on the following page describes the high-level flow of a user interaction with P-QUAD. The initial stage requires the user to log into the application to connect their account to the P-QUAD device that they are interacting with. After the user has been connected to the desired device, they are prompted with the selection of the template for the environment variables within P-QUAD. There is option to sect a saved template or to create a new one. If the user selects the create a new template option, they enter the desired characteristics of the P-QUAD machine, and their template is saved. If the user selects a previously saved template, they are given the option to edit the settings for the P-QUAD template and then it is saved.

Figure 14: Use Case Diagram

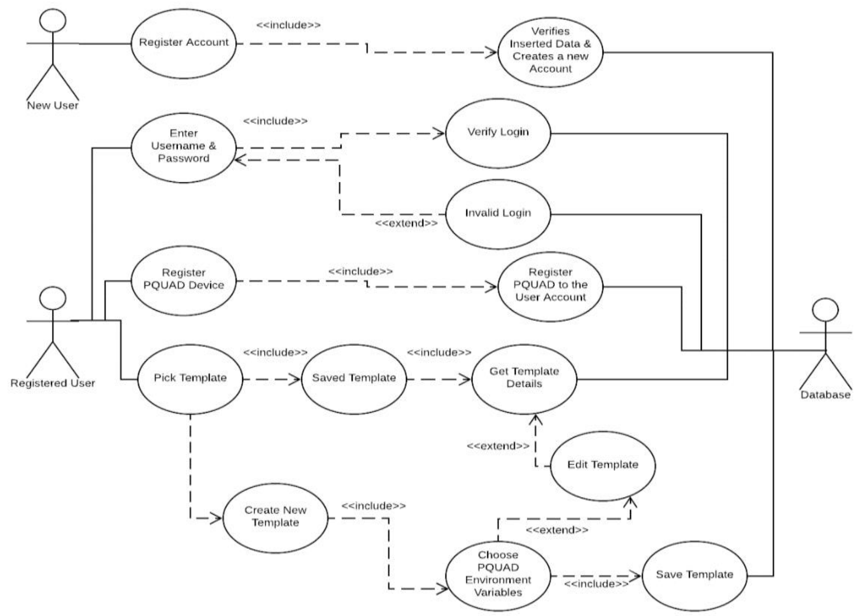
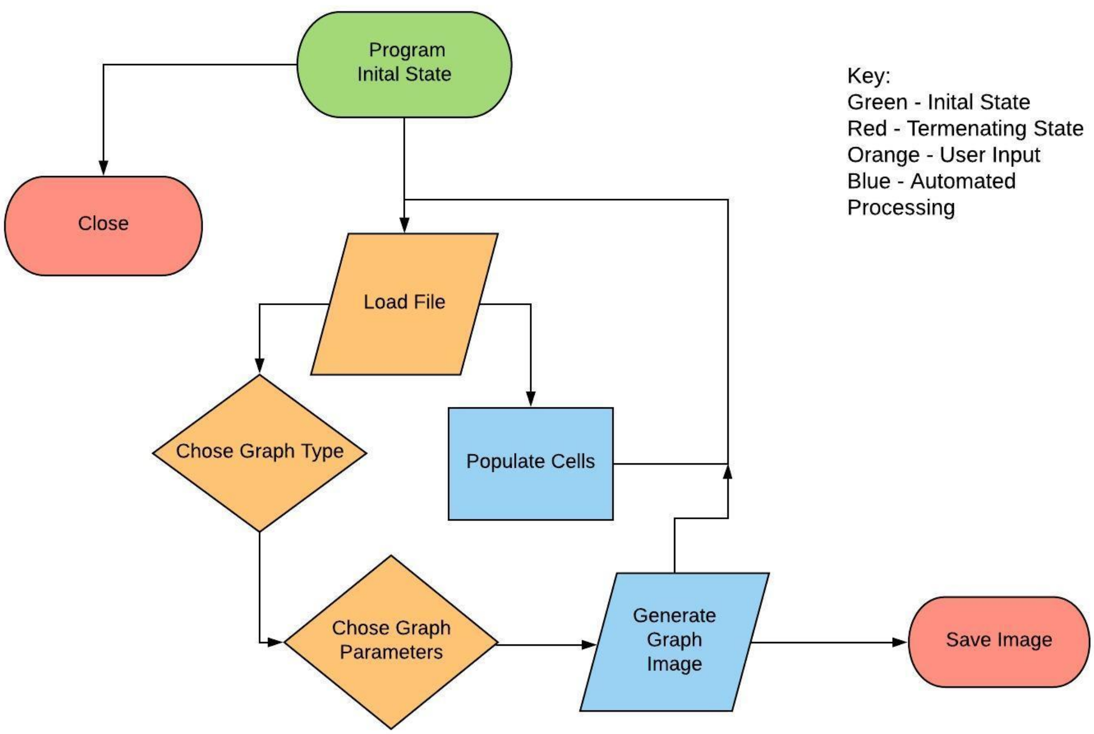


Figure 15: High Level User Interaction Flow Diagram



5.6.4 Low Level User Interaction Flow

The following figure 16 discusses the low-level software architecture flow for user interactions with P-QUAD. Initially, the P-QUAD device needed to initialize the environment variables and alert conditions. The user is prompted to select whether they would like to have conditions input manually or select from a saved template from the account that is linked to the P-QUAD device. After the template for the P-QUAD environment has been fetched, P-QUAD initialized the device's settings based on the template the user has selected.

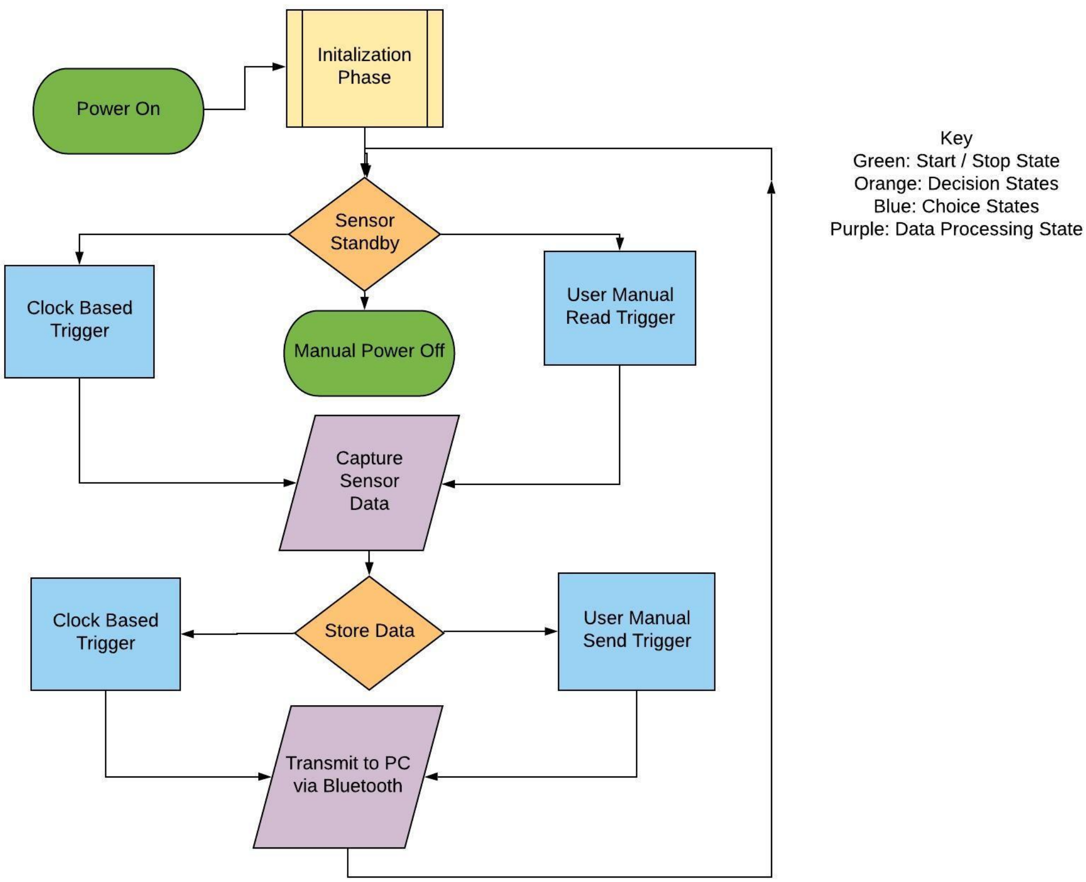


Figure 16: Low Level User Interaction Flow Diagram

5.6.5 Embedded Software Design

P-QUAD's primary focus is to monitor plants and modify the environment based on needs and given parameters while keeping power level at a minimum. The embedded architecture was designed using a state-machine framework that utilizes low-power settings and interrupts to maximize battery life. The first state called "INITIALIZATION", begins with initializing the on-chip modules and prompting the user for the desired P-QUAD template. After the template has been loaded into the P-QUAD device and the system has been initialized to the proper settings, the device enters the next state, called the "POLLING" state. The polling state prioritizes low-power and minimum resource usage while periodically checking for flags based on temperature, humidity, light, etc. The machine continues to run until the internal batteries have completely died or until the user terminates power to the device. If an environmental condition has caused an alert flag to be set, then the P-QUAD device enters the "ALERT" state, in which the P-QUAD device saves data pertaining to potential hazards indicated by the environment variables and sensors. These are represented in figure 17 below.

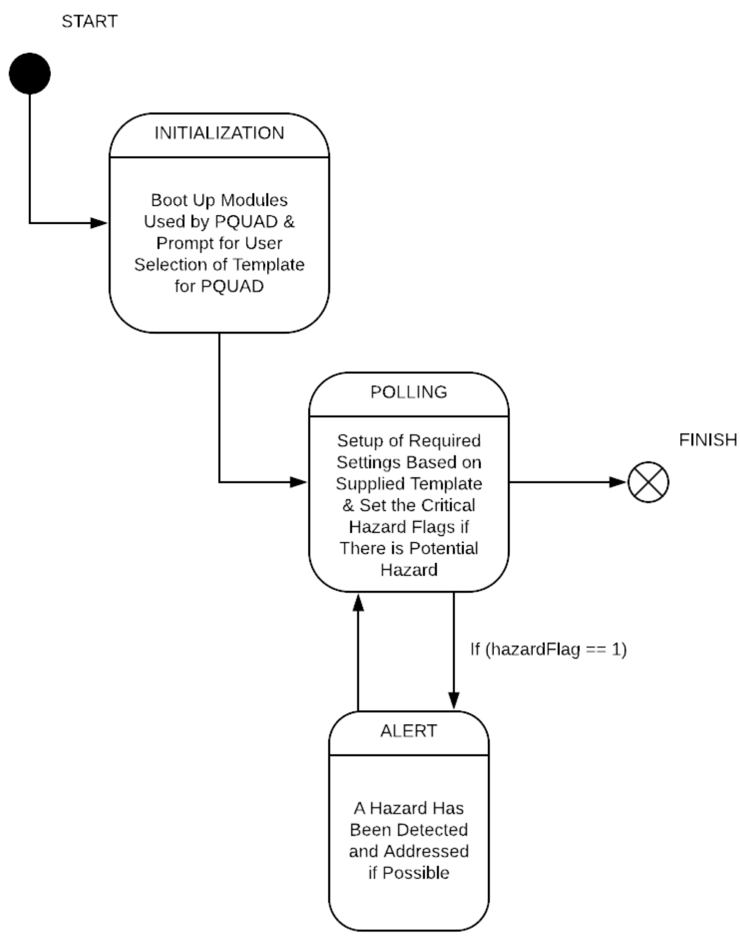


Figure 17: Embedded System Diagram

5.6.6 Software Implementation Plan

P-QUAD's plan for development is integral in completing the project in a complete and efficient way. Our plan consists of a total of six stages that include: planning, analysis, design, implementation, testing & integration, and maintenance. By the conclusion of this paper we should have made it into the implementation stage. During this stage we started developing source code for the system. Within our implementation stage, we used an agile development model. Through the use of agile development, shown in figure 18, we can reduce the strain on the software team if there are any unexpected or unforeseen changes to the software specifications.

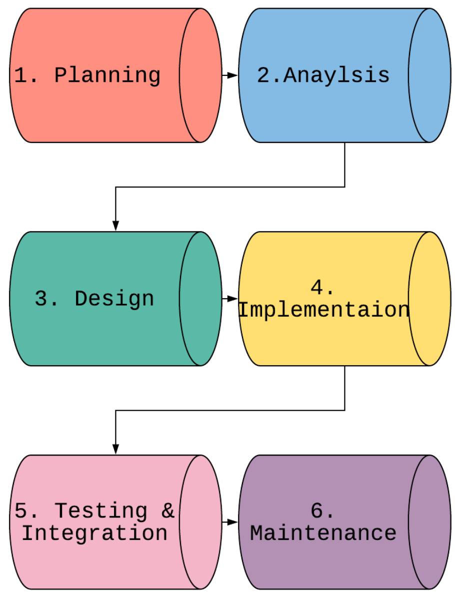


Figure 18: Software Implementation Plan Diagram

5.6.7 Agile Development

The agile methodology was chosen because of the structure. In the agile development model, the software team works in sprints to complete a subset of requirements by the time of the next sprint. The requirements are composed of tasks that each member worked on to complete the sprint. After each sprint, we can access the number of requirements we have satisfied and adjust based on the progress from the previous sprint. The flow of the sprints is visualized in figure 19.

Figure 19: Agile Development Process

5.6.8 Software Design Implementation and Research

The following sections discusses the implementation details for the software design of P-QUAD's system. Throughout each section the research conducted pertaining to the implementation specific strategy is also included in the section.

5.6.9 Mobile Application

Despite having an interface for the user to interact with on the P-Quad we felt it pressing to look into a mobile application platform as an additional platform of support for the P-Quad given the fact that today's market is making a large divergence towards mass integration with mobile platforms such as android or iOS. While we might consider some control aspect for the P-Quad this would be a secondary goal as the system has all the controls on it easily accessible to the user. However, the more substantial issue we could overcome with a mobile application would be the issue of parsing and indexing the data given from the serialized outputs. A solution to this issue would to be to create an app that acts as a data logger/acquisition platform. A data logger is traditionally an electronic device, but more recently can be web or mobile application platforms whose goal is to record a stream of data over time in relation to a location via sensors or external instruments. It is the job of these data loggers to continually parse the data stream from the sensors or instruments and store them in a manner that the user would later be able to access them later. Typically, this data is accessed via a database, either local or service-based, and stored in such a manner that it can be transmitted across various platforms such as a desktop or mobile application at a later point in time. The reason this would be a good solution for the P-Quad is the fact that with the on-system screen we are limited with the amount of data we can display at any given time due to size constraints of the screen. Additionally, we could run into issues with storing large tracks of continuous readings to the MCU due to the data being stored being much larger by comparison than the available memory of the chip. Therefore, the proposed mobile software application would negate these two issues and allow the P-Quad to meet more of its intended project scope. The general design of the application would be as follows: a routine in the MCU would poll the data from the sensors like normal and store it for a predetermined amount of time which would be over multiple sampling periods, after this time passes the MCU would transmit the data via some connection model (Bluetooth, wi-fi, radio, etc.) to a database (either local or service-based). The data would be transmitted in some form of standardized format such as XML, JSON, or YAML in order simplify the relaying process to the database.

Once in the database the user can call the data using a mobile application to present the data in an orderly tabular matter using a GUI programming with different options to chunk and sort the data (by minute, hour, day, week, and month potentially) so that they can gauge the environment the plant has been growing in over a duration. The next portion of the application development cycle discussion is split into two major sections the back-end development and front-end development.

5.6.10 Swift

Swift is a mobile development language that allows users to write code to produce Apple IOS mobile applications. The purpose of Swift was to create a more comprehensive and updated version of objective-c. Although the creators wanted to implement embedded c in their own way, the code bares very little resemblance to objective-c. In 2014, Apple introduced Swift to the World-Wide Developers Conference. Upon its release, Apple noted that Swift would only be compatible in their own IDE. For someone to develop mobile applications for IOS they need to have the Apple proprietary IDE called XCode. Therefore, it's no surprise that XCode can only be downloaded and installed on IOS operating systems. This is a major concern for developers. If there are developers who don't own a Mac than they won't be able utilize Swift in their development. The specifications for a computer to run XCode are described in the table 19 below.

Table 19: XCode Hardware Requirements

Required Operating System	Mac OS X 10.3 or later
Languages Available	Swift, Objective-C
Recommended CPU	i5 or better
Minimum RAM	4GB
Minimum Required Disk Space	128 GB

5.6.11 Android Studio

When developing mobile applications that are not required to run on IOS mobile devices, there are many more options available to the developer. The next largest competitor for mobile devices in the US are devices that run Android. Native applications that run on Android can be developed using Google Android software called Android Studio. Applications written using the Android Studio IDE are typically written in Java, but it is possible to write the source code in other object-oriented languages such as C++. The first benefit of using Android studio comes in the form of accessibility. There is no requirement for the operating system running Android Studio. The host machine can run Mac OS, Windows, or Linux. The table 20 below illustrates the specifications needed for the host machine to run Android Studio. Another downside to using Android Studio is that if the application needs to be run on an IOS mobile device, then Android Studio is not a viable option for mobile development. The developer must use Swift.

Table 20: Android Studio Hardware Requirements

Operating System	Mac OS	Windows	Linux
Languages allowed	Java, C++	Java, C++	Java, C++
Minimum RAM	2GB	2GB	2GB
Minimum Disk Storage	2GB	2GB	2GB
JDK (Java Development Kit)	JDK 6	JDK 8	JDK8

5.6.12 React Native

After reading into the most common mobile development platforms, it seems that the biggest issue between platforms is what devices the application would compatible with. In the world of Mac and IOS devices Swift is the only option and although there are other options than Android Studio there is still the issue of not being able to deploy the app onto IOS native devices. For a developer to deploy the exact same application across different platforms, one of the best options is React Native. React is a JavaScript library created specifically for developing user interfaces. React was developed by Facebook in 2013 to provide developers with a way to create web applications. Later in 2015, at a Facebook conference they announced the release of React Native. React Native uses the React architecture used to create web applications and overlays them onto IOS and Android applications.

In comparison to the other development platforms, React Native has various advantages that our team could benefit from. The first benefit is that React Native is a child of JavaScript language so writing the code for React Native applications are written in JavaScript. This includes all of the fundamental UI that is capable in objective-c or Java. One feature of React Native that no other development platform utilizes is called Hot Reloading. What Hot Reloading allows you to do is to load libraries into the applications at execution without affecting the state of the application. This means that as a developer, the application can still be running while we inject new files that have been edited at run time. This could be a substantial debugging tool while we develop the application. Like I stated earlier, both Apple and Android applications can be written using React Native and React Native gives options for optimization on different devices. The last feature of React Native that provides a huge advantage is the debugging and testing application called Expo. If the development computer and device both run Expo, the developer can run the application onto the phone through the Expo client app. The only caveat is that the host machine and device must be connected to the same wi-fi network.

There are many ways to develop React Native code, and the most commonplace way is through the use of a text editor that acts as the IDE. There have been many reviews on which text editor is most useful to develop using React Native. The most commonplace is Atom.io as the IDE/text editor. The following table 21 shows a few different options that we could use when developing with React Native. (68)

Table 21: Development Environment Comparisons

IDE/Text Editor	Atom	Deco IDE	Sublime Text 2(or 3)	Visual Studio Code
Operating System	Mac OS Windows, Linux	Mac OS	Mac OS Windows, Linux	Mac OS Windows, Linux
License Required	Open Source	Open Source	Free Download, Pay for License	Free To Download

5.7 Backend Development

The often-unseen portion of application development, backend development is a crucial part to the functionality and performance of any application. The backend performs multiple integral roles such as where the storage of data transmitting to a from the application took place. Backend also often acts as a computation offset to the entire framework as typically the user wants as little load calculation placed on their end as possible. An application's backend remedies this by using the its frontend interface to make calls to the backend so that it may perform the more intensive computations. These results are then either stored in the backends database aspect for later access and usage or the results are transmitted once again to the front through the use of the backend API in order to give the seamless appearance that whatever operation that appears on the user's end has been completed on the user's environment entirely while the resultant operation took far less of the user's local resources than it would if the task had been performed on the host machine itself.

5.7.1 Databases

One of the first things to consider in our applications development cycle is where the data was stored in order for the user to access it and analyze if the plant has been transported under proper conditions. In a way this is the crux of most of the application as if the data is unreachable or presented in a way that obfuscates its nature from the user then it is largely worthless. When looking at a database, the first thing we must consider is should we use a local database, or a service-based database. Starting with a description of the latter, a service-based database is a that can only be accessed via a server using an MDF (Media Descriptor File) and must have a constant server service/daemon running in order for it to be accessed since this is what processes all requests and accesses. It should be noted that none of the data would be accessible unless access to the server is granted. By comparison a local database is a database which stores all of its data to a local medium such as a hard disk drive, compact disc, or magnetic tape. Before going further, it should be noted that the nomenclature of local has some mixed meaning. A local database can be hosted and accessed online via internet or other connections such as LAN. By using the term local we refer to the locality of the data stored not the method of contacted the database itself meaning that while a local database can be hosted to a server changes to it can be reflected without having to call access to the server via an MDF file. Comparing the two types for the purpose of P-Quad we decided that it would be best to go with a local database since it proved simpler and faster to reflect the changes locally and then have them stored to an online host rather than to parse everything through a server. (25)

Next, we had to decide would we host the database on our own hardware or rent out space from a database storage. While hosting on our own hardware certainly gives us the most control it also puts addition expensive on us to assemble our own host and the cost, we would incur for the uptime for said host in addition to any troubleshooting of the hardware setup or integration with the application were among the reasons we decided to look into paying for a database hosting service. However, we have a lot to consider in this choice which was broken down and analyzed in the subsequent part addressing our different options for hosting. (24)

Before we dive deep into the various options, we wanted to understand the difference between relational and non-relational databases and how this difference could impact our selection and ultimately performance of the application. Of the major ways of structuring a database, relational databases are the oldest technology available. The core idea behind relational is to provide an expressive query language in order to manipulate the data in highly sophisticated way such that they can provide use to both operational or analytical applications. In addition to this the secondary functionality often tightly coupled with the first is the idea of accessing the data via an index in order to work alongside the query language to perform operations, it should be noted that the indexing should take place entirely on the databases side and not on the side of the application. A unique byproduct of these requirements is that it produced a largely universal language known as the structured query language or SQL which is based off the mathematical fields of set theory and relational theory (largely universal refers to the fact that it is an approved standard by the ANSI and ISO with the core language structure remaining the same but can often include minor tweaks varying from provider to another). One of the biggest benefits of SQL is the speed which it can carry out complex queries into large databases and extract specific data. A downside to the SQL/relational database paradigm is that interfacing and parsing the data for applications remains a non-trivial task despite the plethora of resources for the topic. (26) (27)

The other common database solution available today is the non-relational database which by the nature of its design remains fairly unstandardized and comes in a variety of flavors such as wide-column NoSQL, graph stores, key-value data stores, and document stores. The power of non-relational, henceforth referred to as NoSQL, databases is that they can excel in fitting non-standardized or unconventional data formats such as chat messages, user logs and session data, large data designed from streaming such as video or images, and data from devices or items in the IoT (internet of things). In a sense NoSQL's greatest weakness is also its greatest strength, while the lack of standardization can often make interfacing and initializing a daunting task the open-endedness of its design paradigm is what allows it to fit so many data structures and types that may not work well or at all under a conventional schema table from SQL. Flexible data models aren't the only benefit of NoSQL databases, they also possess the ability to be highly scalable. What this means is that the database can scale out on any platform be it a locally deployed service or a cloud-based solution this enables nearly unlimited growth of a database while still maintain higher throughput and lower latency than their relational counterparts. (28)

Finally, having established some foundational concepts between the two let us compare the strengths and weakness of relational and non-relational databases in order to properly compare them. Starting with SQL based databases is they have a powerfully and fully featured language that allows for rapid performance of simple query commands such as insert, update, delete, or retrieve. SQL databases have well defined structures and schemas which can help to sanitize the stream of data input for devices such as P-Quad. Additionally, these schemas can be made to include date and times of the data's arrival and perform various sorting operations based off this. However, on the weaknesses of SQL databases is the storage of the data might prove difficult depending on the type of file formatting chosen to transmit the data and even overcoming this it might prove difficult to transmit this data to an application interface. Also, we would be remiss to no discover the possibility of SQL injections possible with the relational databases. While commonly an attack vector traditionally targeting websites a SQL injection is possible on any platform utilizing SQL for its database. SQL injection in which attackers exploit a security vulnerability typically in application software's user input field’s embedding SQL statements in them which causes unexpected execution of the command. This can cause the command to execute allowing an unauthorized change to the database or at worse case throw a handling error allowing the attacker to access to the entire database’s schema meaning that data in the system has been completely disclosed to an unauthorized source. Now to consider the strengths and weaknesses of non-relational databases. (29) Again, first and foremost the biggest strength of NoSQL databases is they allow storage of any data type primitive or complex and relational chaining of data can still be achieved to some degree in NoSQL by embedding documents inside other documents or by using multi-dimensional data types such as arrays. NoSQL also makes the storage type interfacing of data much easier for transmission to the application since the formatting of the document is typically kept in the JSON file format. While it isn’t largely relevant for this singular project in particular the scalability of NoSQL databases would be greatly beneficial in the event of the P-Quad being produced in a massive quantity the scaling up of the database would become a non-issue. The next benefit is cost which is much cheaper for NoSQL than its relational counterpart in fact it’s only around 10% of the cost as a relational database. Finally, their design makes them very resource lax meaning they do not have an issue continually running on lower powered devices. As expected there do remain tradeoffs for these benefits first tying back into security matters some NoSQL databases do not consider the atomicity of directions nor the trustworthiness of the information, they transmit this can make proactively defending them substantially more difficult even without the threat of an SQL injection. Another issue is due to the newer nature of its market presence there isn’t as extensive as support from the community or as many comprehensive literatures on the topic yet as it still remains a relatively young technology. Having a full range of knowledge of the general market solutions available for database technologies we now dive into several specific considerations and decide which would fit the P-Quad the best. (28)

5.7.2 Google Firebase

The first consideration is Google's Firebase which is not just a database, but an entirely mobile and web application development platform. From the database perspective the service provides a NoSQL cloud database with real time syncing across multiple clients in real-time. The data is all stored as JSON documents which allows us for easy communication across multiple platforms. Additionally, the provided client libraries are able integrate with a variety of platforms including Android, iOS, JavaScript, Java, Objective-C, Swift, and Node.js applications including these the database is also compatible with the REST API and includes bindings for several JavaScript frameworks including AngularJS, React, Backbone.js, and Ember.js. A unique benefit to firebase is it offers a service known as Firebase Cloud Messaging which is a cross-platform solution for messages and notifications on Android, iOS, and web applications (30). In regard to storage security firebase provides secure file uploads and downloads for firebase-based apps independent of the app’s connected network quality. The hosting policies of firebase deliver files over a CDN (content delivery network) which is secured through HTTP Secure and Secure Sockets Layer (SSL) which provides additional secure transferring of files. Finally, in terms of pricing Firebase offers three plans Spark, Flame, and Blaze however for the scope of the P-Quad we looked only into the spark plan. First the plan is provided free (free as in beer) with some limitations on its capabilities. It allows for 100 simultaneous connections with up to 1GB of storage for the real-time database with a 10GB/month bandwidth. The storage allowed in the firebase storage system overall is 5GB and there are the following limits on operations 20K/day writes and deletes and 50k/day reads (30).

5.7.3 Amazon DynamoDB

Another strong contender for consideration is Amazon's DynamoDB a key-value and document-oriented database. One of the major attractive things about DynamoDB is it has a large library of language bindings supporting Java, Node.js, Golang, C#, .NET, Perl, PHP, Ruby, Haskell, and Erlang as well as offering Hadoop integration. These factors make up for a highly versatile platform of support selection. An additional unique factor to DynamoDB is that purchasing a service is based on throughput rather than on storage which is a dynamic take on service provision that can allow for reduced pricing in cases of smaller storage databases. Scaling of the database is done for the user via an auto scaling setting that adaptatively allocates more cloud resources to users as their usage grows, it should be noted however that should the user choose this feature can be disabled and the user left in charge of all scaling of the system (31). The users and administrators of the database may also request the throughput to be changed and DynamoDB spreads the traffic load over a number of solid-state drive severs which can shape the performance down to a predictable level. As of 2013 Amazon also made a kit available to users for development in local environments while still being able to leverage a large portion of the Dynamo toolkit. One of the main selling points of DynamoDB is its claimed ability to reduce latency time for mobile applications due to the serverless system. In regard to pricing Amazon provides the free tier of DynamoDB to any users and begins to charge once these limits have been exceeded. The free plan includes 25GB of storage per month, 200 million requests per month divided by 25 write capacity units and read capacity units (1 write capacity unit providing up to 3600 writes per hour and 1 read capacity unit providing up to 7200 reads per hour) with the ability to deploy globally in up to two Amazon webservice regions. Finally, on demand backup is offered at a rate of $.10 per GB a month which only includes the active amount of GB at the end of the period not targeting data removed or deleted (31).

5.7.4 Oracle

A more traditional platform is Oracle RDBMS or as it is colloquially known as Oracle DB. Oracle DB is the longest existing commercial relational database using SQL in the market today. In addition to the standard features common to SQL databases the Oracle DB hosts a rich and versatile body or tools and support to complement the database. The first of these is Oracle's active data guard which limits access to data in physical standby databases to a read only state so that more intensive tasks, such as reporting, data extraction, and backups have a minimal intrusion on the response times to their commits (32). In regard to scalability oracle offers scalability plans based on real application clusters which is Oracle's database repeated redundantly over a computer clusters architecture to ensure high availability while still being able to offer large enough room to expand for customers. Oracle offers an array of analytical tools for the database including a partitioning tool which stores large objects in a collection of secular small pieces that appear seamlessly in the real time application as a uniform object. Oracle also leverages enterprise R language environments to allow in-database data mining algorithms and online analytical processing models and queries while also supporting specific pre-constructed data models based off an industry specific dataset including models for retail, communication, airline, and utilities. For the performance aspect Oracle offers several features like advanced compressing which features informational lifecycle management capabilities, In-Memory which is a column oriented in memory store which is integrated with the aim to improve the performance of analytical workloads while attempting to work around oracles traditional row memory format and mitigate performance loss that can come with this, and finally a caching system known as TimesTen Cache which allows the caching of a database's subset in the application level in order to improve response time.(32) Oracle DB also boasts a wide range of security features such as transportation data encryption, data redaction, enforced segregation of duties, principles of least privileges, and fine-grained label-based access control. Finally, in regard to pricing Oracle offers two primary models for payment pay as you go option and monthly flex rate both of which break down pricing costs by OCPU (Oracle Computing Unit) per hour with the two options going for $.4032 and $.2688 respectively for standard edition oracle database (33)

5.7.5 Microsoft Azure

A cloud platform that has been growing in popularity over the past few years is Microsoft's Azure which is a comprehensive cloud computing platform created for the building, testing, deploying, and managing of applications and application services via Microsoft owned data centers over a globalized network Azure's scale allow it to provide solutions via SaaS (software as a service), PaaS (Platform as a service), and IaaS (Infrastructure as a Service) while supporting a wide range of programming languages, tools, and frameworks both first and third party (45). Azure offers mobile services such as mobile engagement collection which uses real-time analytics that highlight's user behavior and provide push notifications to mobile devices in addition Azure's HockeyApp is a tool that can be used for development, distribution, and beta-testing of mobile applications. For its storage end support Azure provides support for both REST and SDK APIs for storage and access of data in cloud environments as well as alternative storage modes such as a NoSQL table store, which lets programs store structured text in partitioned collections that are then accessed via a partition key similar to a lookup table, a blob store , which allows programs to store unstructured binary or textual data as blobs that can then be accessed via a HTTPS path this additionally provides security mechanisms to control access to said data, next is queue service which allows for communication between programs asynchronously using queue structures. With regards to how it handles databases Azure provides several solutions, the first being CosmosDB, a NoSQL database service implemented as a subset of the SQL SELECT statement on JSON documents. A unique feature about CosmosDB is the feature that makes it dynamically tunable along three dimensions the throughput, storage space size, or consistency with each of these featuring multiple levels of tuning in order to narrow down the exact performance cost tradeoff desired for a service. CosmosDB is supported by the following environments .NET, .NET Core, Node.js, Java, and Python and may also be accessed via the REST API (minus direct mode access). In addition to this support CosmoDB offers several different querying mechanisms such as SQL-like language adjusted to match JSON data types, Microsoft's LINQ (language integrated queries) library, JavaScript language integrated queries for the server-side SDK modeled after Underscore.js API, and finally support for the MongoDB query language via MongoDB driver level protocol support. Azure offers a more traditional database approach with the Azure SQL Database option (formerly and henceforth referred to as SQL Azure) (44). Features of SQL Azure include a wide range such as continuous dynamic scaling to minimize application downtime. Additionally, it offers extension into Microsoft's cloud environment using Microsoft SQL Server. SQL Azure targets both business and consumer applications both web and mobile environments and includes multiple database management for multi-tenant applications. SQL Azure is known for its ability to create development and test databases in order to speed up the overall development cycle. One thing noted about the Azure service is that it has predefined classifications in order to help the user decide what pricing range they wish to invest starting with basic target at small databases supporting a single active operation best in mind for development or testing databases and small scale applications, standard is the traditional cloud service package providing multiple concurrent queries includes use cases such as a workgroups or web applications, and finally followed by premium which is designed for high transactional volume with the ability to support many concurrent users typically selected for databases supporting mission critical application (45). Looking at the basic package and breaking down the cost into a rate comparison of DTUs (Database transaction units) per month Microsoft's Azure tends to be a more pricey solution with the basic package offering 5 DTUs a month and a max storage size of 2GB at the price of $4.8971 per month; a DTU is based on a bundle measure of computes, storage and IO resources typically measure for single databases and customers who wish to have simple preconfigured resources compared to the vCore based model (45).

5.7.6 MongoDB

A new and popular choice for the non-relational databases is MongoDB. MongoDB was built around applications that have databases that use both structured and unstructured data. The general overview is the database engine works to connect the databases to MongoDB applications using their database drives which include support for the following languages C, C#, C++, Java, Node.js, Perl, PHP, Python, Motor (Python Asynchronous), Ruby, Mongoid (Ruby Object Documented Mapper), and Scala. MongoDB supports field and range queries as well as regular expression searches and searches can return specific fields of a document also including user defined JavaScript functions (42). Despite the fact MongoDB wasn't constructed to handle relational data models it can process these however often at the cost of some performance issues and even supports indexing fields in a document with a primary and secondary index. MongoDB supports horizontal scaling using sharding in which the user chooses a shard key, this is what determines how the data in the collection is distributed, and then the data is split into ranges based on said shard key and distributed across multiple shards (a shard being defined as a master with one or more slave nodes) which can then be distributed across multiple servers in order to achieve load balance or to duplicate the data in case of hardware failure (43). A unique feature of MongoDB is that it can be used as a file storage system called GridFS which MongoDB provides the functions for file manipulation and content for developers. GridFS divides a file into parts, also called chunks, and stores each of these chunks on a separate document. MongoDB provides three separate solutions for aggregation of data aggregation pipeline, map-reduce function, and single-purpose aggregation methods. Map-Reduce can be used in cases that require batch processing of data and aggregation options, while stringing together aggregation operators can form a pipeline similar to the Unix pipeline in order to streamline the process and enables results comparable to the results for SQL GROUP BY

(42). When it comes to pricing MongoDB provides a community server edition free to any developer which however lacks some configuration to security of the server and a lower throughput and less predictable memory latency than its enterprise counterpart with a storage capacity of 512MB and shared RAM per application non-scalable (43).

5.8 Project Source Control

The software to make P-QUAD's system function completely requires us to use some form of source control or version control to prevent the us, the developers, from breaking the entire application when changes are made. Before the different source control tools are discussed, the different ways in which version control is handled on the server end are discussed and their similarities and differences are also examined. After we have discussed the different types of servers that use source control the different options available for us to use for P-QUAD's version control is discussed and then selected.

5.8.1 Client-Server Source Control

The first type of version control software is the client-server model of source control. This form of source control takes a similar approach as a webserver does for the clients that it is serving. There is only a single source that each of the clients are accessing at any given time. In client-server source control model, there is only a single repository that contains all the code for a given project. All the developers then share that one repository. There are a couple of benefits to using this form of version control. One of which is that all the developers were working on the same code, which makes it easier to communicate about different problems or debugging of code. Another benefit is that the whenever changes are made to the repository, all the other users see the updates in real time when looking at the repository. One of the downfalls of this control scheme is that when users make changes to the repository, any of the code that other developers are using that gets changed could break the application build on the next compilation. Another obstacle one might face has to do with web connectivity while working on the project. If work is done offline from the server and the code on the server changes before being reconnected, then the work being done offline could be behind the new code in the repository on the server.

5.8.2 Distributed Source Control

The next type of version control is the distributed source control approach. Distributed revision control systems do not adopt the same single shared repository structure that client-server model uses. Instead, the approach is to use a peer-to-peer approach to source control. Rather than a single, centralized repository that all developers synchronize to, each developer has their own working copy of the codebase that is completely separate from the centralized database containing the code. This immediately results in some differences compared to client-server approach. There is no single reference copy of the codebase, there are only working copies. The operations like commits, viewing the history of the repository, and reverting changes become very fast because there is no need to communicate with a central server. The only time communication is necessary is when changes are being pushed or pulled from or to other peers. This means that each working copy functions as a remote backup of the codebase and its change-history giving a much better way of protecting from data loss in the codebase or across peers.

The table 22 below shows the advantages and disadvantages of using the two different types of source control methods.

Table 22: Comparison of Version-Control Models

	Centralized Source Control	Distributed Source Control
Advantages	• All developers work on the same single repository, so discussion of defects and debugging is much easier • Final release version resides on the top of the source control line	• Allows users to work productively without being connected to a network • Common operations are faster (no need to communicate with central server) • Allows private work, so user drafts don't affect other users • Working copies function as remote backups which removes the single point of failure of centralization • Various development models can be applied to the maintenance of codebase • Centralized control of "release version" of code
Disadvantages	• The central repository holding the codebase becomes a single point of failure if the machine holding the codebase is lost or corrupted • Everyone must know the changes made in a recent change before applying their own	•

After investigating the advantages and disadvantages of the different source control schemes, it becomes hard to determine the significant disadvantages of using a distributed model compared to the centralized model shown in figure 20. Using a distributed version control system only provides more options and control in comparison to the centralized control system. If we wanted to implement a centralized control system on a distributed model, we could. Due to the popularity of distributed version control in the industry, we have decided that there are many advantages to using this model over the centralized approach.

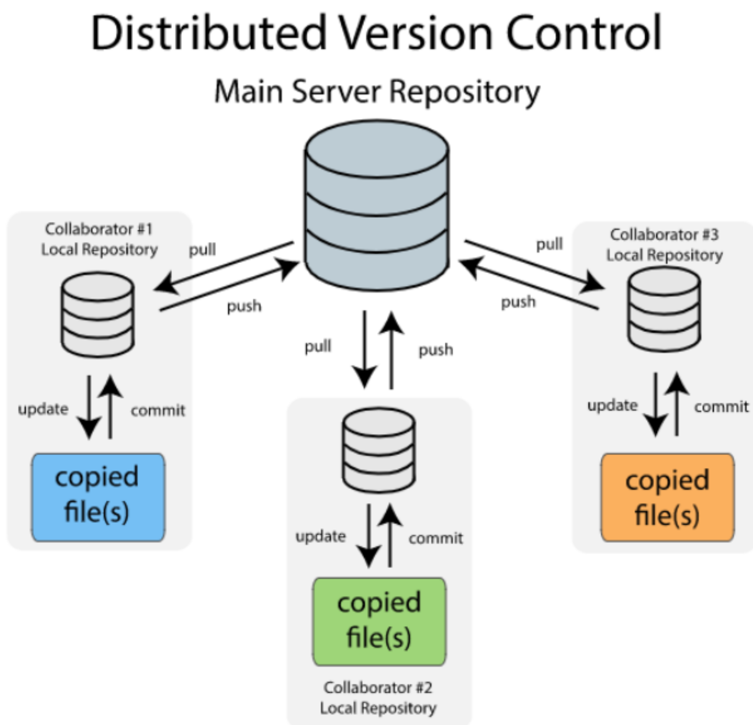


Figure 20: Distributed Version Control

5.8.3 Apache Subversion (SVN)

Apache Subversion, also known as SVN, is centralized software versioning and revision control system that is open source under the Apache License. It was created in 2000 by CollabNet Inc. was an attempt to write an open-source version-control system that operated much like its predecessor Concurrent Versions System (CVS) that fixed a lot of the bugs seen in CVS while adding new features. Subversion uses an inter-file branching model from Perforce to implement branching and tagging. A branch is a separate line of development whereas tagging refers to the labeling of a repository at a certain point in time used to locate a certain point in time of development. In Subversion the only difference between branches and tags are the way in which they are used. All the versions in the branch maintain the history of the file up to the point in which it was copy including any of the changes made since the copy. The use of merge commands can be used to move the changes from one branch to another branch. The following figure 21 shows a potential workflow of a Subversion project. Although there are many projects that have successfully used Subversion and have been satisfied with their experience, we have decided that we used a distributed control system for the implementation of P-QUAD's source control.

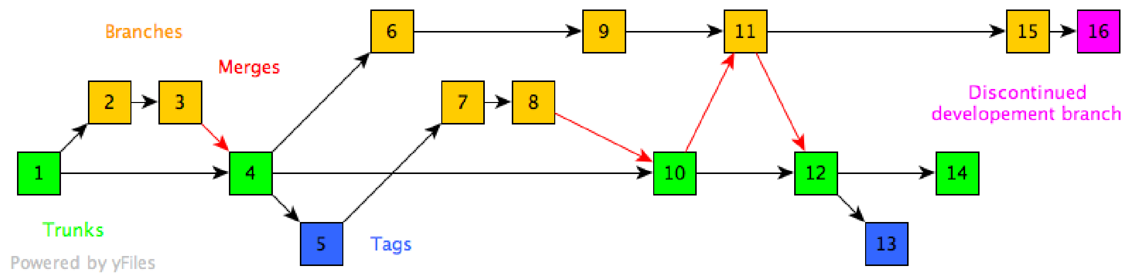


Figure 21: Flow Control Using Subversion

5.8.4 Git

Another highly used version-control system in the software engineering industry is Git. Unlike Subversion, Git uses a Distributed revision-control system to manage and track changes in a codebase. It is aimed at speed, data-integrity, and support for distributed, non-linear workflows.

Linus Torvalds, the same creator of the Linux operating system, began development of Git in 2005 with several other kernel developers. Since then, the current maintainer is Junio Hamano. Due to the nature of distributed version-control systems, each of the developers on P-QUADs team would have a full-fledged repository on their own personal computers to develop software. Git is free and open-source software that is under the terms of the GNU Pubic License version 2.

Git also has a variety of different workflow models to help organize the control over a codebase of a project. In the next few sections we discuss the details and implementation of the different workflow models. (74)

5.8.5 Centralized Workflow using Git

This workflow is the most like a Centralized Version Control System such as Subversion. It utilizes a central repository to serve as a single point of entry for all the changes to a project. The main development branch is called the “master” branch and is the only branch used in the project (this is called the “trunk” branch in other centralized systems). Using this workflow in combination with Git provides a few advantages. The first is that every developer has their own local copy of the master branch. This means everyone was isolated from one another and let each developer work independently from all the changes to a project. A single developer can forget about the upstream developments until it is convenient with them. (74)

5.8.6 Feature Branch Workflow

The core idea behind the Feature Branch Workflow is that all the development of new features being added to the system should take place in their own dedicated branch that is not the master branch. This encapsulation makes it easier for multiple developers to work on a particular feature without tainting the master branch. This means that the master branch was never broken, which was a huge advantage for continuous integration environments. Another additional bonus that our team could leverage when using the feature branch workflow is the use of pull requests. Pull requests are a way to initialize the discussion around the code involved in a feature branch. The provide the developers the opportunity to "validate" a branch before it gets integrated into the official project. This makes it much easier for others to collaborate on different developers work. (75)

5.8.7 Higher Level Workflow Designs

Gitflow was first published and designed by a man named Vincent Driessen at nvie. This workflow uses a much stricter branching model that is focused around different project releases. This provides a robust framework for managing larger projects. This doesn't add any commands or concepts behind the Feature Branch Workflow however, it does assign specific roles to different branches and defines exactly how and when these branches should interact with one another. In addition to the different feature branches that are used in the Feature Branch Workflow, there are different branches for preparing, maintaining and recording releases. The figure 22 below shows a possible flow of a project that is utilizing the Gitflow model. (75)

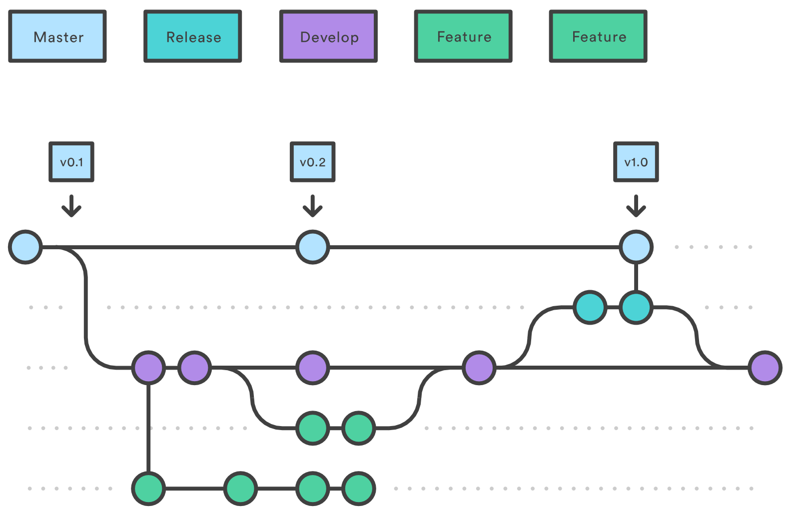


Figure 22: Example of Gitflow Workflow

<u>Forking Workflow</u>

The Forking Workflow shown 23 is fundamentally different then all of the different workflows that I have discussed so far. Instead of using a single server-side repository to act as the "central" codebase, each developer gets their own server-side repository. This means that each developer has two total Git repositories: a private local one and public server-side one. This workflow is often used in public open source projects. The main advantage of the Forking Workflow is that contributions can be integrated with needing everyone to push to a single central repository. The developers push to their own server-side repository and then the project maintainer can push the official changes to the official repository. This means that any developer can commit to the official repository but only if the maintainers gives them write access to the official repository. (75)

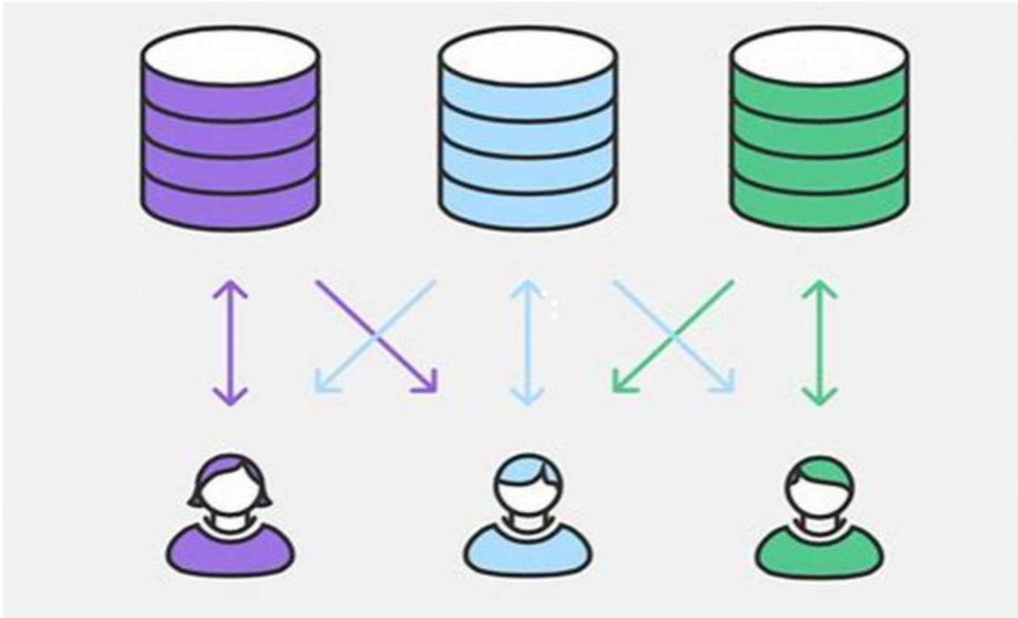


Figure 23: Forking Workflow Example

5.8.8 Source Control Selection

For the P-QUAD specific implementation we have decided that we used Git as our system of version-control. We liked the idea of using a distributed version-control system compared to a centralized system. It is also a toll that is widely used in our industry and getting comfortable using Git in a smaller project setting. Another reason we have chosen Git is that there are several workflow models that we can follow that help to make version-control much easier to use. Using these workflow templates can also make it easier to develop a coding plan for the project. The template we used is the feature workflow branch. This template makes the most sense for us because of the simplicity and organization. Each developer was able to work on their own copy of the project and before the changes are merged into the master branch, we can do a pull request to manage the code being added. Now that we must organize our project into features for the corresponding branches, we can assign features to developers and then the developers can work on their own branch in parallel with one another. The higher-level workflow models are overly complex for the project scope of P-QUAD.

5.9 Potential Software Issues

Although there are a number of potential issues that may occur on the software side of the project the first that occurs is the use of nested interrupts in our project. Due to the nature of our project, there are several timers that control the polling of different sensors that throw interrupts as the timer finishes. However, within that poling interrupt there are other interrupts that might be flagged within that first interrupt, which initiates the next service routine. To remedy the second interrupt might make a substantial amount of time so it was important to keep track of the priority of the interrupts and when they may possibly occur

5.10 Final Coding Plan

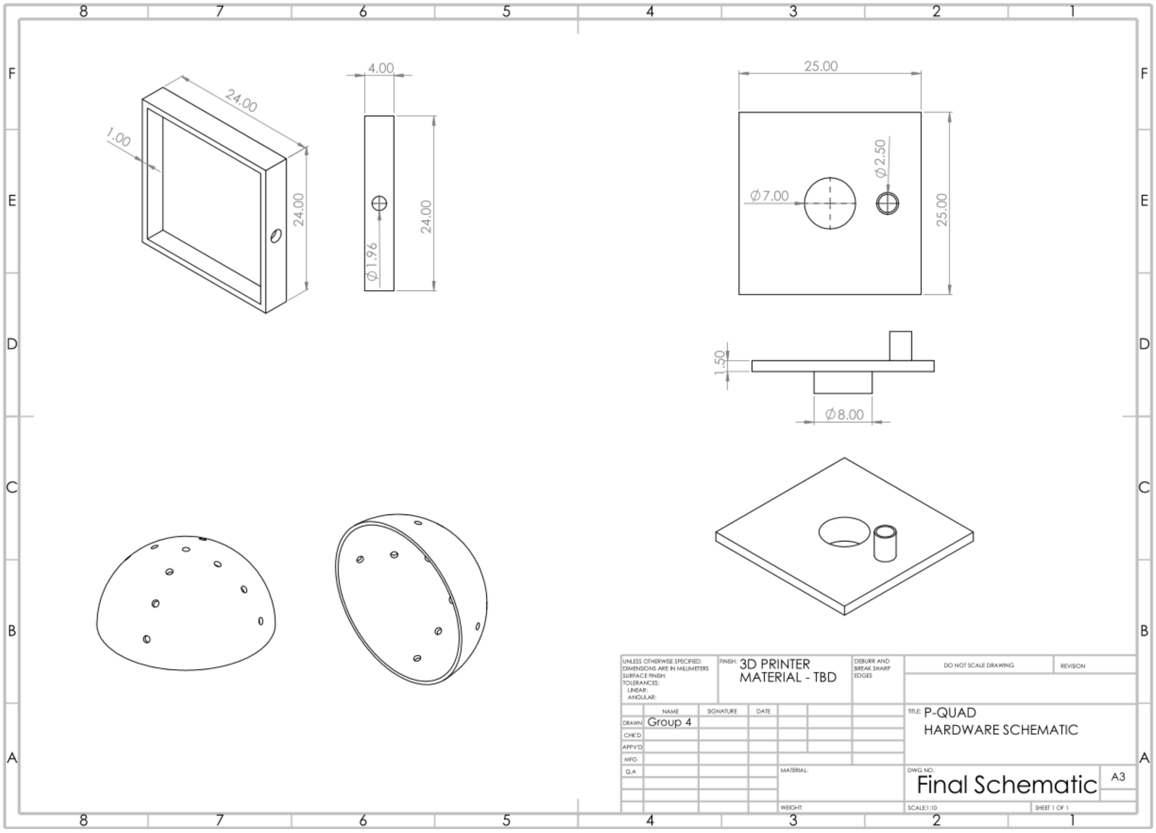
For the actual implementation of P-QUAD's software system and the development of the features, the following table describes the sprints, their tasks, the start date, and whether a feature branch was created. The use of agile as well as Feature Branch Workflow, made it possible to come out with a final coding plan for the project. Each sprint lasted a single week and the tasks involved may or may not be finished by the end of the sprint. In the event that we don't meet the criteria for the task to be finished by the start of the next sprint, the team collaborated to finish the unfinished tasks before moving on to the next task, this is represented below in table 23.

Table 23: Software Sprints and Tasks

Sprint	Tasks to be Completed	Start Date	Feature Branch Created
1	Initialize Git repository with empty main	12/01/2018	N/A
	Set up database structure	12/01/2018	Yes
	Begin mobile application development	12/01/2018	Yes
	Set up hardware timers for polling sensors	12/02/2018	Yes
	Write I2C driver (generic read/write)	12/02/2018	Yes
	Set up external interrupts for environment variables	12/02/2018	Yes
	End Sprint 1	12/07/2018	N/A
2	Program LCD Display Interface	12/08/2018	Yes
	Write SPI driver for Wi-Fi Module	12/08/2018	Yes
	Write message API for Temp/Humidity Sensor	12/08/2018	No (I2C driver branch)
	Get mobile application and database in communication	12/08/2018	
	End Sprint 2	12/13/2018	N/A
3	Get database and Wi-Fi module in communication	12/27/2018	Yes
	Clean up Mobile application GUI	12/27/2018	Yes
	Write API calls for accessing light sensor	12/28/2018	Yes
	Write driver for in device light fixture	12/28/2018	Yes
	End Sprint 3	01/02/2019	N/A
4	Testing	01/03/2018	N/A

5.11 Hardware Enclosure Design

We decided early one that rather than trying to retrofit a prefabricated enclosure or making one out of wood that we would come up with a design in SolidWorks in order to suit our specific needs in term of shape, thickness, and dimensions. Additionally, we felt that a 3-D printed enclosure would provide better sealing factor than wood or other materials like sheet metal which we felt would help keep that plant more protected and secure. Our prototype design is shown on the next page in figure 24.



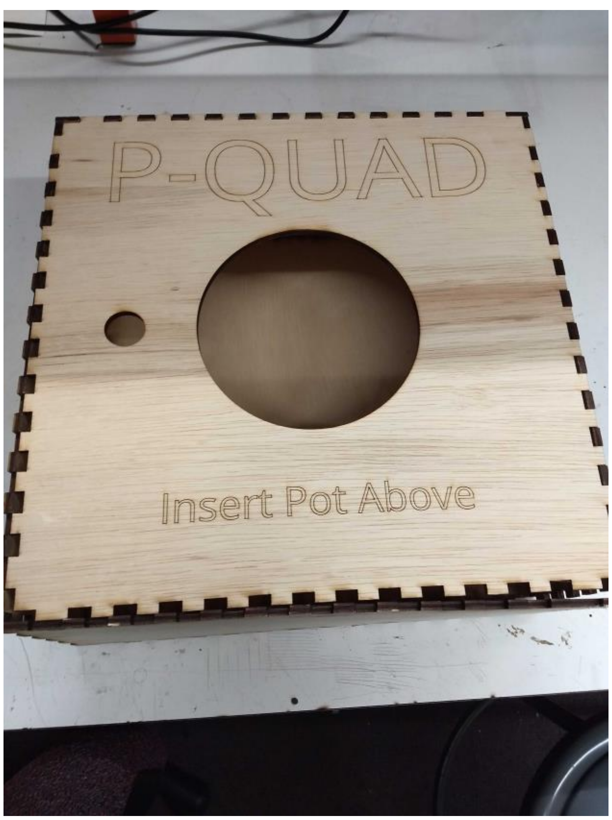


Figure 24a and b: A - Enclosure Housing Design B- Fabricated base Design

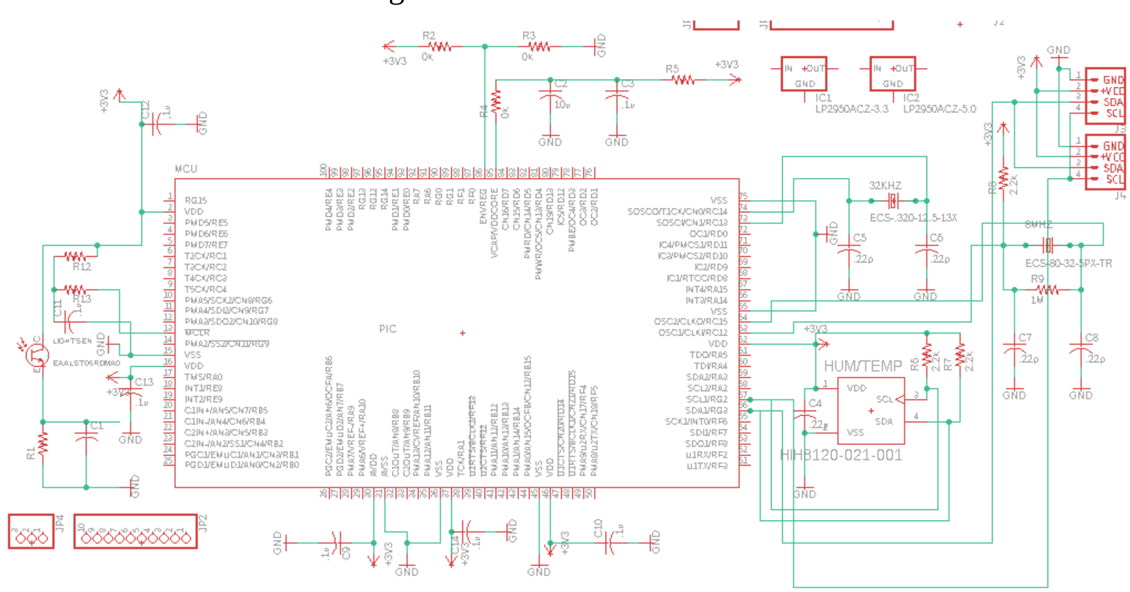
6 Printed Circuit Board

A printed circuit board is the brain of any small project like P-Quad. It is needed to make the system run, from powering all the individual components to transfer of data from and to the user. PCB is where most of the research and development time is spent to make sure the entirety of a system is functional. A PCB not only needs to work properly with proper hardware but also it needs a well implemented software, so the PCB is able to function the way it is meant to. The board is made of 4 different material that are called silkscreen, solder mask, copper and substrate all of these are explained in some detail in later sections of this chapter. Almost all printed circuit boards are mechanically used to support and also to connect electrical components together. This is achieved by something known as traces that connect these components electrically. For example, if we have on our PCB board transistors, capacitors, resistors or other devices we connected these using tracks, pads, and other features that are etched from copper sheets onto a nonconductive substrate.

6.1 Integrated Schematics

The schematics in figure 25, that were made for this project included all the sensors, battery, headers, DC power jack, crystals. For the sensors just connecting them to a power supply Vcc and connecting the pins to the microcontroller is not enough. To prevent ambient noise and to have the sensor function properly, we implemented resistors and capacitors where appropriate. For example, one of the sensors that requires a voltage with the range of 2v to 3.3v needs resistors to protect it from burning up. Headers as can be seen in the schematics are laid out on the edge, and they are what makes it easy to plug in the components via male to female or female to male connections. Headers are where most of the traces end up that provide either power or transfer data like 5V header or a header which used as SDA or SCL for data transfer. The schematic below shows how the printed circuit board is wired.

Figure 25: PCB IC Schematics



6.2 Different Software

6.2.1 KiCad

For the first software that we really wanted to use was KiCad. This software is free to use for student or even for professionals. KiCad even though, harder to use than Eagle it seemed very interesting and powerful. KiCad has five main parts that take a project from inception to completion. The project manager, the schematic capture editor, the PCB layout program, the Gerber viewer and finally tools to convert images to footprints for PCB artwork. KiCad's PCBnew supports up to 32 copper layers and 32 technical layers with a great feature called auto router which automatically calculates multiple paths for the PCB to connect all the I/O pins. The biggest advantage KiCad has over Eagle is the fact the library search is incredibly more user friendly. Where Eagle requires the user to input the name of the component almost exactly how it appears in the library storage data base for KiCad and numbers of letters shows the result. And from there the user can pick the parts. But in the end our group favored Eagle Software more than KiCad.

6.2.2 Eagle

Eagle software was what the group ended up using for a student Eagle is free to use however, for a full version it requires a subscription. The biggest disadvantage of Eagle is without a doubt the library. It is very easy to find new library parts and import those parts to the library however, when searching these or any parts in the library it is very difficult to actually find and implement them in the schematics. But other than this issue Eagle is what the group ended up using. We picked our microcontroller before checking the Eagle library if it is in there and it was not, and we could not find a symbol or and foot print on Snapeda or even Digikey. And in the following section we explain the entire process of importing a custom device in the Eagle library. (46)

6.2.3 Custom Library

Custom library is a crucial part of Eagle. For our project we initially chose a microcontroller Microchip PIC24FJ128GA006 and it was not in the library for Eagle and we were not able to find the symbol or the footprint for this microcontroller, so we ended up making a custom device for the Eagle library. There are many steps involved in making a custom device however, the two most important steps that need to be done is making a foot print that is the actual outline on top of the printed circuit board, and the second part is the symbol. After the two parts are made then they relate to their respected I/Os and the pins.

6.2.3.1 Foot Print

In order to make a custom device first thing we had to do was get familiar with the data sheet for PIC24FJ128GA006. During the part creation process we found that the data sheet was basically the best thing to use to make sure the end product would work properly. A package information from the data sheet as an important purpose like offering a set of pads that provide the component with both electrical connectivity to the PCB board and also to solder the part onto the board itself therefore, making sure the pads are placed exactly in the right spot and these pads having the right size for either surface mounted or through hole is essential because if the pad's size is not correct there is a chance of a short-circuit or the part might not even alien properly. What we had to do for foot print is consider four very important aspect of foot print creation explained below.

- Pads. The pads like explain above are the most important things that are physically laid on the board and they can be red for surface mount or green for through hole. And they are connected to a trace and provide electrical connection for the part these pads belong to.
- Dimensions. A package dimension is the size of the entire component and the space that the component occupies on the printed board including the pads and the traces. Making sure to have enough space between components is very important so they don't overlap each other therefore, making the mounting of the parts impossible.
- Silkscreen. This part is optional however, the outline for silkscreen lays on top of the dimensions for the component and helps the PCB manufacture.
- Name. This is a very important step. To name the part and imprint the name on the PCB helps when soldering the part on the PCB and afterwards to identify the part.

After understanding these basic we moved on to searching and gathering information from the data sheet. The first thing was to mind the pad dimension. The length and the width of the pads were created in the eagle. From there we measured the pad pitch which is the distance between the two pads on both sides on the X and Y axis. The next step was to figure out the pad coordinates which required basic math and input the coordinates in the grid system for precise pad placement. After placing the pad on the Eagle foot print work bench, we changed the names of each of the pads naming them 1 to 64 representing the 64 individual pins on the microcontroller. To help identifying the pads we put a circular image in the silk layer to mark pin one on the board. The next part was to input the silkscreen outline to identify on the board the entire shape of the component which makes it very easy to know which component went where. and the last part is to include the name and the value of the component in the silk layer which made it very easy to know exactly which component occupies that space. The foot print layout is very important to get right the first time that it is implemented because if there are any issues with the dimension being incorrect even in the slightest, the components may they be surface mounted or through holes, they did not fit and there was a huge chance of a short circuit because the traces might overlap each other.

6.2.3.2 Symbol

After the footprint was made for our microcontroller from Microchip PIC24FJ128GA006 the next step was to create the symbol. For the symbol the first thing was to get acquainted with the information from the datasheet about each of the sixty-four pins in the microcontroller. Microchip PIC24FJ128GA006 microcontroller is a very commonly used product and it has many I2C pins named SCL and SDA which was perfect for our project because all the sensors that we picked have I2C requirements. In Eagle we first drew made a rectangular shape that resembled the actual Microchip PIC24FJ128GA006 and made sure to leave enough space for the implementation of pins and their names. After a similar shape to the Microchip PIC24FJ128GA006 was made we implemented sixty-four pins in the same shape that the Microchip PIC24FJ128GA006 is laid out. After the pins were connected to the Microchip PIC24FJ128GA006 symbol we named each of the pins from one to sixty-four and then we named each of the pins. We had multiple VSS and VDD representing ground and power. And we also named according to the datasheet multiple SDA and SCL pins that gave power to the sensors and make them operational.

After the symbol was made in its entirety with all the correct pins and the correct locations with the proper names, we imported the symbol and the footprint that we made to the Eagle library manager and from there we connected the symbol and the footprint together. In order to connect the symbol and footprint together, we had to connected individual sixty-four pins on the footprint to the sixty-four pins that we made in the symbol. This is the most crucial part of making a new device and importing it to the Eagle library because we even one pin from the footprint is not connected properly to the pin of the symbol then the Microchip ATMega2560 will not function like it is meant to. Furthermore, we did not know if we connected the pins incorrectly until we ordered the Microchip ATMega2560 and implemented it in the PCB and then tried to power and run the PCB which was a mistake. This mistake could have caused a lot of headache and waste a lot of time. Therefore, we made sure that we have connected all the pins from the footprint to the symbol correctly and we doubled checked and compared with the datasheet of Microchip ATMega2560. After all the pins from the footprint and the symbol are connected, we finally imported the newly made device to the Eagle library. There is another option in Eagle if we want to only use the device we created in one specific project or if we want to import the device in the standard library so we can use the device whenever we want for a different project without having to import it again, we are very proud of this device Microchip ATMega2560 from scratch therefore, we not only saved it in our standard library but also uploaded it to SnapEda for other students to use.

6.3 PCB Terminology

On the next page we discuss common PCB terminology and key phrases that we encountered during our research and design process of the PCB list in table 24 on the next page sources (47) (48) (51) Printed circuit boards can be created in any manner that is fitting the designer of the board there are no specific design that must be followed. There are many different software packages like discussed before that can be used to design and order the PCB and there are many different companies that design and create the board for people. In order to effectively make a printed circuit board design and have the PCB made it is very important to understand the basic terminology below we explained some of the commonly used term that anyone working on PCB must familiarize themselves.

Terminology	Description
Via	Via are holes that are implemented in a PCB when there are too many traces that needs to be run from components from one layer to another layer in order to avoid traces on top of other traces. And they are used to pass signal from these layers. Vias are uncovered so that they can be easily soldered
V-Source	In order to snap a board along some line to make it of a fitting size V-Source is used to mark these lines
DRC	This feature is used in Eagle and is called design rule check. And is a software check of the design to make sure it does not have any errors like traces overlapping each other or touching each other and if the drill holes that are too small for components to fit.
Pad	A portion of exposed metal on the surface of a board to which a component is soldered thus making the electrical connection to the components.
Thermal	A small trace used to connect a pad to a plane. If a pad is not thermally relieved, it becomes difficult to get the pad to a high enough temperature to create a good solder to it and takes an abnormally long time to reflow.
Plane	A larger circuit board composed of many smaller boards which is broken apart before use. Automated circuit board handling Equipment frequently has trouble with smaller boards.
Surface Mount	An easier method of soldering components onto the PCB which out the need to put the leads through the PCB. This Construction method allows components to be simply set on the board and soldered. This is the main method of assembly now.
Plated Through hole	A hole on the PCB that has an annular metal ring and these holes go all the way through the board. These are mostly used to have a component soldered with long metal leads.
Trace	One of the most important part of any PCB are traces and they are a continuous path of cooper on a circuit board that connect components electrically.
Mouse Bites	These are number of drill hits that are clustered together in turn creating a weak spot where the board can be broken easily afterwards
Drilling hit	These are places on the PCB where a hole should be drilled or are drilled.
Finger	In order to connect two circuit boards together there are exposed metal pads along the edge of the boards, and these are called fingers.

6.3.1 Silkscreen

The top most layer of a printed circuit board shown in figure 26 that is used in order to write numerical values and names of components is called silkscreen. The letters, number, symbols and imagery on a circuit board can be implemented on the silkscreen of the printed circuit board. In the following figure we can see the model number for the battery and the printed word PWR and an image all are made on the silkscreen. The silkscreen is usually used on the components side of the board and it is very helpful in identifying different components and their part numbers and the test points. Silkscreen have a cost to it and when implementing silkscreen, we have to consider this cost.

Figure 26: Example Silkscreen

The norm for a silk-screening of printed circuit board requires screens of polyesters that are stretched across the aluminum frame. There are three different types of methods that are used to apply silkscreen to the PCB

- Direct Legend Printing (DLP)
  - This is the most accurate method for silk-screening and is most legible printing of the letters.
- Liquid Photo Imaging (LPI)
  - This is a little less expensive than the direct legend printing discussed above however, it is less accurate and less legible.
- Manual Screen-Printing
  - This is the most basic and cheapest method that is used in printing.

6.3.2 Solder mask

Solder mask is a thin layer of polymer that is applied to the copper traces that are implemented on the printed circuit board. Solder pads on the PCB that have solder joints on them after they are dipped into the solder pot. Solder bridges are electrical connections that occur by accident and they can cause a short-circuit and maybe even fire and to prevent this catastrophic damage to the PCB solder mask are used to protect the printed circuit board from against oxidation and help prevent these solder bridges from causing fire and damaging the PCB. Like discussed before the top most layer in a printed circuit board is the silk layer and right underneath the silk layer is where the solder mask is placed on top of the copper layer. The coating of the solder mask can help from any kind of electrical shorts and even from corrosion of the printed circuit board. Another benefit of solder mask is it works as an electrical insulation in turn making it possible to have traces that have higher voltage to be placed closer to each other hence, making more space for other components and having additional traces, making a complex printed circuit board design possible. When a company comes up with a printed circuit board design they mass produce the PCB and for that using solder mask is essential however, it does make it easier and more efficient if we solder the PCB by hand making sure that all the traces and all the components are placed in the correct place and that all the parts are safe to use.

6.3.3 Copper

After the Silk layer is the solder mask and then comes the copper layer which is a thin copper foil like aluminum foil that we use at home however, the copper foil in printed circuit boards are laminated to the board with heat and adhesive on both sides of the printed circuit board. The reason both sides of the printed circuit boards are covered in copper foil is because there can be different components that are soldered on both sides and there needs to be copper foil for connectivity for the parts. The thickness of the cooper can vary and is specified by weight in ounces per square foot the most commonly used is 1 ounce of copper per square foot. If a project has much higher component density, then a multi-layer printed circuit board is used because circuit traces from and to the components on the inner layers would have taken up too much surface space. (62)

6.3.4 Substrate

Substrate is a very important layer on the PCB. As can be seen in the figure below. Substrate is the middle layer that hold the circuit and all of its components and the traces to the board. In addition to holding the components to the board, substrate also provides insulation between the conductive parts so there is no chance of any electrical shorts or fire cause the printed circuit board to be rendered useless and costing time and money. Some of the most common types of substrate used are FR-4 which is a fiberglass-epoxy laminate material. However, there are many different types of substrate material used and below are some of these and how they work.

- Ceramic
    - Due to their low demand for thermal resistivity ceramic substrate are used in power electronics.
- FLEX
    - Some small devices like remote controller or some older phones are small and compact, and they demand some flexibility therefore, FLEX is used as a substrate because it is thin and flexible plastic that can be fitted in places inside devices that have shapes that need to be more flexible and maybe even dynamic.
- RF
    - RF are used in printed circuit boards that are needed for produce higher power radio frequencies. RF are perfectly suited for that due to their low

dielectric plastics. And RF also work great with electrical performances properties.

- FR-2
  - The lowest type of substrate, FR-2 is a paper material with phenolic binder and they are the most inexpensive to use and are found in many inexpensive products that are used by everyday consumers.
- FR-4
  - The most commonly used material for substrate shown in figure 27. Using fiberglass, PCB with this material can be very rigid therefore, printed circuit boards comprised of FR-4 can be cut or drilled in many different shapes that suits the end user. FR-4 is of better and stronger quality than FR-2 and is more resistant to breaking and cracking due to their rigidness.

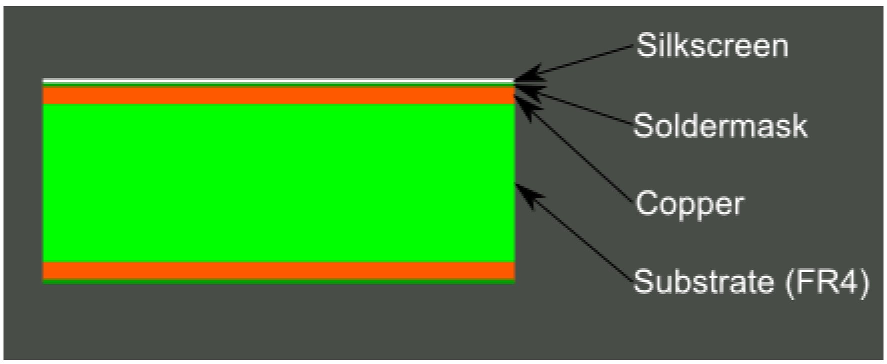


Figure 27: PCB Layers

The substrates act as a laminated electrical insulator because, as can be seen in the figure above, substrate is in the middle of the two conductive parts of the printed circuit board the copper on the top layer and the copper on the bottom layer. The insulators internal electric charge does not flow freely.

6.4 PCB Constraints

Designing a printed circuit board is not something that can be done without following guidelines that are put in place to prevent common missteps. There are many constraints that are taken into account when designing printed circuit boards for example the thermal issues, size, shape and even the track design, how far the traces are from each other and also what type of printed circuit board it is, all of these constraints are a factor that may affect the overall design of a project like this one. Printed circuit boards for each project can be as diverse as the projects therefore, there are no rules set for individual PCB but rather one set of rules, constraints and guidelines that when followed any PCB design turned out safe and functional. Below are some of the most important constraints that need to be followed.

- The first constraints that must be considered is the size of the printed circuit board. We made sure that we have sufficient space on the board for all the components to be placed in their proper pads and footprint. We made sure that we estimated the size of the board properly to accommodate all the traces. Traces are copper lines that help with the flow of electrons, basically they help with the flow of electricity from components to either the microcontroller of other electrical parts. Having sufficient space for the traces is crucial because any traces that are touching each other would have caused short-circuit and might lead to fire or damage to the entire printed circuit board.

- Many companies make their printed circuit board their own way therefore, it is a good idea to have reference hole on the circuit board that way it is clear for the text fixtures. Making sure that when choosing the reference point of the printed circuit board that we match the company where we are ordering from, we met their standards.
- Another important factor when designing a PCB is how many layers the printed circuit board have. Having multiple layers increases the cost of the printed circuit board. Depending on how complex the printed circuit board is, the amount of layers that we use might be very important because having more components on the printed circuit board warranted many more traces that need to me connected and to flow without touching other traces, and in that case more layers are needed for the traces and via to move to other layers without crossing over other traces.
- Another very important thing to considering when designing the printed circuit board is how the board was mounted. In the case of P-Quad we have determined that mounting the board inside the Pod area with a small pocket made of plastic. The printed circuit board is put in this small pocket to protect it from water damage. We have a small part of the printed circuit board free of any components or traces, so we can screw the board to the edge of the pocket therefore, reducing any chance movement of the board during transit.

6.4.1 Thermal Issues

Overheating issues are not very common with printed circuit board however, it is an important issue that needs to be considered when designing a board. Depending on the size of the circuit board overheating issues is more apparent. If the circuit board has a lot of components, then the thermal and overheating issue was something that was considered. When dealing with thermal issues we had to make sure that there in enough room between the components so there is more space for the heat to dissipate. It is a great idea to use temperature sensors that are surface mounted, next to the overheating component to maintain and manage the temperature and prevent overheating and damaging the part.

6.4.2 Traces Guidelines

One of the most important part of any printed circuit boards are traces and they are a continuous path of cooper on a circuit board that connected components electrically shown in table 25. The traces that are implemented onto the printed circuit boards carry only a limited amount of current to take electricity to and from components. The current that is carries through the traces varies depending on the component. Some parts that need to supply power to other components for example a power rail might need traces that can carry a lot more current then compared to smaller parts. For traces to carry more current the width of traces is taken under consideration.

Table 25: Trace Width Guideline

Width of trace for 1 oz	Width of trace for 2 oz	Current in Amps
10	5	1
20	15	2
50	25	3

- Implementing standard traces is essential to use in printed circuit board because the size and amount of traces matters. If all the traces that are used end up too narrow or small and to compacted together then there is a much higher chance of a short to occur ending in a fire or damaging of the components on the printed circuit board. However, if the traces used are too wide and to far apart then there is a lot of space that might have been wasted. Because all the traces must be implemented to make the printed circuit board work this led to multiple layer to connect all the components to the board. In that case more via were needed to take the traces from one layer to the other, this process ended up costing a lot more therefore, it is very important to implement the right amount of traces with the correct size and width.

6.5 PCB Details

Printed circuit board is the hearth of P-Quad because it monitors the environment inside the Pod and it sends the information to the LCD display for the user to see and the printed circuit board also gathers and save data for future analysis. For our protect we design the printed circuit board keeping in mind that it is in somewhat close proximity to water which can cause malfunction and also the printed circuit board is very close to the LED lights used for the plants thus increasing the heat printed circuit board experiences. We go into details of the printed circuit board for P-Quad to better understand how we were able to overcome these challenges and design the board.

6.5.1 PCB Powered

The first step to making any system function to complete a task, the system must be provided with power by way of flowing electrons i.e., electricity. In order to power our printed circuit board, we had to implement a female pin that was soldered onto the printed circuit board. The power source we decided on ended up being a wall wart that provided us with enough voltage that powers the microcontroller Microchip ATMega2560 and also it has enough voltage to run all the sensors and the LCD display. In order to choose the correct wall wart, we calculated the total voltage consumed by the printed circuit board when the PCB has all the components implemented onto it. We added up all the voltages from the sensors, LCD display, Microcontroller, and then multiplied it by .2 to have increase the voltage rating by 20 percent. This 20 percent accounted for any type of surges or blackouts. The wall wart we used is a DC/DC converter with a stepdown transformer built-in, plus a rectifier.

6.5.2 Voltage Regulator

Voltage regulators are very important to have in any printed circuit board that might need different amounts of voltages. In P-Quad we have a wall wart that is 5 Volt nominals and 2.2 Amps with maximum output wattage of 10 watts. Voltage regulator is used to increase or decrease voltage going to a system. For example, a voltage regulator that is stepping down voltage from ten volts to 5 volts has three pins. One of the pins that is connected to the incoming voltage of 10 volts and then another one that is the step-down voltage pin. The third pin sometimes is not used with is the earth ground. In the case of P-Quad we do not really need a voltage regulator mainly because we took care when choosing how parts like the microcontroller, the sensors and even the LCD display. Below is the table 26 that shows how much voltage, currents and watts each component in the printed circuit board uses.

Name	Description	Voltage	Current
Ambient Light Sensor	A low cost ambient light sensor which consist of phototransistor. Due to high rejection ratio of infrared radiation, the ambient light sensor is close to human eyes spectrum.	2.5V-5.5V	70uA
Microcontroller	High performance CPU with 100 pins with multiple I2C pins. With 128k program memory and 5 16bit timers	2V-3.6V	18mA I/O pins
Humidity/Temp Sensor	Digital output relative humidity and temperature sensors providing an accuracy level of +/- 2% for Rh and +/- .5C for temp	2.3V-5.5V	0.65mA
Arducam LCD Display	LCD screen that displays all the information for the end user.	5V	1.1mA

Table 26: Voltage Load per Peripheral

In P-Quad we implemented two voltage regulators so far; however, that number might be increased too three. Our main power is coming from a DC/DC wall wart that outputs 5V and 2 Amps. This 5V main power supply goes into a 5V to 3.3V voltage regulator that becomes the VCC power supply for the microcontroller and connected to all the VCC pins of the MCU. The second voltage regulator is used to stepdown the voltage coming from the 3.6V or even a 9V battery down to 3.3V and 5V. This supplied power to the microcontroller and the LCD display also the sensors when the main power is switched off.

6.5.3 Electrical Switch

An automatic electrical was implemented in the printed circuit board to accommodate the voltage supply from the two different sources that are sued by the printed circuit board. One of the power supplies is the wall wart that supplied the printed circuit board with all the power needed for the function of the system and the second power supply was a battery that provided the power to the entire system when the supply from the wall wart is cut off during transit. The main function of a switch that can be automatic is to open or close a switch that allowed the power to be drawn from either wall wart or the battery. We have decided to have the manual switch used for this function, so we can give the user more options and make it the users choice when to press the switch for the transfer of power from either wall wart or the battery. A manual switch is a component that opens and closes a latch on an electric circuit. This way switches allow the control of current flow in a particular direction into a system without having to open a system and change the circuit manually every time there needs to be made a change. Switches are very important components in P-Quad (58).

6.5.4 PCB Parts Powered

The parts used for P-Quad have different requirements that need to be met for them to function properly. All the parts like the sensors, voltage regulator, MCU, and the LCD display has power needs that differ from each other and mixing the different valued voltage from one part to the other can cause the component to malfunction and even be damaged; hence, it is imperative to make sure all the components get the voltage that is approved by the datasheet and within their voltage rating. All the sensors that we selected use the I2C pins on our microcontroller Microchip ATMega2560. All the sensors requiring 3.3V to 3.6V are connected to the I2C pins on the microcontroller, furthermore, all the I2C pins on the Microchip ATMega2560 are labeled as SCL and SDA. We have two pins for each of these so that we can connect all the sensors that need SCL and SDA can me connected via male to female pins and get the appropriate power from the Microchip ATMega2560 microcontroller. However, to over the LCD display I needed to implement a voltage regulator which stepped 5 volts down to 3.3v. Therefore, the voltage that comes coming from the wall wart is max 5V and then I stepped it down to about 3v to 3.3v. The 3.3V from the voltage regulator is connected to one of the pins of the Microchip ATMega2560 which is power the Microchip ATMega2560. To power the LCD display and one of our sensors that requires 5volts, I traced a line with one additional trace branching off the first trace to a male pin. Then we used this pin for VCC for the LCD display and the sensor that requires 5volts.

6.6 PCB Design

The printed circuit board design that was implemented in for this project is very basic. We made the board two layered with multiple via to make sure that each layer was grounded properly. When designing the printed circuit board there cannot be any “islands” which are small parts of the board that have traces all around it making them isolated from the rest of the components on the printed circuit board. This makes those parts of the printed circuit board to not be grounded and thus making them not function properly. To avoid that we implement via on those islands to ground them. Below is the figure of the printed circuit board that we made in EagleCad.

6.6.1 Layout

The software design of our printed circuit board was accomplished using EagleCad, which helps to show the best method of using layers when making the layout for the printed circuit board. As in the following figure we can clearly see that our printed circuit board uses two layers due to the complexity of the connections needed for all the traces to provide power to the components and for data transfer. In the layout the top layer is represented with a red copper color and the bottom layer, which is the bottom of the printed circuit board, is represented by green color. The main difficulty with the proper layout of the printed circuit board is the safe routing of all the traces of each of the components from the voltage regulator and the microcontroller. Voltage regulator is like the hub, meaning that is where the highest traffic of traces can be located, other than the microcontroller. To tackle this problem of high number of traces we used two layers of the printed circuit board, the top layer and the bottom layer. The component footprints placements was also very important, the most efficient location for these components on the boards can be seen in the figure below, the reason for placing the footprints in such a manner is to reduce the size of the printed circuit board; hence, reducing the space the board takes up, additionally, smaller printed circuit board means cheaper manufacturing cost.

6.6.2 Zones

Making sure that all the connections on a printed circuit board are secure is one of the most important aspects of designing a PCB. This requires the designer to be diligent in making sure that every connection to each component that is being used in the board design is isolated from other connections and that every other connection, such as ground, can be the common plane. The printed circuit board that we made had the entire zone that was made into ground, and the reason for that is to prevent any short circuit failures. (62)

6.7 PCB Vendor and Assembly

A printed circuit board is manufactured using materials discussed in the earlier chapters, materials like silkscreen, solder mask, substrate and copper. Printed circuit board can be considered as a mechanical part because of the electrical components are soldered on to the printed circuit board which is a mechanical process. All of these components like resistors, capacitor, resonators are connected by using printed wires on the board, these wires are traces defined in the last chapter. These traces or connections act as a highway connecting all the electrical components like sensors, LED, LCD display etc. These highways are etched securely onto the board allowing the flow of electricity

6.7.1 Circuit Board Types

When printed circuit board design is developed, there are variation of printed circuit boards types that must to considered, types like single, flex, double flex. Single layer printed circuit boards are the easiest to make and cost the least amount of money. The type of board that was required for our design, was two layered board. The main reason for that is due to the complexity of all the traces running from components, voltage regulators, and microcontroller.

6.7.2 Surface Mounted

Soldering of electrical or mechanical components can be challenging with different design to accomplish this task it makes it easier. One of the methods of soldering a component is surface mounted, where the conductive part is placed directly on top of the printed circuit board and soldered. This method is used to reduce the surface area of the components when soldered on a printed circuit board; therefore, reducing the cost of manufacturing the board. Surface mounted method also reduces the chances of accidental touching of two conductors like in through hole method. (50) (58)

7 Administrative Content

7.1 Milestone Discussion

For the sake of simplifying the development processes it was decided that we would set a series of internal milestones in order for us to gauge our project progress and work flow. These milestones would be accompanied by a series of dates in order to generally date the amount of time each milestone should take and around when it should be completed for the project to generally be considered on trac. The duration of work for each section has been left as an approximation for how long each milestone took, since some of the development processes was nonsequential. One thing that the group agreed on was that we wanted to start working on the prototype and code base for the project over the fall to spring semester break in order to maximize the amount of implementing, troubleshooting, and debugging the code base for the project this helped us maximize time for troubleshooting.

Table 27: Senior Design Milestones

Milestone	**Duration**	**Dates**
Senior Design 1	16 Weeks	8/20 - 12/6
Project Idea Brainstorming	2.5 Weeks	8/20 - 9/5
Divide and Conquer Report	4 Weeks	8/20 - 9/14
60 Page Report	6 Weeks	9/14 - 11/1
100 Page Report	1.5 Weeks	11/1 - 11/16
PCB and Enclosure Designs	4 Weeks	11/1 - 11/30
120 Page Report	2 Weeks	11/6 - 12/3
Senior Design 2	16 Weeks	1/5 – 4/22
Construct Prototype	4 Weeks	12/3 – 2/28
Test Prototype	4 Weeks	3/1 - 4/15
Peer Presentation	1 Week	2/1
Final Report	2 Weeks	4/16
Final Presentation	1 Week	4/19

7.2 Budget and Finance Discussion

When deciding to design P-Quad we had in mind the cost of this undertaking, with no third part financing we have as a group must spend our money to lift this project of the ground. We tried to find a sponsor but were unsuccessful. Money is a constraint that affects the quality and efficiency of a product that is being developed. But for any project to be successful the aim and the goal is always to minimize cost and optimize efficiency of that project therefore, leading to a robust design and increase the chances for a successful product in the market place. Below is the estimated total budget needed for the entire project. The pre-determined total cost is to run from $450-$500. Overestimations have been made for cost predications to have the right means and finances when starting the physical part of the project. The main budget target is to maintain/decrease the estimated total in USD. The ‘Actual’ column compares the predicted and actual cost after the purchases have been made.

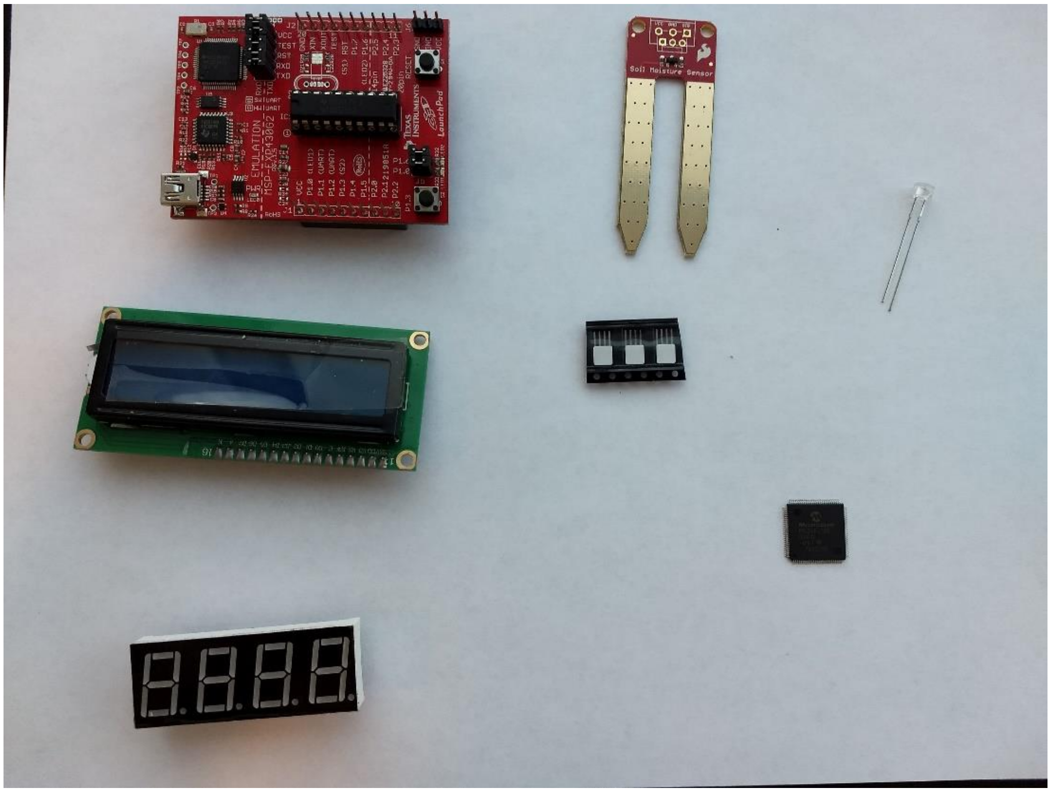
Table 26 shows the estimate cost versus the actual cost:

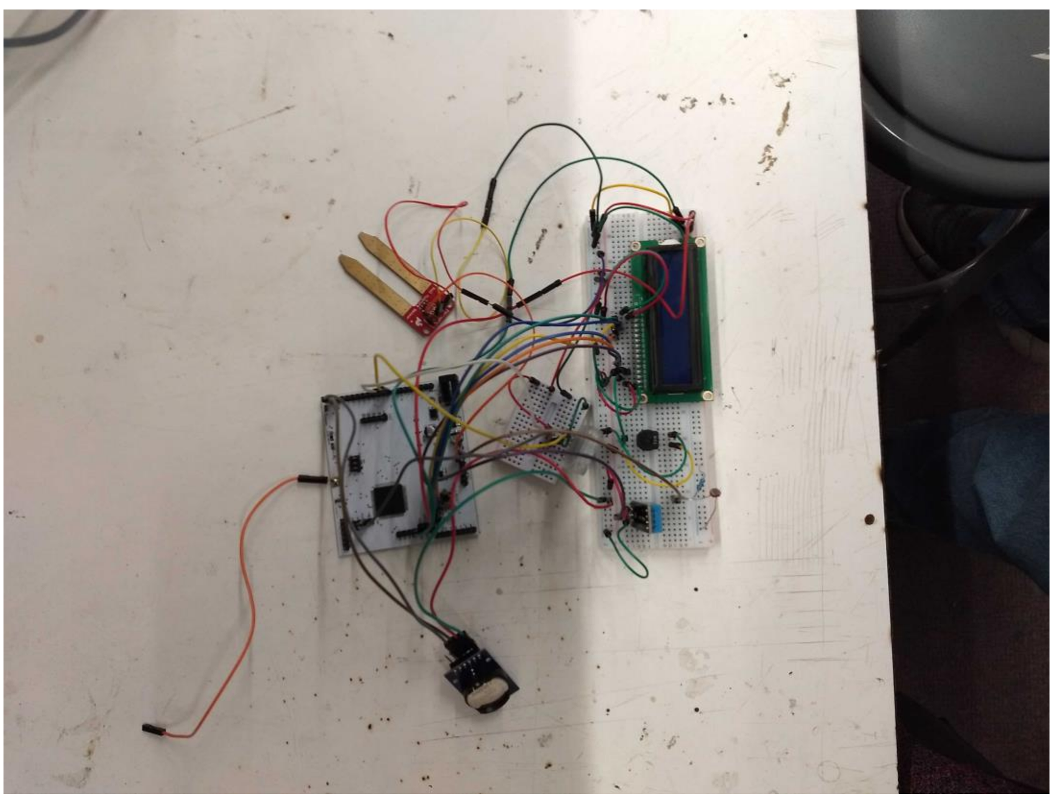
Table 26: Budget Estimations and Bill of Materials

Equipment	**Cost (Pre)**	**Actual**
PCB	$50	$61.68
Components + MCU	$30	$59.04
Sensors	$80	$45.42
Bluetooth Module	$10	$22.99
SD Module	$10	$6.59
RTC	$15	$7.99
Light Fixture	$40	$7.99
LCD display	$30	$18.99
Wooden Enclosure	$25	$42.79
Plant	$10	$10.99
Power Supplies (Battery + Wall Charger)	$20	$32.96
Development Boards	$15	$31.99
Fabrication Tools	$50	$69.25
Total Cost	$435	$418.67

7.3 Design Parts Photographs

Figure 28-29: P-Quad Major Component Picture Pre-SD2 and Late-SD2





7.4 Project Operation

This section details the steps and guidelines the user should follow in order to achieve successful monitoring of the plant's environment. The first step is to ensure that you have one of the two proper power sources either the wall jack connected to an outlet or a 9V battery plugged into the battery backup (rechargeable 9V's are recommended but not required). Once the system is powered on it will automatically enter the setup state without user input required this should take no more than 10-30 seconds to fully initialize all features. After this state it will enter into standby mode from which this point the user can let the system collect and transfer the data by itself. Should the user wish to collect readings at a specific moment they can press the polling button which will immediately gather data in the system storing it as well as displaying it to the LCD output. Additionally, should the user wish to transfer the data over Bluetooth rather than letting the system handle it for them they may press the transmit via Bluetooth button broadcasting the data to a user's computer. For receiving the Bluetooth broadcast data, the user must have a serial communication line open either on a desktop system or a phone terminal, for best results we recommend a desktop COMPORT line to be opened. Once the serial connection has been established the user may then load any terminal interfacing program such as TeraTerm, HyperTerminal, or Gobetwino. For ease of access sake we recommend Gobetwino as it will handle the file creation and formatting however the other two's automatically logging abilities should suffice as well. Once a file of appropriate format has been generated (.CSV, .txt, .log) the user made launch the supporting software either by double clicking the compiled python program or running the python script in the interpreter shell environment with the command (INSERT COMMAND). Once the program is launched a GUI will appear with a blank window the user may then go over to the Graph Generation button. Upon pressing it a file explorer window will open and prompt the user to select one of the support data-set files, note if there are no files in the current directory the user may switch to whichever directory they wish to store their files in. Upon selecting of the file, a prompt will appear asking the user what type of graph they wish to generate at which point they select their preference. Following this a prompt asking for the X and Y axis parameters will pop-up subsequently requiring a choice for generation same as the graph type. After selecting these prompt's choices, a window will appear with the desired graph type of the chosen parameters based on the users request and at this point the user may either save the image to the disk, zoom in or out on certain sections of the graph or close the graph. It is also at this point the user may view the tabular representation of the data in the main window of the program, NOTE this window will not be accessible during the graph generation processes the user may close the graph generation windows at any time to view the tabular data however in order to start another graph generation the file MUST be re-selected. At any point during the projects broadcasting of readings the user may run the notification script to have email alerts sent to them they just need to go to the script and run in the python interpreter with the command python EmailNotification.py. They will then be prompted to enter their gmail address and app password (note an app password is a gmail specific platform dependent password the user may generate under settings) and then enter the COMPORT the device is currently connected to. After this it will prompt the user asking if they wish to begin a monitoring cycle if they choose yes, the system will listen for specific key flags that indicate something is wrong and if detected it will send an email. Otherwise it will state no issues found and ask if it wishes to be rerun if user selects yes, the process will repeat, or if they select no the program will terminate.

8 Appendices

8.1 Appendix A: Copyright Permissions

- All images from the PIC24FJ family datasheets (Figure 6) are property of Microchip permissions for images are pending
- Permission Pending from DLNWare for figures 7-10
- Permission Pending from readthedocs.io for figure 15
- Permission Pending from getmailbird.com for figures 18-19
- Permission Pending from Csharpcorner for figures 20-24
- Permission Pending from Sparkfun for figure 26

8.2 Appendix B: References

(1) https://robinsonloveplants.com/how-many-lumens-per-plant/
(2) https://home.howstuffworks.com/how-to-care-for-house-plants3.htm
(3) http://www.cannagardening.com/how_air_temperature_affects_plants
(4) http://www.just4growers.com/stream/temperature-humidity-and-c02/understanding-the-optimum-temperature-for-plants.aspx
(5) https://mytrellis.com/blog/smstypes
(6) https://en.wikipedia.org/wiki/Soil_moisture_sensor
(7) http://mea.com.au/soil-plants-climate/soil-moisture-monitoring/learning-centre/how-do-you-measure-soil-moisture
(8) https://en.wikipedia.org/wiki/Tensiometer_(soil_science)
(9) https://observant.zendesk.com/hc/en-us/articles/208067926-Monitoring-Soil-Moisture-for-Optimal-Crop-Growth
(10) https://www.lumigrowth.com/par-lumen/
(11) https://www.lumigrowth.com/understanding-lumens/
(12) http://agverra.com/blog/soil-types/
(13) https://en.wikipedia.org/wiki/Photodetector
(14) https://en.wikipedia.org/wiki/Ambient_light_sensor
(15) https://www.parrot.com/us/connected-garden/parrot-pot#technicals
(16) https://patentimages.storage.googleapis.com/50/16/fa/17ec4c76f99903/US8250805.pdf
(17) https://electronicsforu.com/resources/electronics-components/humidity-sensor-basic-usage-parameter
(18) https://www.mouser.com/applications/smart-agriculture-sensors/
(19) https://cropwatch.unl.edu/ssm/sensing
(20) https://en.wikipedia.org/wiki/Data_logger
(21) https://www.mccdaq.com/appnotes/android-data-acquisition-apps.aspx
(22) http://www.vbforums.com/showthread.php?530655-What-is-the-Different-between-Service-Base-database-and-a-local-database
(23) https://www.codeproject.com/Questions/312261/Confused-in-Local-Database-and-Service-Based-Datab
(24) https://www.mongodb.com/scale/relational-vs-non-relational-database
(25) https://hackernoon.com/relational-versus-non-relational-database-d5d1c439fb86
(26) https://acodez.in/sql-and-nosql-an-overview/
(27) https://en.wikipedia.org/wiki/SQL_injection

(28) https://firebase.google.com
(29) https://aws.amazon.com/dynamodb/
(30) https://en.wikipedia.org/wiki/Oracle_Database
(31) https://cloud.oracle.com/iaas/database/pricing
(32) https://www.sitepoint.com/data-serialization-comparison-json-yaml-bson-messagepack/
(33) https://en.wikipedia.org/wiki/JSON
(34) https://en.wikipedia.org/wiki/Regular_expression
(35) https://en.wikipedia.org/wiki/BSON
(36) http://yaml.org
(37) https://en.wikipedia.org/wiki/YAML
(38) https://en.wikipedia.org/wiki/MessagePack
(39) https://en.wikipedia.org/wiki/XML
(40) https://en.wikipedia.org/wiki/MongoDB
(41) https://azure.microsoft.com/en-us/product-categories/databases/
(42) https://azure.microsoft.com/en-us/services/cosmos-db/
(43) https://en.wikipedia.org/wiki/Microsoft_Azure
(44) https://www.autodesk.com/products/eagle/blog/library-basics-part-1-creating-first-package-autodesk-eagle/
(45) https://en.wikipedia.org/wiki/Via_(electronics)
(46) https://bayareacircuits.com/v-scoring-for-printed-circuit-boards/
(47) https://en.wikipedia.org/wiki/Thermal_relief
(48) https://medium.com/@rxseger/surface-mount-electronics-for-hobbyists-easier-than-you-think-fdd2dd60952b
(49) https://learn.sparkfun.com/tutorials/pcb-basics/terminology
(50) https://en.wikipedia.org/wiki/Representational_state_transfer
(51) https://www.webopedia.com/TERM/A/API.html
(52) http://www.benjaminfranklinplumbingiowa.com/what-type-of-plumbing-pipe-is-best/
(53) https://www.rfwireless-world.com/Terminology/NFC-vs-RFID-vs-Bluetooth-vs-wifi.html
(54) https://en.wikipedia.org/wiki/Radio-frequency_identification
(55) https://electronics.howstuffworks.com/gadgets/high-tech-gadgets/rfid3.htm
(56) https://learn.sparkfun.com/tutorials/switch-basics/all
(57) https://electronics.stackexchange.com/questions/102158/what-are-the-most-common-data-storage-devices-in-small-consumer-electronics
(58) https://www.greennet.org.uk/support/understanding-file-sizes
(59) https://www.oransi.com/page/air-filter-guide
(60) https://en.wikipedia.org/wiki/Copper_pour
(61) https://www.nema.org/Standards/Pages/A-Brief-Comparison-of-NEMA-250-and-IEC-60529.aspx#download
(62) http://dlnware.com/dll/Slave-Address-and-Data-Direction
(63) http://dlnware.com/dll/Repeated-START-Sr-Condition
(64) http://dlnware.com/dll/STOP-P-Condition
(65) http://dlnware.com/dll/START-S-Condition

(66) https://facebook.github.io/react-native/blog/2016/03/24/introducing-hot-reloading.html
(67) https://compass.astm.org/EDIT/html_annot.cgi?D1357+95(2011)
(68) https://compass.astm.org/EDIT/html_annot.cgi?E644+11
(69) https://compass.astm.org/EDIT/html_annot.cgi?E879+12
(70) https://na.industrial.panasonic.com/whats-new/embedded-wi-fi-modules-pan93109320-series
(71) https://na.industrial.panasonic.com/whats-new/embedded-wi-fi-modules-pan93109320-series
(72) https://www.atlassian.com/blog/software-teams/version-control-centralized-dvcs
(73) https://www.atlassian.com/git/tutorials/comparing-workflows